RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE

For Quadrant: **Retail Electric Quadrant**

Requesters: **Open ADE Task Force**
Request No.: **R10008**
Request Title: **Energy Services Provider Interface Standard**

1. RECOMMENDED ACTION:

X	Accept as requested
	Accept as modified below
	Decline

EFFECT OF EC VOTE TO ACCEPT RECOMMENDED ACTION:

X	Change to Existing Practice
	Status Quo

2. TYPE OF DEVELOPMENT/MAINTENANCE

Per Request:

X	Initiation
	Modification
	Interpretation
	Withdrawal
X	Principle
X	Definition
X	Business Practice Standard
	Document
	Data Element
	Code Value
	X12 Implementation Guide
	Business Process Documentation

Per Recommendation:

X	Initiation
	Modification
	Interpretation
	Withdrawal
X	Principle
X	Definition
X	Business Practice Standard
	Document
	Data Element
	Code Value
	X12 Implementation Guide
	Business Process Documentation

3. RECOMMENDATION

SUMMARY:

The UCAIug OpenADE Task Force submitted a request for the initiation of NAESB Model Business Practices on July 29, 2010 (R10008) to standardize the interface which allows for the exchange of ~~e~~Energy ~~u~~Usage ~~i~~Information between designated parties. The UCAIug OpenADE Task Force provided the artifacts on which these Model Business Practices were based.

These Model Business Practices will build on the NAESB Energy Usage Information (EUI) Model and, subject to the Governing Documents and any requirements of the Applicable Regulatory Authority, will help enable Retail Customers to share ~~e~~Energy ~~u~~Usage ~~i~~Information with Third Parties who have acquired the right to act in this role. This Energy Services Provider Interface (ESPI) will provide a consistent method for Retail Customers to authorize a Third Party to gain access to ~~e~~Energy ~~u~~Usage ~~data~~Information. Doing so will help enable Retail Customers to choose Third Party products to assist them to better understand their energy usage and to make more economical decisions about their usage. ESPI will contribute to the development of

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE
For Quadrant: Retail Electric Quadrant

Requesters: Open ADE Task Force
Request No.: R10008
Request Title: Energy Services Provider Interface Standard

an open and interoperable method for Third Party authorization and machine-to-machine exchange of Retail Customer Energy ~~u~~Usage ~~i~~Information.

RECOMMENDED STANDARDS:

REQ.21 ENERGY SERVICES PROVIDER INTERFACE

EXECUTIVE SUMMARY

This document establishes the Model Business Practices for the Energy Services Provider Interface (ESPI). For Retail Customers to better realize the benefits of the Smart Grid, Retail Customer ~~related data (e.g.,~~Energy ~~u~~Usage ~~i~~Information~~, etc.)~~ should be made available in a timely manner to the Retail Customer and to the Authorized Third Parties chosen by the Retail Customer.

ESPI encompasses a variety of interactions between Retail Customers, Distribution Companies, and Third Parties. In a business environment where best practices are voluntary, Model Business Practices should be applied within the context of regulatory requirements and agreements. These Model Business Practices define ESPI as a specific available interface, but any obligation to use it would be established by Governing Documents and Applicable Regulatory Authority rules and regulations. and not by these Model Business Practices.

These Model Business Practices are not intended to apply to the Data Custodian's disclosure, collection, use and handling of Energy Usage Information in connection with the Data Custodian's or its agents' utility services, product or service fulfillment or billing and collection activities.

INTRODUCTION

The North American Energy Standards Board (NAESB) is a voluntary non-profit organization comprised of members from all aspects of the natural gas and electric industries. Within NAESB, the Retail Electric Quadrant (REQ) and the Retail Gas Quadrant (RGQ) focus on issues impacting the retail sale of energy to Retail Customers. REQ / RGQ Model Business Practices are intended to provide guidance to Distribution Companies, Suppliers, and other Market Participants involved in providing energy service to Retail Customers. The focus of these Model Business Practices is the Energy Services Provider Interface.

The purpose of ESPI is to provide a consistent and broadly applicable interface to enable Retail Customer authorization of exchange of EUI from Data Custodians to Third Parties. For the purpose of the descriptions of interactions in ESPI, actions of contracted agents of a Distribution Company are considered the actions of the Distribution Company.

These Model Business Practices are voluntary and do not address policy issues that are the subject of state legislation or regulatory decisions. These voluntary Model Business Practices have been adopted by NAESB with the realization that, as the industry evolves, additional and amended Model Business Practices may be necessary. Any industry participant seeking additional or amended Model Business Practices (including principles, definitions, data elements, process descriptions, and technical implementation instructions) should submit a request to the NAESB office, detailing the change, so that the appropriate process may take place to amend the Model Business Practice.

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE

For Quadrant: **Retail Electric Quadrant**

Requesters: **Open ADE Task Force**
Request No.: **R10008**
Request Title: **Energy Services Provider Interface Standard**

BUSINESS PROCESSES AND PRACTICES

Overview

REQ.21.1 Principles

REQ.21.1.1 The processes for ESPI should minimize the complexity associated with authorizing Third Parties to access Retail Customers' ~~e~~Energy ~~u~~Usage ~~data~~Information.

REQ.21.1.2 The processes associated with ESPI are subject to and should be consistent with any related requirements established by the Governing Documents and Applicable Regulatory Authority.

REQ.21.2 Definitions

REQ.21.2.B Technical Definitions

REQ.21.2.B.1 ~~**Authorizing Entity:** An Entity (e.g. PUC, Distribution Company) who approves Third Parties to utilize ESPI-compliant system(s) within a jurisdiction.~~

~~**REQ.21.2.B.2**~~ **Third Party:** An Entity which provides some service to a Retail Customer based on Energy Usage ~~i~~Information for the Retail Customer to which it does not have direct access and over which it has no direct authority ~~over~~ other than: the Data Custodian and its contracted agents, the Applicable Regulatory Authority, ISOs or other regional entities. ~~A Third Party relies on a Data Custodian to provide access to Retail Customer information.~~

REQ.21.2.B.~~3~~2 **Authorized Third Party:** A Third Party that is permitted to receive EUI in accordance with applicable law, regulation, the Governing Documents and any requirements of the Applicable Regulatory Authority ~~has been approved by an Authorizing Entity for the relevant jurisdiction~~ and has met the requirements of the Applicable Regulatory Authority and Governing Documents to utilize the Energy Services Provider Interface.

REQ.21.2.B.43 **Energy Service Provider Interface:** A standardized machine-to machine interface that permits a Data Custodian to share, at the Retail Customer's request ~~and under the Retail Customer's~~ or direction, a broad set of that Retail Customer's Energy Usage Information held by that Data Custodian ~~data~~ with Authorized Third Parties.

REQ.21.2.B.54 **Personally Identifiable Information:** Subject to the Governing Documents and any requirements of the Applicable Regulatory Authority, the~~any~~ following information about an individual Retail Customer:~~maintained, including~~ (1) any information that can be reasonably used to distinguish or trace a~~n individual's~~Retail Customer's identity, such as name, social security number, date and place of birth, mother~~'~~'s maiden name, or biometric records; and (2) any other information that is reasonably linked or linkable to a~~n individual~~Retail Customer, such as medical, educational, financial, and employment information[1]. EUI may be included as PII in some jurisdictions or by some Data Custodians.

REQ.21.2.B.65 **Data Custodian:** A Data Custodian holds Retail Customer ~~resource~~Energy Usage ~~i~~Information and will share this information with Authorized Third Parties only in accordance with the Governing Documents, any requirements of the Applicable Regulatory Authority and, subject to the Governing Documents and the requirements of the Applicable Regulatory Authority, at the request or direction of the Retail Customer. A Data Custodian typically has direct access to the ~~pertinent~~Energy Usage ~~i~~Information (e.g., by directly acquiring electricity usage data from a meter). A Data Custodian may be a Distribution Company.

REQ.21.2.B.76 **Energy Usage Information*:*** Any information and data from a smart meter identifiable to an individual~~concerning a~~ Retail Customer~~'s use of energy.~~ concerning that Retail Customer's energy usage, which may be made available pursuant to the Governing Documents consistent with any requirements of the Applicable Regulatory Authority.

REQ.21.2.C Acronyms

[1] Definition based upon NIST Special Publication 800-122, Guide to Protecting the Confidentiality of Personally Identifiable Information (PII) April 2010, page 2-1.

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE
For Quadrant: Retail Electric Quadrant

Requesters: Open ADE Task Force
Request No.: R10008
Request Title: Energy Services Provider Interface Standard

Abbreviation / Acronym	Meaning
ADE	Automatic Data Exchange
ESPI	Energy Services Provider Interface
EUI	Energy Usage Information
NISTIR	National Institute of Standards and Technology Interagency Report
PII	Personally Identifiable Information

REQ.21.3 Model Business Practices

REQ.21.3.1 General Practices for Energy Services Provider Interface (ESPI)

REQ.21.3.1.1 To the extent required by the Applicable Regulatory Authority~~,~~ or as otherwise agreed by the Data Custodian consistent with any requirements of the Applicable Regulatory Authority, Authorized Third Parties and Data Custodians should exchange Retail Customer's EUI at the Authorized Third Party's or Retail Customer's request or direction pursuant to the requirements as set forth in this NAESB REQ.21, subject to the Governing Documents.

REQ.21.3.1.2 The ESPI relationship requires a set of agreements between a Retail Customer-Authorized Third Party, a Retail Customer-Data Custodian, and an Authorized Third Party-Data Custodian to help ensure that the appropriate information is provided as needed and other information access is restricted.

REQ.21.3.1.3 A Third Party should not be able to access EUI or any other ~~Personally Identifiable Information (~~PII~~)~~ from a Data Custodian, except as permitted or required by the Governing Documents, the Applicable Regulatory Authority or, subject to the Governing Documents and the requirements of the Applicable Regulatory Authority, as otherwise requested or directed by the Retail Customer. ~~PII may only be provided to a Third Party by the Retail Customer.~~

REQ.21.3.1.4 Subject to the Governing Documents and Applicable Regulatory Authority, ESPI should enable a Retail Customer to share EUI for such Retail Customer with Authorized Third Parties who have acquired the right to act in this role.

REQ.21.3.1.5 A system conforming to ESPI should allow exchange of Energy U~~u~~sage ~~i~~Information without requiring Third Party access to any other PII.

REQ.21.3.1.6 All information exchanged by ESPI should be secure in accordance with the security recommendations ~~stated~~referenced herein. Such recommendations are subject to the relevant Governing Documents and any requirements of the Applicable Regulatory Authority.

REQ.21.3.1.7 A Retail Customer should have the ability to ~~authorize~~request or direct the Data Custodian to release EUI for such Retail Customer to an Authorized Third Party who has acquired the right to act in this role, subject to the Governing Documents and Applicable Regulatory Authority.

REQ.21.3.1.8 Subject to the Governing Documents and any requirements of the Applicable Regulatory Authority, a Retail Customer should have the ability to ~~authorize~~request or direct that multiple Authorized Third Parties ~~to~~ have limited time based access to specified EUI ~~or other types of information for such Retail Customer~~, with any default expiration for such access established by such Governing Documents or any requirements of the Applicable Regulatory Authority.

REQ.21.3.1.9 Subject to the Governing Documents and any requirements of the Applicable Regulatory Authority, a Retail Customer should have the ability to designate a specific expiration date, extend any specific expiration date, or indicate an open-ended access timeframe other than the default access period.

REQ.21.3.1.10 A system conforming to ESPI should have the capability to support the Retail Customers' ability to select ~~/~~or revoke which Authorized Third Parties are ~~authorized for~~permitted access to EUI for that Retail Customer.

REQ.21.3.1.11 A system conforming to ESPI should have the capability to notify the relevant Authorized Third Parties, Data Custodian(s) and

Retail Customer(s) when access has been granted, access has been changed, or access has been revoked or otherwise terminated for a UsagePoint.

REQ.21.3.1.12 ~~Subject to the Governing Documents and Applicable Regulatory Authority, a~~ A system conforming to ESPI should be consistent with ~~the applicable~~any guidelines around security and authorization for Third Party data access as ~~set forth in NISTIR 7628~~determined to be applicable by the Governing Documents or any requirements of the Applicable Regulatory Authority.

REQ.21.3.1.13 Future versions of ESPI should be backwards compatible, including provisions for exchanging versioning information and negotiating interface capabilities.

REQ.21.3.1.14 Any Third Party wishing to access EUI via ESPI must establish and maintain a trusted relationship with each Data Custodian who provides an ESPI compliant system. Subject to the Governing Documents and Applicable Regulatory Authority, both the Data Custodian and the Authorized Third Party should disallow EUI access requests from Entities who are not Authorized Third Parties.

REQ.21.3.1.15 Subject to the Governing Documents and any requirements of the Applicable Regulatory Authority, confidentiality should be maintained during communications of any EUI or other PII~~information~~.

REQ.21.3.1.16 Subject to the Governing Documents and any requirements of the Applicable Regulatory Authority, a Third Party~~ies~~ must be ~~authorized by the Authorizing Entity and/or the Data Custodian to be~~ an Authorized Third Party ~~and~~to utilize the Data Custodian's ESPI compliant system and must maintain ~~their~~its status as an Authorized Third Party. In the case of a transfer, merger, reorganization or sale of or involving an Authorized Third Party, the Data Custodian is not required to notify the Retail Customer of the transfer, merger, reorganization or sale and a new authorization, request or direction is not required for the Distribution Company to continue to disclose the EUI to the transferee, subsequent owner or successor of the Authorized Third Party.

REQ.21.3.1.17 If ~~an Authorizing Entity~~it exists within a jurisdiction, ~~the Authorizing Entity should make available to Retail Customers~~ a list of Third Parties who have been authorized by the Applicable Regulatory Authority or the Data Custodian to use ESPI in accordance with and subject to the Governing Documents and any requirements of the Applicable Regulatory Authority should be made available by the Applicable Regulatory Authority or the Data Custodian, as applicable.

REQ.21.3.1.18 Subject to the Governing Documents and Applicable Regulatory Authority, EUI should be made available to Authorized Third Parties (as directed by the Retail Customer) in a reasonable and timely fashion. It is recognized that a Data Custodian providing EUI directly from the smart meter or before the data is validated for billing purposes can only provide the EUI as that data is registered by or recorded in the smart meter. Retail Customers and the Third Parties to which such data is disclosed should acknowledge that there are inherent limitations in EUI disclosed before the Data Custodian has verified and validated it for billing purposes. Further, these Model Business Practices do not establish or recommend any intervals at or for which EUI will or should be provided or available.~~—~~

REQ.21.3.1.19 When the required Authorized relationship described in this recommendation for an Entity is revoked or otherwise terminated, access to EUI by such Entity via ESPI should not be granted.

REQ.21.3.1.20 Participants in ESPI and their relationships should be identified ~~with~~by globally unique identifiers.

REQ.21.3.1.21 ~~Procedures for the creation and dissolution of trusted relationships between any two parties should be preconditions for the use of ESPI. The standardization of these procedures, however, is outside the scope of this Model Business Practice.~~

~~REQ.21.3.1.22~~ Upon dissolution of any of the required trusted relationships for an Entity, any ESPI relationships should be terminated and parties notified via a defined method.

REQ.21.3.1.2~~2~~3 If and when the relationships or criteria change, pursuant to these ~~m~~Model ~~b~~Business ~~p~~Practices and/or as agreed to among any two or more of the parties, ~~change,~~ all affected parties should be notified via a defined method.

REQ.21.3.1.234 Interoperable and widely supported technologies should be used to help ensure adoption regardless of which development and deployment platforms are used.

REQ.21.3.1.245 The technologies chosen should be well specified, with active communities, tools, and/or frameworks available.

REQ.21.3.1.256 ~~Technologies chosen should be compatible and interoperable with technologies specified for access to HAN resources.~~

~~**REQ.21.3.1.27**~~ To the extent required by the Applicable Regulatory Authority, Authorized Third Parties and Data Custodians should follow privacy guidance recommended in NAESB REQ.22, "Third Party Access to Smart Meter-based Information", subject to Governing Documents and any requirements of the Applicable Regulatory Authority.

REQ.21.3.1.286 This Model ~~b~~Business ~~p~~Practice only ~~constrains~~applies to applications purporting to conform to it. It is not intended to be applicable for all customer information transfers to Authorized Third Parties, but rather, only those transfers between applications conforming to ESPI; provided not all data elements must be used in order to be ESPI conformant.

REQ.21.3.1.279 Future versions of ESPI, and extensions employed by Authorized Third Parties and Data Custodians to exchange Retail Customer's EUI at the Retail Customer's request where not specified by ESPI, should conform to NAESB REQ~~.~~.18, as EUI may be updated from time to time.

REQ.21.4 Models

REQ.21.4.1 Profile of REQ.18 Energy Usage Information Model

The following model represents the implementable profile for ESPI of NAESB PAP10 EUI model. Note that associations stereotyped <<link>> are marked as Non-navigable, since they are actually represented using atom:link.

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE

For Quadrant: **Retail Electric Quadrant**

Requesters: **Open ADE Task Force**
Request No.: **R10008**
Request Title: **Energy Services Provider Interface Standard**

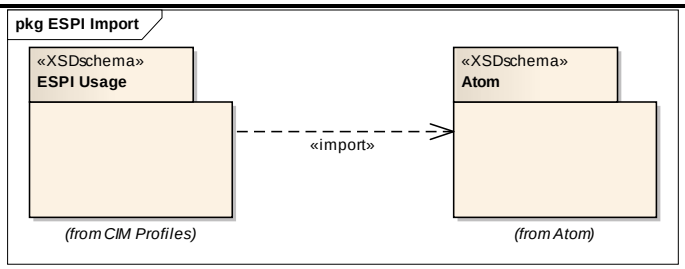

Figure 1: ESPI Import

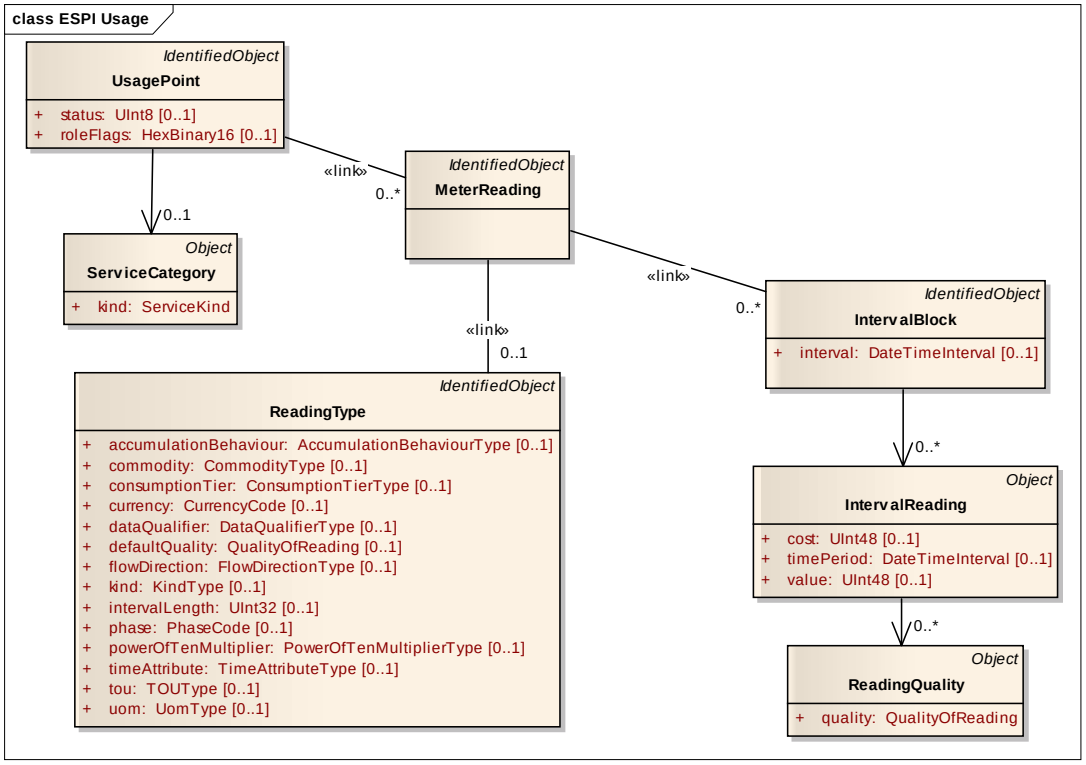

Figure 2: ESPI Usage

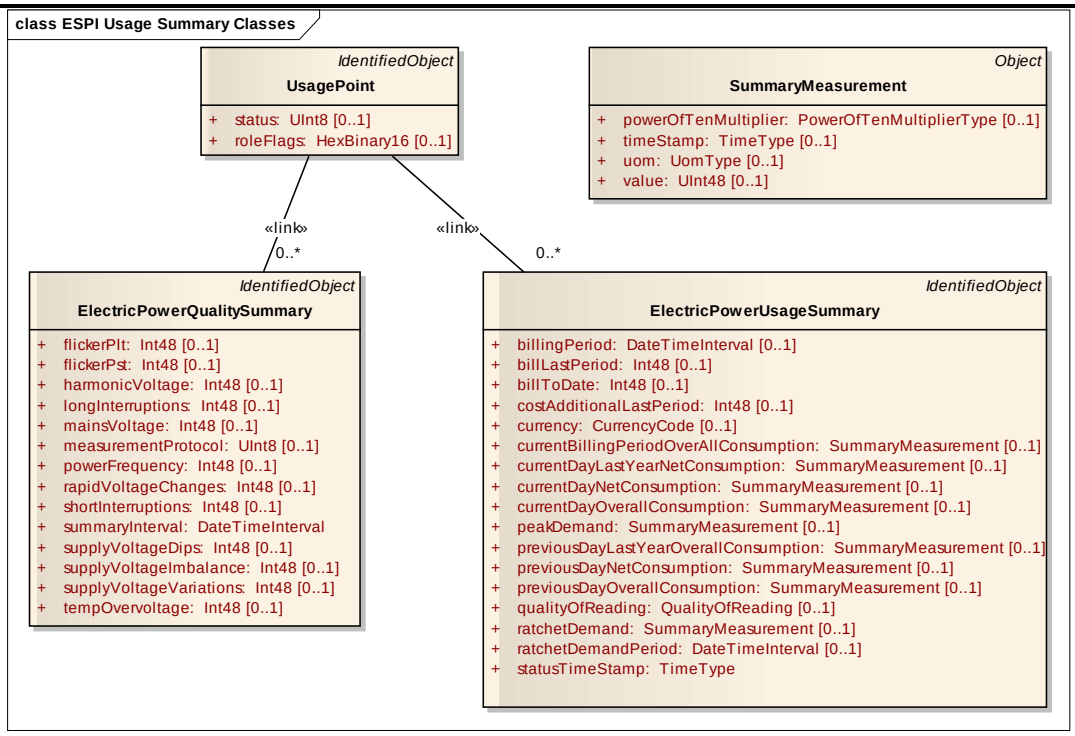

Figure 3: ESPI Usage Summary Classes

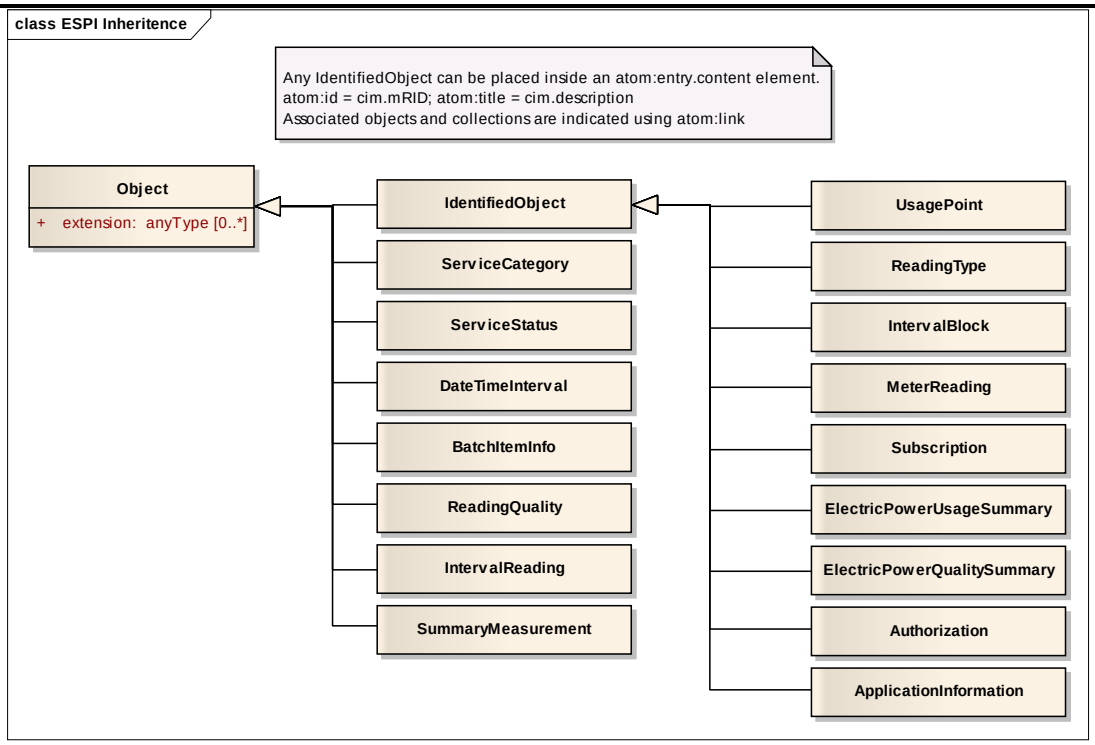

Figure 4: ESPI Inheritence

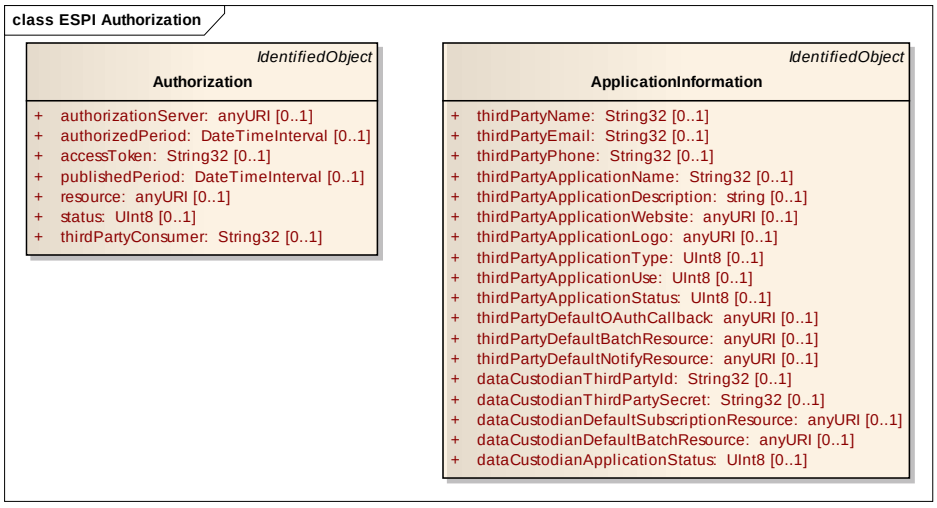

Figure 5: ESPI Authorization

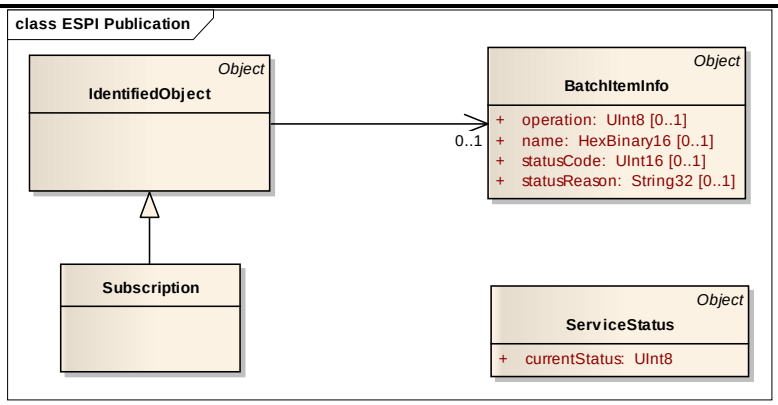

Figure 6: ESPI Publication

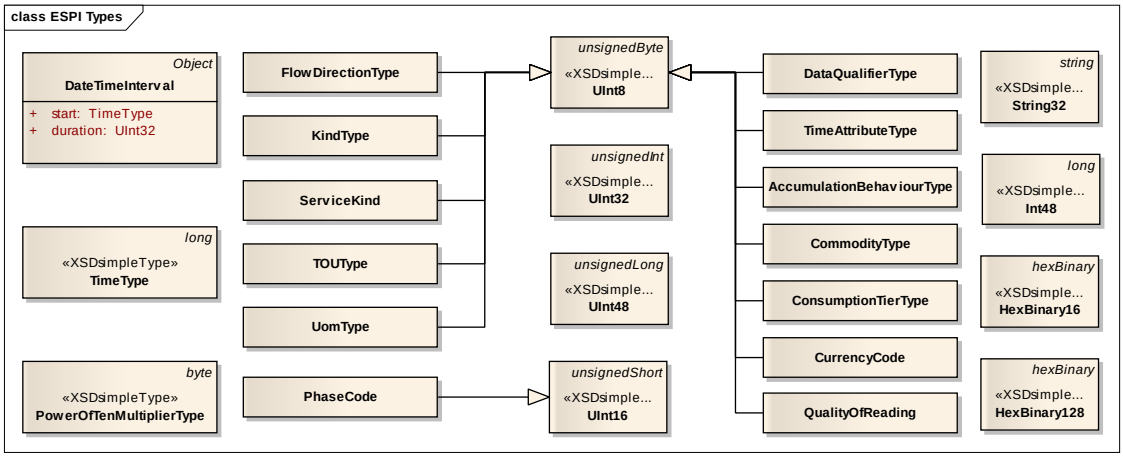

Figure 7: ESPI Types

BatchItemInfo

Includes elements that make it possible to include multiple transactions in a single (batch) request.

Name	Type	Description
operation	*UInt8*	Specifies the operation requested of this item. 0=Create 1=Read 2=Update 3=Delete
name	*HexBinary16*	An identifier for this object that is only unique within the containing collection.
statusCode	*UInt16*	Indicates the status code of the associated transaction.

Name	Type	Description
		200 - Ok 201 - Created 204 - No Content 301 - Moved Permanently 302 - Redirect 304 - Not Modified 400 - Bad Request 401 - Unauthorized 403 - Forbidden 404 - Not Found 405 - Method Not Allowed 410 - Gone 500 - Internal Server Error
statusReason	*String32*	Indicates the reason for the indicated status code.

Object

Superclass of all object classes to allow extensions.

Name	Type	Description
extension	*anyType*	Contains an extension.

ServiceStatus

Contains the current status of the service.

Name	Type	Description
currentStatus	*UInt8*	The current status of the service. 0 = Unavailable 1 = Normal, operational

Subscription

Defines the parameters of a subscription between ~~t~~Third ~~p~~Party and ~~d~~Data ~~c~~Custodian.

ApplicationInformation

Contains information about a Third Party Application requesting access to the DataCustodian services. Information requested may include items such as Organization Name, Website, Contact Info, Application Name, Description, Icon, Type, default Notification and Callback endpoints, and may also include agreement with terms of service.

Name	Type	Description
thirdPartyName	*String32*	The name of the organization to which access will be granted.

Name	Type	Description
thirdPartyEmail	*String32*	The e-mail address of the organization to which access will be granted. (For debugging - not to be shared with customers)
thirdPartyPhone	*String32*	The phone number of the organization to which access will be granted. (For debugging - not to be shared with customers)
thirdPartyApplicationName	*String32*	The name of the application to which access will be granted.
thirdPartyApplicationDescription	*string*	A description of the application.
thirdPartyApplicationWebsite	*anyURI*	The link to the main page of the application.
thirdPartyApplicationLogo	*anyURI*	The link to the logo image for the application. Size greater than 180 x 150 may be cropped or reduced.
thirdPartyApplicationType	*UInt8*	A code indicating the type of the application. Defined types are: 1 - Web Application 2 - Desktop Application 3 - Mobile Application
thirdPartyApplicationUse	*UInt8*	A code indicating the expected use of the application. Defined uses are: 1 - Energy management 2 - Comparisons 3 - Government
thirdPartyApplicationStatus	*UInt8*	A code indicating the current status of the application. Defined statuses are: 1 - Development 2 - Production (Live) 3 - Retired (Remove)
thirdPartyDefaultOAuthCallback	*anyURI*	The default redirect back to the application after authorization grant.
thirdPartyDefaultBatchResource	*anyURI*	The default endpoint for asynchronous delivery of Batch data using push.
thirdPartyDefaultNotifyResource	*anyURI*	The default endpoint for third party notification of Batch data availability, that is then requested from the ~~d~~Data ~~c~~Custodian via the Batch resource.
dataCustodianThirdPartyId	*String32*	A key to be associated with this application, to be provided in OAuth requests. (Provided

Name	Type	Description
		by dataCustodian, cannot be modified)
dataCustodianThirdPartySecret	*String32*	A secret to be associated with this application, used to sign OAuth requests. (Provided by dataCustodian, cannot be modified)
dataCustodianDefaultSubscriptionResource	*anyURI*	The default endpoint for Subscription requests. (Provided by dataCustodian, updated in approved applications objects, cannot be modified by third party)
dataCustodianDefaultBatchResource	*anyURI*	The default endpoint for Batch requests. (Provided by dataCustodian, updated in approved applications objects, cannot be modified by third party)
dataCustodianApplicationStatus	*UInt8*	A code indicating the current status of the application. (Provided by dataCustodian, cannot be modified) Defined statuses are: 1 - Review 2 - Production (Live) 3 - On hold 4 - Revoked

Authorization

Represents a permission granted by an owner for access to a resource.

Name	Type	Description
authorizationServer	*anyURI*	Contains the URI link to the authorization endpoint associated with this authorization.
authorizedPeriod	*DateTimeInterval*	Restricts access to requests or subscriptions within this date time interval.
accessToken	*String32*	Contains the access token associated with this authorization.
publishedPeriod	*DateTimeInterval*	Restricts access to only the objects within the associated resource that were published within this date time interval.
resource	*anyURI*	Contains the identifier of the resource, same as was specified in OAuth "scope". **[Please confirm CSWG has reviewed and approved.]**
status	*UInt8*	The status of this authorization. 0 - Revoked 1 - Active
thirdPartyConsumer	*String32*	Contains the identifier for the Third Party.

IdentifiedObject

This is a root class to provide common naming attributes for all classes needing naming attributes.

ElectricPowerQualitySummary

A summary of power quality events. This information represents a summary of power quality information typically required by customer facility energy management systems. It is not intended to satisfy the detailed requirements of power quality monitoring. All values are as defined by measurement Protocol during the period. The standards typically also give ranges of allowed values; the information attributes are the raw measurements, not the "yes/no" determination by the various standards. See referenced standards for definition, measurement protocol- and period.

Name	Type	Description
flickerPlt	*Int48*	A measurement of long term Rapid Voltage Change in hundredths. flickerPlt is derived from 2 hours of Pst values (12 values combined in cubic relationship).
flickerPst	*Int48*	flickerPst is a value measured over 10 minutes that characterizes the likelihood that the voltage fluctuations would result in perceptible light flicker. A value of 1.0 is designed to represent the level that 50% of people would perceive flicker in a 60 watt incandescent bulb. The value reported is represented as an integer in hundredths.
harmonicVoltage	*Int48*	A measurement of the Harmonic Voltage during the period. For DC, distortion is with respect to a signal of zero Hz.
longInterruptions	*Int48*	A count of Long Interruption events (as defined by measurementProtocol) during the summary interval period.
mainsVoltage	*Int48*	A measurement of the Mains [Signaling] Voltage during the summary interval period in uV.
measurementProtocol	*UInt8*	A reference to the source standard used as the measurement protocol definition. Examples are: 0 = "IEEE1519-2009" 1 = "EN50160"
powerFrequency	*Int48*	A measurement of the power frequency during the summary interval period in uHz.

Name	Type	Description
rapidVoltageChanges	*Int48*	A count of Rapid Voltage Change events during the summary interval period
shortInterruptions	*Int48*	A count of Short Interruption events during the summary interval period
summaryInterval	*DateTimeInterval*	Interval of summary period
supplyVoltageDips	*Int48*	A count of Supply Voltage Dip events during the summary interval period
supplyVoltageImbalance	*Int48*	A count of Supply Voltage Imbalance events during the summary interval period
supplyVoltageVariations	*Int48*	A count of Supply Voltage Variations during the summary interval period
tempOvervoltage	*Int48*	A count of Temporary Overvoltage events (as defined by measurementProtocol) during the summary interval period

ElectricPowerUsageSummary

Summary of usage for a billing period [We assume monthly billing period, however, many of the data elements in this section indicate daily or incremental information. Accordingly, these data elements need clarification/revision for consistency.]

Name	Type	Description
billingPeriod [It appears these items are related to Monthly usage: billingPeriod, billLastPeriod, billtodate, costAdditionalLastPeriod, currency, currentBillingPeriodOverAllConsumption, peakDemand, ratchetDemand, ratchedDemandPeriod However, these values appear to be net daily usage: currentDaylastyearNetConsumption, currentDayNetConsumption,	*DateTimeInterval*	The billing period to which the included measurements apply

Name	Type	Description
currentDayOverallConsumption, previousDatLastyearOverallConsumption, previousDaynetConsumption, previousDayOverallConsumption]		
billLastPeriod	*Int48*	The amount of the bill for the previous period , in millionths of the currency specified in the ReadingType for this reading (e.g. 840 = USD, US dollar).
billToDate [running total$ is typically not calculated until end of BillCycle due to step rates which can be associated with peak usage or aggregated usage.]	*Int48*	The bill amount related to the billing period as of the date received, in millionths of the currency specified in the ReadingType for this reading. (e.g. 840 = USD, US dollar).
costAdditionalLastPeriod	*Int48*	Additional charges from the last billing period, in millionths of the currency specified in the ReadingType for this reading. (e.g. 840 = USD, US dollar).
currency	*CurrencyCode*	The ISO 4217 code indicating the currency applicable to the bill amounts in the summary. See list at http://www.unece.org/cefact/recommendations/rec09/rec09_ecetrd203.pdf
currentBillingPeriodOverAllConsumption	*SummaryMeasurement*	The total consumption for the billing period
currentDayLastYearNetConsumption	*SummaryMeasurement*	The amount of energy consumed one year ago interpreted as same day of week same week of year (see ISO 8601). [This implies daily updates which are not easily reported with current designs. This comment also applies to any daily information or data element requiring daily accumulations.]
currentDayNetConsumption	*SummaryMeasurement*	Net consumption for the current day (delivered - received)
currentDayOverallConsumption	*SummaryMeasurement*	Overall energy consumption for the current day

Name	Type	Description
peakDemand	*SummaryMeasurement*	Peak demand recorded for the current period [This is assumed to be monthly but peak demand may only to stored when it is used for billing.]
previousDayLastYearOverallConsumption	*SummaryMeasurement*	The amount of energy consumed on the previous day one year ago interpreted as same day of week same week of year (see ISO 8601).
previousDayNetConsumption	*SummaryMeasurement*	Net consumption for the previous day [Daily information on net amounts for net metering may not be available. In addition, the term net metering can have varying definitions.]
previousDayOverallConsumption	*SummaryMeasurement*	The total consumption for the previous day
qualityOfReading	*QualityOfReading*	Indication of the quality of the summary readings
ratchetDemand	*SummaryMeasurement*	The current ratchet demand value for the ratchet demand period [If an account is not billed on a ratchet demand, this value may not be available.]
ratchetDemandPeriod	*DateTimeInterval*	The period over which the ratchet demand applies
statusTimeStamp	*TimeType*	Date/Time status of this UsageSummary

ServiceCategory

Category of service provided to the customer.

Name	Type	Description
kind	*ServiceKind*	Service classification Examples are: 0 - electricity 1 - gas The list of specific valid values per the standard are itemized in ServiceKind.

UsagePoint

Logical point on a network at which consumption or production is either physically measured (e.g. metered) or estimated (e.g. unmetered street lights).

Name	Type	Description
status	*UInt8*	Specifies the current status of this usage point. The only valid values are: 0 = off 1 = on
roleFlags	*HexBinary16*	Specifies the roles that this usage point has been assigned. Bit 1 - isMirror Bit 2 - isPremisesAggregationPoint Bit 3 - isPEV Bit 4 - isDER Bit 5 - isRevenueQuality Bit 6 - isDC Bit 7-16 - Reserved

HexBinary128 «XSDsimpleType»

A 128-bit field encoded as a hex string (32 characters / 16 octets)

HexBinary16 «XSDsimpleType»

A 16-bit field encoded as a hex string (4 characters / 2 octets)

Int48 «XSDsimpleType»

Signed integer, max inclusive 281474976710655 (2^48-1), restriction of xs:long

String32 «XSDsimpleType»

Character string of max length 32

UInt16 «XSDsimpleType»

Unsigned integer, max inclusive 65535 (2^16-1), same as xs:unsignedShort

UInt32 «XSDsimpleType»

Unsigned integer, max inclusive 4294967295 (2^32-1), same as xs:unsignedInt

UInt48 «XSDsimpleType»

Unsigned integer, max inclusive 281474976710655 (2^48-1), restriction of xs:unsignedLong

UInt8 «XSDsimpleType»

Unsigned integer, max inclusive 255 (2^8-1), same as xs:unsignedByte

AccumulationBehaviourType

The only valid values are:

0 = Not Applicable
1 = BulkQuantity
3 = Cumulative
4 = DeltaData
6 = Indicating
9 = Summation
12 = Instantaneous

CommodityType

The only valid values are:

0 = Not Applicable
1 = Electricity secondary metered value (a premise meter is typically a secondary meter)
2 = Electricity primary metered value
4 = Air
7 = NaturalGas
8 = Propane
9 = PotableWater
10 = Steam
11 = WasteWater
12 = HeatingFluid
13 = CoolingFluid

ConsumptionTierType

The only valid values are:

0 = Not Applicable
1 = Block Tier 1
2 = Block Tier 2
3 = Block Tier 3
4 = Block Tier 4
5 = Block Tier 5
6 = Block Tier 6
7 = Block Tier 7
8 = Block Tier 8
9 = Block Tier 9
10 = Block Tier 10
11 = Block Tier 11
12 = Block Tier 12
13 = Block Tier 13
14 = Block Tier 14
15 = Block Tier 15
16 = Block Tier 16

CurrencyCode

Follows codes defined in ISO 4217. Full list at tiny.cc/4217.

0 - Not Applicable
36 - Australian Dollar
124 - Canadian Dollar
840 - US Dollar
978 - Euro

DataQualifierType

The only valid values are:

0 = Not Applicable
2 = Average
8 = Maximum
9 = Minimum
12 = Normal

DateTimeInterval

Interval of date and time. End is not included because it can be derived from the start and the duration.

Name	Type	Description
start	*TimeType*	Date and time that this interval started.
duration	*UInt32*	Duration of the interval, in seconds.

FlowDirectionType

The only valid values are:

0 = Not Applicable
1 = Forward
19 = Reverse

KindType

The only valid values are:

0 = Not Applicable
3 = Currency
8 = Demand
12 = Energy
37 = Power

PhaseCode

The only valid values are:

0 = Not Applicable
129 = Phase AN
128 = Phase A
132 = Phase AB
64 = Phase BN
64 = Phase B
32 = Phase CN
32 = Phase C
224 = Phase ABC
66 = Phase BC
40 = Phase CA
512 = Phase S1
256 = Phase S2
768 = Phase S1S2
513 = Phase S1N
257 = Phase S2N
769 = Phase S1S2N

PowerOfTenMultiplierType «XSDsimpleType»

The only valid values are:

0 = None
1 = deca=x10
2 = hecto=x100
-3 = mili=x10-3
3 = kilo=x1000
6 = Mega=x106
-6 = micro=x10-3
9 = Giga=x109

QualityOfReading

List of codes indicating the quality of the reading, using specification:

0 – valid (validated)
7 - manually edited
8 - estimated
10 - questionable
11 - derived
12 - projected (forecast)
13 - mixed
14 - raw
15 - normalized for weather
16 - other

ServiceKind

The only valid values are:

0 - electricity
1 - gas
2 - water
4 - pressure
5 - heat
6 - cold
7 - communication
8 - time

SummaryMeasurement

An aggregated summary measurement reading.

Name	Type	Description
powerOfTenMultiplier	*PowerOfTenMultiplierType*	The multiplier part of the unit of measure, e.g. "kilo" (k)
timeStamp	*TimeType*	The date and time (if needed) of the summary measurement.
uom	*UomType*	The units of the reading, e.g. "Wh"
value	*UInt48*	The value of the summary measurement.

TOUType [TOU A, B, etc… represent any period. The Third Party should have a specific mapping as to what each Tier means. A mapping spec. needs to be defined.]

The only valid values are:

0 = NotApplicable
1 = TOU A
2 = TOU B
3 = TOU C
4 = TOU D
5 = TOU E
6 = TOU F
7 = TOU G
8 = TOU H
9 = TOU I
10 = TOU J
11 = TOU K
12 = TOU L
13 = TOU M
14 = TOU N
15 = TOU O

TimeAttributeType

The only valid values are:

0 = Not Applicable
1 = 10-minute
2 = 15-minute

4 = 24-hour
5 = 30-minute
7 = 60-minute
11 = Daily
13 = Monthly
15 = Present
16 = Previous
24 = Weekly
32 = ForTheSpecifiedPeriod
79 = Daily30minuteFixedBlock

TimeType «XSDsimpleType»

Time is a signed 64 bit value representing the number of seconds since 0 hours, 0 minutes, 0 seconds, on the 1st of January, 1970.

UomType

The only valid values are:

0 = Not Applicable
5 = A (Current)
29 = Voltage
31 = J (Energy joule)
33 = Hz (Frequency)
38 = Real power (Watts)
42 = m3 (Cubic Meter)
61 = VA (Apparent power)
63 = VAr (Reactive power)
65 = Cos? (Power factor)
67 = V² (Volts squared)
69 = A² (Amp squared)
71 = VAh (Apparent energy)
72 = Real energy (Watt-hours) [Include kWh or kVAh.]
73 = VArh (Reactive energy)
106 = Ah (Ampere-hours / Available Charge)
119 = ft3 (Cubic Feet)
122 = ft3/h (Cubic Feet per Hour)
125 = m3/h (Cubic Meter per Hour)
128 = US gl (US Gallons)
129 = US gl/h (US Gallons per Hour)
130 = IMP gl (Imperial Gallons)
131 = IMP gl/h (Imperial Gallons per Hour)
132 = BTU
133 = BTU/h
134 = Liter
137 = L/h (Liters per Hour)
140 = PA(gauge)
155 = PA(absolute)
169 = Therm

IntervalBlock

Time sequence of Readings of the same ReadingType.

Name	Type	Description
interval	*DateTimeInterval*	Specifies the time period during which the contained readings were taken.

IntervalReading

Specific value measured by a meter or other asset. Each Reading is associated with a specific ReadingType.

Name	Type	Description
cost [This must be optional as it is difficult today to derive.]	*UInt48*	Specifies a cost associated with this reading, in millionths of the currency specified in the ReadingType for this reading. (e.g. 840 = USD, US dollar)
timePeriod	*DateTimeInterval*	The date time and duration of a reading. If not specified, readings for each "intervalLength" in ReadingType are present.
value	*UInt48*	Value in units specified by ReadingType

MeterReading

Set of values obtained from the meter.

ReadingQuality

Quality of a specific reading value or interval reading value. Note that more than one Quality may be applicable to a given Reading. Typically not used unless problems or unusual conditions occur (i.e., quality for each Reading is assumed to be 'Good' (valid) unless stated otherwise in associated ReadingQuality).

Name	Type	Description
quality	*QualityOfReading*	Quality, to be specified if different than ReadingType.defaultQuality. The specific format is specified per the standard is defined in QualityOfReading.

ReadingType

Characteristics associated with all Readings included in a MeterReading.

Name	Type	Description
accumulationBehaviour	*AccumulationBehaviourType*	Code indicating how value is accumulated over time for Readings of ReadingType. The list of valid values per the standard are defined in AccumulationBehaviorType.

Name	Type	Description
		Examples are: 0 = Not Applicable 1 = BulkQuantity 3 = Cumulative
commodity	*CommodityType*	Code for commodity classification of Readings of ReadingType. The valid values per the standard are defined in CommodityType. Examples are: 0 = Not Applicable 1 = Electricity secondary metered value (a premise meter is typically a secondary meter) 2 = Electricity primary metered value 4 = Air 7 = NaturalGas
consumptionTier	*ConsumptionTierType*	Code for consumption tier associated with a Reading of ReadingType. The valid values are define in ConsumptionTierType. Examples are: 0 = Not Applicable 1 = Block Tier 1 2 = Block Tier 2
currency	*CurrencyCode*	Code for the currency for costs associated with this ReadingType. The valid values per the standard are defined in CurrencyCode. Examples are: 0 - Not Applicable 36 - Australian Dollar 124 - Canadian Dollar 840 - US Dollar 978 - Euro
dataQualifier	*DataQualifierType*	Code describing a salient attribute of Readings of ReadingType. Valid values per the standard are defined in DataQualifierType. Examples are: 0 = Not Applicable 2 = Average
defaultQuality	*QualityOfReading*	Default value to be used if no value of ReadingQuality.quality is provided.

Name	Type	Description
		Specific format and valid values per the standard are specified in QualityOfReading.
flowDirection	*FlowDirectionType*	Direction associated with current related Readings. valid values per the standard are defined in FlowDirectionType. Examples are: 0 = Not Applicable 1 = Forward 19 = Reverse
kind	*KindType*	Code for general classification of a Reading of ReadingType. Valid values per the standard are defined in KindType. Examples are: 0 = Not Applicable 3 = Currency 8 = Demand
intervalLength	*UInt32*	Default interval length specified in seconds for Readings of ReadingType.
phase	*PhaseCode*	Code for phase information associated with Readings of ReadingType. Valid values per the standard are defined in PhaseCode. Examples are: 0 = Not Applicable 129 = Phase AN 128 = Phase A 132 = Phase AB
powerOfTenMultiplier	*PowerOfTenMultiplierType*	Code for the power of ten multiplier which, when used in combination with the uom, specifies the actual unit of measure for Readings of ReadingType. Valid values per the standard are defined in PowerOfTenMultiplierType. Examples are: 0 = None 1 = deca=x10 2 = hecto=x100 -3 = mili=x10-3
timeAttribute	*TimeAttributeType*	Code used to specify a particular type of time interval method for Readings of ReadingType. Valid values per the standard

Name	Type	Description
		are defined in TimeAttributeType. Examplesare: 0 = Not Applicable 1 = 10-minute 2 = 15-minute
tou	*TOUType*	Code for the TOU type of Readings of ReadingType. valid values per the standard are defined in TOUType. Examples are: 0 = NotApplicable 1 = TOU A 2 = TOU B
uom	*UomType*	Code for the base unit of measure for Readings of ReadingType. Used in combination with the powerOfTenMultiplier to specify the actual unit of measure. Valid values per the standard are defined in UomType. Examples are: 0 = Not Applicable 5 = A (Current) 29 = Voltage

REQ.21.4.2 Additional Models in Support of Services

REQ.21.4.2.2 Authorization

An ~~A~~authorization is a Customer grant of Authorized Third Party access to specific resources. The attributes of this object are listed below. The structure and format of these fields ~~is~~are defined by OAuth.

- Information consumer (Third Party) identifier (Consumer Key)
- Resource subject (Scope)
  Specifies the resource to which access from the information provider is granted by the customer to the information consumer
- Authentication Token (Access Token)
  One or more token / secret pairs proving the identity of the requester to be the Customer associated with the resource. Different methods may be defined against which token / secret pairs are created and verified.
- Authorized period
- Status (requested, valid, invalid, error, unavailable)

REQ.21.5 Related Model Business Practices

REQ.21.5.1 Conformance to REQ.18 Energy Usage Information Model

ESPI services conform to REQ.18 Energy Usage Information Model (PAP10), due to the ability to directly transform between the models without loss of precision, and without external inputs. The ESPI model uses the same concept names and structures as NAESB PAP10 EUI.

REQ.21.6 Technical Implementation

REQ.21.6.1 Protocol Specifications

ESPI endpoints implement the following protocol aspects.

REQ.21.6.1.1 Security

Providers of ESPI services, including both ~~d~~Data ~~c~~Custodian and ~~t~~Third ~~p~~Party, protect their systems, networks, and interface endpoints against threats, as recommended in NIST-IR and Security Profile For Third Party Data Access.

REQ.21.6.1.1.1 Encryption

Establishment of mutually-authenticated encrypted channels is performed using HTTP/S [HTTPS over WS rest Only using syndication? What about SFTP? If we do WS, this results in large data flowing. We will need to include SLAs.], as documented in IETF RFC 2818, over which ~~data~~EUI may be securely transferred between Data Custodian and Authorized Third Party.

REQ.21.6.1.1.2 User Authorization

OAuth, as documented in IETF RFC 5849, is used for authorization grant and access by Retail Customers and Authorized Third Parties to shared Data Custodian resources. This protocol results in access tokens that are used to subscribe to specific user ~~data~~EUI, or to request it immediately, if supported.

REQ.21.6.2 Communication Specifications

This section defines the expected behavior of implementations using the ESPI RESTful style. This style uses HTTP methods as verbs and URIs as nouns.

During initial configuration, the ~~d~~Data ~~c~~Custodian issues the Authorized Third Party an ID~~d~~ (key) and secret required by RFC 5849. If supported, ESPI service providers may make available an "ApplicationInformation" feed and allow applications to be created, updated, and deleted using AtomPub methods. They may require trusted credentials for access.

ESPI endpoints expose resources as described by Atom, IETF RFC 4287.

- Representations are identified as media type "`application/atom+xml`"
- ESPI namespace and types ("`http://naesb.org/espi`") are used for objects in `<content>` element
- `espi:mRID` is implemented by `atom:id`
  - UUIDs are used, as specified in IETF RFC 4122
- `espi:description` is implemented by `atom:title`
- `atom:published` and `atom:updated` are used
- Associated objects use `atom:link` (`rel="related"`)

The following addressable objects (specializations of IdentifiedObject) are defined by the ESPI schema, and can be made available using AtomPub feeds. [ATOMPub is an HTTP syndication protocol. It would be better to simply leverage basic schemas.]

- UsagePoint
- ReadingType
- IntervalBlock
- MeterReading
- Subscription
- ElectricPowerUsageSummary
- ElectricPowerQualitySummary
- Authorization
- ApplicationInformation

Links shall use the following tags and values to convey link types.

Association	**rel**	**type**
UsagePoint → MeterReading	`related`	`application/atom+xml`
UsagePoint → ElectricPowerQualitySummary	`related`	`application/atom+xml`
UsagePoint → ElectricPowerUsageSummary	`related`	`application/atom+xml`
MeterReading → IntervalBlock	`related`	`application/atom+xml`
MeterReading → ReadingType	`related`	`application/atom+xml`

ESPI endpoints use HTTP and/or HTTPS, IETF RFC 2616 and 2818, to expose ESPI resources using the method conventions in Atom Publishing Protocol, IETF RFC 5023.

URIs are kept as short as possible, and do not exceed 255 bytes.

Relative URIs may be used when resources are on the same host. Additional definition regarding URIs and HTTP/S follow the IETF specifications.

A URI example is provided below.

- `https://espi.datacustodian.com/{third_party_id}/Batch`

Since all URIs are opaque references, there is no mandated form. However, it may be useful to organize them hierarchically, in order to define URIs for the appropriate containers (feeds), and to manage permissions. URIs should be as persistent as possible, but they may change. `atom:id`, however, does not change, even if the resource is moved or replicated. Clients accessing out-of-date URIs may be redirected, but if they are not, may need to request the current preferred resource location.

The following query parameters are used to filter the resources returned by a feed. These use typical "`?name=value[&…]`" syntax.

- `published-max, published-min`
- `updated-max, updated-min`
- `max-results`
- `start-index`

Date and time values for the above parameters use RFC 3339 format.

REQ.21.6.3 Examples

The following examples show the creation, retrieval, update, and deletion of an object within a feed.

Upon authorization of a resource, the OAuth "scope" attribute contains the URI of the resource. With it, the client can request a subscription to it as in the example below. Note that this URI is a reference to the target resource of the subscription. (i.e., The resource being subscribed to). The Subscription object identifier is not specified, since it is assigned by the Data Custodian. Note also that signed OAuth parameters are included in the header to prove authorization.

```
POST /Subscription HTTP/1.1
Host: espi.datacustodian.com
Content-Type: application/atom+xml
Content-Length: 163

<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
    xsi:schemaLocation="http://naesb.org/espi espi.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <link rel="related" href="/User/9b6c7063"/>
  <content>
    <Subscription xmlns="http://naesb.org/espi"/>
  </content>
</entry>
```

The server may refuse this request, if valid authorization was not provided, in which case result "`401 Not Authorized`" or similar, is returned.

If the request is accepted, the server responds with the full resource representation, as in the response example below.

```
HTTP/1.1 200 OK
Content-Type: application/atom+xml
Content-Length: 335

<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
    xsi:schemaLocation="http://naesb.org/espi espi.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <id>urn:uuid:e69c4c25-2885-4de0-a3d8-d29b5f823b79</id>
  <link rel="self" href="/Subscription/7f23"/>
  <link rel="related" href="/User/9b6c7063"/>
  <content>
    <Subscription xmlns="http://naesb.org/espi"/>
  </content>
  <published>2012-01-23T05:11:38Z</published>
  <updated>2012-01-23T05:11:38Z</updated>
</entry>
```

Retrieval of the object is performed using GET. The example below shows the request – the response is the same as the response to POST above.

```
GET /Subscription/7f23 HTTP/1.1
Host: espi.datacustodian.com
```

Deletion uses DELETE, as in the example below. Again, authorization parameters are included.

```
DELETE /Subscription/7f23 HTTP/1.1
Host: espi.datacustodian.com
```

Response is simply the status of the request, as below.

```
HTTP/1.1 200 OK
```

Batch processing involves inclusion of the "`Batch`" attributes with regular objects in a list, as in the example below. This example shows delivery of new objects.

```
<?xml version="1.0" encoding="UTF-8"?>
<feed
    xmlns="http://www.w3.org/2005/Atom"
    xsi:schemaLocation="http://naesb.org/espi espi.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <id>urn:uuid:046638c0-8701-11e0-9d78-0800200c9a66</id>
  <title>ThirdPartyX Batch Feed</title>
  <updated>2012-05-21T18:01:00Z</updated>
```

```
  <link rel="self" href="/83e269c1"/>
  <entry>
    <id>urn:uuid:c990b150-8320-11e0-9d78-0800200c9a66</id>
    <link rel="self" href="/ThirdParty/83e269c1/Batch"/>
  <entry>
    <id>urn:uuid:c990b150-8320-11e0-9d78-0800200c9a66</id>
    <link rel="self" href="/User/9b6c7063/UsagePoint/01"/>
    <link rel="up" href="/User/9b6c7063/UsagePoint"/>
    <link rel="related" href="/User/9b6c7063/UsagePoint/01/MeterReading"/>
    <title>Elm St.</title>
    <content>
      <UsagePoint xmlns="http://naesb.org/espi">
        <ServiceCategory>
          <kind>0</kind>
        </ServiceCategory>
      </UsagePoint>
    </content>
    <published>2012-05-21T18:01:00Z</published>
    <updated>2012-05-21T18:01:00Z</updated>
  </entry>
  <entry>
    <id>urn:uuid:f2034e91-8320-11e0-9d78-0800200c9a66</id>
    <link rel="self" href="/User/9b6c7063/UsagePoint/01/MeterReading/01"/>
    <link rel="up" href="/User/9b6c7063/UsagePoint/01/MeterReading"/>
    <link rel="related" href="/User/9b6c7063/UsagePoint/01/MeterReading/01/IntervalBlock"/>
    <link rel="related" href="/ReadingType/07"/>
    <title>Hourly Energy Consumption</title>
    <content>
      <MeterReading xmlns="http://naesb.org/espi"/>
    </content>
    <published>2012-05-21T18:01:00Z</published>
    <updated>2012-05-21T18:01:00Z</updated>
  </entry>
  <entry>
    <id>urn:uuid:f2034e93-8320-11e0-9d78-0800200c9a66</id>
    <link rel="self"
href="/User/9b6c7063/UsagePoint/01/MeterReading/01/IntervalBlock/0173"/>
    <link rel="up" href="/User/9b6c7063/UsagePoint/01/MeterReading/01/IntervalBlock"/>
    <title/>
    <content>
      <IntervalBlock xmlns="http://naesb.org/espi">
        <interval>
          <duration>86400</duration>
          <start>1325397600</start>
        </interval>
        <IntervalReading>
          <cost>3000000</cost>
          <timePeriod>
            <duration>3600</duration>
            <start>1325397600</start>
          </timePeriod>
          <value>383</value>
        </IntervalReading>
        <IntervalReading>
          <cost>3000000</cost>
          <timePeriod>
            <duration>3600</duration>
            <start>1325401200</start>
          </timePeriod>
          <value>427</value>
        </IntervalReading>
      </IntervalBlock>
    </content>
    <published>2012-05-21T18:01:00Z</published>
    <updated>2012-05-21T18:01:00Z</updated>
  </entry>
  <entry>
    <id>urn:uuid:2557def0-8321-11e0-9d78-0800200c9a66</id>
    <link rel="self" href="/ReadingType/07"/>
    <link rel="up" href="/ReadingType"/>
    <title>Energy Delivered (kWh)</title>
    <content>
      <ReadingType xmlns="http://naesb.org/espi">
        <accumulationBehaviour>4</accumulationBehaviour><!--DeltaData-->
        <commodity>1</commodity><!--Electricity-->
        <consumptionTier>0</consumptionTier><!--N/A-->
```

```
        <currency>0</currency><!--N/A-->
        <dataQualifier>12</dataQualifier><!--Normal-->
        <flowDirection>1</flowDirection><!--Forward-->
        <kind>12</kind><!--Energy-->
        <phase>0</phase><!--N/A-->
        <powerOfTenMultiplier>3</powerOfTenMultiplier><!--kilo-->
        <timeAttribute>0</timeAttribute><!--N/A-->
        <tou>0</tou><!--N/A-->
        <uom>72</uom><!--Watt hours-->
      </ReadingType>
    </content>
    <published>2012-05-21T18:01:00Z</published>
    <updated>2012-05-21T18:01:00Z</updated>
  </entry>
</feed>
```

REQ.21.6.4 Conformance

Conformant Data Custodian implementations include the following:

- Subscriptions
  - Accept POST to Subscription, Batch resource
  - Allow subscriptions to authorized resources
- Delivery
  - Accept GET to Batch resource, specific to each Authorized Third Party. [Please better explain the options for delivery by defining conformance in terms of the Use Cases and tie the abstract services to the Use Cases.]
  - Optionally support POST to Authorized Third Party Notification resource
  - Optionally support POST to Authorized Third Party Batch resource
  - Optionally support GET of resources directly

Conformant Third Party implementations include the following:

- Security
  - Server certificates and mutually authenticated HTTPS
  - Make requests to OAuth endpoints
- Subscriptions
  - Submit POST requests to Subscription, Batch resource
  - Sign requests with access tokens
- Delivery
  - Submit GET request to Batch resource
  - Optionally accept POST to Authorized Third Party Notification resource
  - Optionally accept POST to Authorized Third Party Batch resource
  - Optionally GET resources directly

All conformant implementations include the following:

- Security
  - Server certificates and mutually authenticated HTTPS
  - Accept requests to OAuth endpoints

- Content
  - Information elements with the meaning defined herein use the format and structure defined herein.
  - Additional information elements not defined herein are placed in extension elements as defined by the ESPI schema herein, use a namespace different from the ESPI schema herein, and are optional.
  - It is recommended that any additional information elements included in an implementation be submitted for consideration in future versions of ESPI.

REQ.21.6.5 XML Schema

The following is the XML Schema (espi.xsd) definition used to declare the format of the ESPI types. This is the official normative version of these definitions. A separate version of this file, in plain text, appropriate for machine reading, can be obtained through the NAESB office.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://naesb.org/espi"
targetNamespace="http://naesb.org/espi" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="0.03">
  <xs:import namespace="http://www.w3.org/2005/Atom" schemaLocation="atom.xsd"/>
  <xs:complexType name="ApplicationInformation">
    <xs:annotation>
      <xs:documentation>Contains information about an Authorized Third Party Application
requesting access to the DataCustodian services. Information requested may include items such
as Organization Name, Website, Contact Info, Application Name, Description, Icon, Type,
default Notification and Callback endpoints, and may also include agreement with terms of
service.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="dataCustodianApplicationStatus" type="UInt8" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A code indicating the current status of the application.
(Provided by dataCustodian, cannot be modified)
Defined statuses are:
1 - Review
2 - Production (Live)
3 - On hold
4 - Revoked</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="dataCustodianDefaultBatchResource" type="xs:anyURI" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The default endpoint for Batch requests. (Provided by
dataCustodian, updated in approved applications objects, cannot be modified by authorized
third party)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="dataCustodianDefaultSubscriptionResource" type="xs:anyURI"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The default endpoint for Subscription requests. (Provided by
dataCustodian, updated in approved applications objects, cannot be modified by authorized
third party)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="dataCustodianThirdPartyId" type="String32" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A key to be associated with this application, to be provided
in OAuth requests. (Provided by dataCustodian, cannot be modified)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="dataCustodianThirdPartySecret" type="String32" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A secret to be associated with this application, used to sign
OAuth requests. (Provided by dataCustodian, cannot be modified)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationDescription" type="xs:string" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A description of the application.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationLogo" type="xs:anyURI" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The link to the logo image for the application. Size greater
than 180 x 150 may be cropped or reduced.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationName" type="String32" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The name of the application to which access will be
granted.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationStatus" type="UInt8" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A code indicating the current status of the application.
Defined statuses are:
1 - Development
2 - Production (Live)
3 - Retired (Remove)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationType" type="UInt8" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A code indicating the type of the application.
Defined types are:
1 - Web Application
2 - Desktop Application
3 - Mobile Application</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationUse" type="UInt8" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A code indicating the expected use of the application.
Defined uses are:
1 - Energy management
2 - Comparisons
3 - Government</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyApplicationWebsite" type="xs:anyURI" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The link to the main page of the
application.</xs:documentation>
            </xs:annotation>
```

```
          </xs:element>
          <xs:element name="thirdPartyDefaultBatchResource" type="xs:anyURI" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The default endpoint for asynchronous delivery of Batch data
using push.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyDefaultNotifyResource" type="xs:anyURI" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The default endpoint for authorized third party notification
of Batch data availability, that is then requested from the data custodian via the Batch
resource.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyDefaultOAuthCallback" type="xs:anyURI" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The default redirect back to the application after
authorization grant.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyEmail" type="String32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The e-mail address of the organization to which access will be
granted. (For debugging - not to be shared with customers)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyName" type="String32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The name of the organization to which access will be
granted.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyPhone" type="String32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The phone number of the organization to which access will be
granted. (For debugging - not to be shared with customers)</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="Authorization">
    <xs:annotation>
      <xs:documentation>Represents a permission granted by an owner for access to a
resource.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="accessToken" type="String32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Contains the access token associated with this
authorization.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="authorizationServer" type="xs:anyURI" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Contains the URI link to the authorization endpoint associated
with this authorization.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="authorizedPeriod" type="DateTimeInterval" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Restricts access to requests or subscriptions within this date
time interval.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="publishedPeriod" type="DateTimeInterval" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Restricts access to only the objects within the associated
resource that were published within this date time interval.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="resource" type="xs:anyURI" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Contains the identifier of the resource, same as was specified
in OAuth "scope".</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="status" type="UInt8" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The status of this authorization.
0 - Revoked
1 - Active</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="thirdPartyConsumer" type="String32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Contains the identifier for the Authorized Third
Party.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="IntervalBlock">
    <xs:annotation>
      <xs:documentation>Time sequence of Readings of the same ReadingType.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="interval" type="DateTimeInterval" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Specifies the time period during which the contained readings
were taken.</xs:documentation>
            </xs:annotation>
```

```
          </xs:element>
          <xs:element name="IntervalReading" type="IntervalReading" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="IntervalReading">
    <xs:annotation>
      <xs:documentation>Specific value measured by a meter or other asset. Each Reading is
associated with a specific ReadingType.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="cost" type="UInt48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Specifies a cost associated with this reading, in millionths
of the currency specified in the ReadingType for this reading.  (e.g. 840 = USD, US
dollar)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="ReadingQuality" type="ReadingQuality" minOccurs="0"
maxOccurs="unbounded"/>
          <xs:element name="timePeriod" type="DateTimeInterval" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The date time and duration of a reading. If not specified,
readings for each "intervalLength" in ReadingType are present.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="value" type="UInt48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Value in units specified by ReadingType</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="MeterReading">
    <xs:annotation>
      <xs:documentation>Set of values obtained from the meter.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ReadingQuality">
    <xs:annotation>
      <xs:documentation>Quality of a specific reading value or interval reading value. Note
that more than one Quality may be applicable to a given Reading. Typically not used unless
problems or unusual conditions occur (i.e., quality for each Reading is assumed to be 'Good'
(valid) unless stated otherwise in associated ReadingQuality).</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="quality" type="QualityOfReading" minOccurs="1" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Quality, to be specified if different
thanReadingType.defaultQuality.

The specific format  is specified per the standard is defined in
QualityOfReading.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ReadingType">
    <xs:annotation>
      <xs:documentation>Characteristics associated with all Readings included in a
MeterReading.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="accumulationBehaviour" type="AccumulationBehaviourType"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code indicating how value is accumulated over time for
Readings of ReadingType. The list of valid values per the standard are defined in
AccumulationBehaviorType.

Examples are:
0 = Not Applicable
1 = BulkQuantity
3 = Cumulative</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="commodity" type="CommodityType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for commodity classification of Readings of ReadingType.
The valid values per the standard are defined in CommodityType.

Examples are:
0 = Not Applicable
1 = Electricity secondary metered value (a premise meter is typically a secondary meter)
2 = Electricity primary metered value
4 = Air
7 = NaturalGas</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="consumptionTier" type="ConsumptionTierType" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for consumption tier associated with a Reading of
ReadingType. The valid values are define in ConsumptionTierType.

Examples are:
0 = Not Applicable
1 = Block Tier 1
2 = Block Tier 2</xs:documentation>
```

```
            </xs:annotation>
          </xs:element>
          <xs:element name="currency" type="CurrencyCode" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for the currency for costs associated with this
ReadingType.  The valid values per the standard are defined in CurrencyCode.

Examples are:
0 - Not Applicable
36 - Australian Dollar
124 - Canadian Dollar
840 - US Dollar
978 - Euro</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="dataQualifier" type="DataQualifierType" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code describing a salient attribute of Readings of
ReadingType. Valid values per the standard are defined in DataQualifierType.

Examples are:
0 = Not Applicable
2 = Average</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="defaultQuality" type="QualityOfReading" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Default value to be used if no value of ReadingQuality.quality
is provided.

Specific format and valid values per the standard are specified in
QualityOfReading.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="flowDirection" type="FlowDirectionType" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Direction associated with current related Readings. valid
values per the standard are defined in FlowDirectionType.

Examples are:
0 = Not Applicable
1 = Forward
19 = Reverse</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="intervalLength" type="UInt32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Default interval length specified in seconds for Readings of
ReadingType.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="kind" type="KindType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for general classification of a Reading of ReadingType.
Valid values per the standard are defined in KindType.

Examples are:

0 = Not Applicable
3 = Currency
8 = Demand</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="phase" type="PhaseCode" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for phase information associated with Readings of
ReadingType. Valid values per the standard are defined in PhaseCode.

Examples are:
0 = Not Applicable
129 = Phase AN
128 = Phase A
132 = Phase AB</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="powerOfTenMultiplier" type="PowerOfTenMultiplierType"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for the power of ten multiplier which, when used in
combination with the uom, specifies the actual unit of measure for Readings of ReadingType.
Valid values per the standard are defined in PowerOfTenMultiplierType.

Examples are:
0 = None
1 = deca=x10
2 = hecto=x100
-3 = mili=x10-3</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="timeAttribute" type="TimeAttributeType" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code used to specify a particular type of time interval method
for Readings of ReadingType. Valid values per the standard are defined in TimeAttributeType.

Examplesare:
0 = Not Applicable
1 = 10-minute
2 = 15-minute</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="tou" type="TOUType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for the TOU type of Readings of ReadingType. valid values
per the standard are defined in TOUType.
Examples are:
0 = NotApplicable
1 = TOU A
2 = TOU B</xs:documentation>
            </xs:annotation>
```

```
          </xs:element>
          <xs:element name="uom" type="UomType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Code for the base unit of measure for Readings of ReadingType.
Used in combination with the powerOfTenMultiplier to specify the actual unit of measure. Valid
values per the standard are defined in UomType.

Examples are:
0 = Not Applicable
5 = A (Current)
29 = Voltage</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="IdentifiedObject">
    <xs:annotation>
      <xs:documentation>This is a root class to provide common naming attributes for all
classes needing naming attributes</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="BatchItemInfo" type="BatchItemInfo" minOccurs="0" maxOccurs="1"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ServiceCategory">
    <xs:annotation>
      <xs:documentation>Category of service provided to the customer.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="kind" type="ServiceKind" minOccurs="1" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Service classification
Examples are:
0 - electricity
1 - gas

The list of specific valid values per the standard are itemized in
ServiceKind.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="UsagePoint">
    <xs:annotation>
      <xs:documentation>Logical point on a network at which consumption or production is
either physically measured (e.g. metered) or estimated (e.g. unmetered street
lights).</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="roleFlags" type="HexBinary16" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Specifies the roles that this usage point has been assigned.
Bit 1 - isMirror
Bit 2 - isPremisesAggregationPoint
Bit 3 - isPEV
Bit 4 - isDER
Bit 5 - isRevenueQuality
Bit 6 - isDC
Bit 7-16 - Reserved</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="ServiceCategory" type="ServiceCategory" minOccurs="0"
maxOccurs="1"/>
          <xs:element name="status" type="UInt8" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Specifies the current status of this usage point.

The only valid values are:

0 = off
1 = on</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ElectricPowerQualitySummary">
    <xs:annotation>
      <xs:documentation>A summary of power quality events. This information represents a
summary of power quality information typically required by customer facility energy management
systems. It is not intended to satisfy the detailed requirements of power quality monitoring.
All values are as defined by measurementProtocol during the period. The standards typically
also give ranges of allowed values; the information attributes are the raw measurements, not
the "yes/no" determination by the various standards. See referenced standards for definition,
measurement protocol  and period.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="flickerPlt" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A measurement of long term Rapid Voltage Change in hundredths.

flickerPlt is derived from 2 hours of Pst values (12 values combined in cubic
relationship).</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="flickerPst" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>flickerPst is a value measured over 10 minutes that
characterizes the likelihood that the voltage fluctuations would result in perceptible light
```

```
flicker. A value of 1.0 is designed to represent the level that 50% of people would perceive
flicker in a 60 watt incandescent bulb.

The value reported is represented as an integer in hundredths.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="harmonicVoltage" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A measurement of the Harmonic Voltage during the period. For
DC, distortion is with respect to a signal of zero Hz.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="longInterruptions" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Long Interruption events (as defined by
measurementProtocol) during the summary interval period.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="mainsVoltage" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A measurement of the Mains [Signaling] Voltage during the
summary interval period in uV.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="measurementProtocol" type="UInt8" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A reference to the source standard used as the measurement
protocol definition.

Examples are:
0 = "IEEE1519-2009"
1 = "EN50160"</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="powerFrequency" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A measurement of the power frequency during the summary
interval period in uHz.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="rapidVoltageChanges" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Rapid Voltage Change events during the summary
interval period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="shortInterruptions" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Short Interruption events during the summary
interval period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="summaryInterval" type="DateTimeInterval" minOccurs="1"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Interval of summary period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="supplyVoltageDips" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Supply Voltage Dip events during the summary
interval period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="supplyVoltageImbalance" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Supply Voltage Imbalance events during the summary
interval period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="supplyVoltageVariations" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Supply Voltage Variations during the summary
interval period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="tempOvervoltage" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>A count of Temporary Overvoltage events (as defined by
measurementProtocol) during the summary interval period</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ElectricPowerUsageSummary">
    <xs:annotation>
      <xs:documentation>Summary of usage for a billing period</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="IdentifiedObject">
        <xs:sequence>
          <xs:element name="billingPeriod" type="DateTimeInterval" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The billing period to which the included measurements
apply</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="billLastPeriod" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The amount of the bill for the previous period , in millionths
of the currency specified in the ReadingType for this reading (e.g. 840 = USD, US
dollar).</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="billToDate" type="Int48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The bill amount related to the billing period as of the date
received, in millionths of the currency specified in the ReadingType for this reading. (e.g.
840 = USD, US dollar).</xs:documentation>
            </xs:annotation>
          </xs:element>
```

```
          <xs:element name="costAdditionalLastPeriod" type="Int48" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Additional charges from the last billing period, in millionths
of the currency specified in the ReadingType for this reading.  (e.g. 840 = USD, US
dollar).</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="currency" type="CurrencyCode" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The ISO 4217 code indicating the currency applicable to the
bill amounts in the summary. See list at
http://www.unece.org/cefact/recommendations/rec09/rec09_ecetrd203.pdf</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="currentBillingPeriodOverAllConsumption" type="SummaryMeasurement"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The total consumption for the billing
period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="currentDayLastYearNetConsumption" type="SummaryMeasurement"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The amount of energy consumed one year ago interpreted as same
day of week same week of year (see ISO 8601).</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="currentDayNetConsumption" type="SummaryMeasurement" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Net consumption for the current day (delivered -
received)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="currentDayOverallConsumption" type="SummaryMeasurement"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Overall energy consumption for the current
day</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="peakDemand" type="SummaryMeasurement" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Peak demand recorded for the current period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="previousDayLastYearOverallConsumption" type="SummaryMeasurement"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The amount of energy consumed on the previous day one year ago
interpreted as same day of week same week of year  (see ISO 8601).</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="previousDayNetConsumption" type="SummaryMeasurement" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Net consumption for the previous day</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="previousDayOverallConsumption" type="SummaryMeasurement"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The total consumption for the previous day</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="qualityOfReading" type="QualityOfReading" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Indication of the quality of the summary
readings</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="ratchetDemand" type="SummaryMeasurement" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The current ratchet demand value for the ratchet demand
period</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="ratchetDemandPeriod" type="DateTimeInterval" minOccurs="0"
maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The period over which the ratchet demand
applies</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="statusTimeStamp" type="TimeType" minOccurs="1" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Date/Time status of this UsageSummary</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:simpleType name="HexBinary128">
    <xs:annotation>
      <xs:documentation>A 128-bit field encoded as a hex string (32 characters / 16
octets)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:hexBinary">
      <xs:maxLength value="16"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="HexBinary16">
    <xs:annotation>
      <xs:documentation>A 16-bit field encoded as a hex string (4 characters / 2
octets)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:hexBinary">
      <xs:maxLength value="2"/>
    </xs:restriction>
```

```
  </xs:simpleType>
  <xs:simpleType name="String32">
    <xs:annotation>
      <xs:documentation>Character string of max length 32</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="32"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="UInt16">
    <xs:annotation>
      <xs:documentation>Unsigned integer, max inclusive 65535 (2^16-1), same as
xs:unsignedShort</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:unsignedShort"/>
  </xs:simpleType>
  <xs:simpleType name="UInt32">
    <xs:annotation>
      <xs:documentation>Unsigned integer, max inclusive 4294967295 (2^32-1), same as
xs:unsignedInt</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:unsignedInt"/>
  </xs:simpleType>
  <xs:simpleType name="UInt48">
    <xs:annotation>
      <xs:documentation>Unsigned integer, max inclusive 281474976710655 (2^48-1), restriction
of xs:unsignedLong</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:unsignedLong">
      <xs:maxInclusive value="281474976710655"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="UInt8">
    <xs:annotation>
      <xs:documentation>Unsigned integer, max inclusive 255 (2^8-1), same as
xs:unsignedByte</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:unsignedByte"/>
  </xs:simpleType>
  <xs:simpleType name="Int48">
    <xs:annotation>
      <xs:documentation>Signed integer, max inclusive 281474976710655 (2^48-1), restriction of
xs:long</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:long">
      <xs:maxInclusive value="281474976710655"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="AccumulationBehaviourType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
1 = BulkQuantity
3 = Cumulative
4 = DeltaData
6 = Indicating
9 = Summation
12 = Instantaneous</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="CommodityType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
1 = Electricity secondary metered value (a premise meter is typically a secondary meter)
2 = Electricity primary metered value
4 = Air
7 = NaturalGas
8 = Propane
9 = PotableWater
10 = Steam
11 = WasteWater
12 = HeatingFluid
13 = CoolingFluid</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="ConsumptionTierType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
1 = Block Tier 1
2 = Block Tier 2
3 = Block Tier 3
4 = Block Tier 4
5 = Block Tier 5
6 = Block Tier 6
7 = Block Tier 7
8 = Block Tier 8
9 = Block Tier 9
10 = Block Tier 10
11 = Block Tier 11
12 = Block Tier 12
13 = Block Tier 13
14 = Block Tier 14
15 = Block Tier 15
16 = Block Tier 16</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="CurrencyCode">
    <xs:annotation>
      <xs:documentation>Follows codes defined in ISO 4217. Full list at tiny.cc/4217.
```

```
0 - Not Applicable
36 - Australian Dollar
124 - Canadian Dollar
840 - US Dollar
978 - Euro</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="DataQualifierType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
2 = Average
8 = Maximum
9 = Minimum
12 = Normal</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="DateTimeInterval">
    <xs:annotation>
      <xs:documentation>Interval of date and time. End is not included because it can be
derived from the start and the duration.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="duration" type="UInt32" minOccurs="1" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Duration of the interval, in seconds.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="start" type="TimeType" minOccurs="1" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Date and time that this interval started.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="FlowDirectionType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
1 = Forward
19 = Reverse</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="KindType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
3 = Currency
8 = Demand
12 = Energy
37 = Power</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="PhaseCode">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
129 = Phase AN
128 = Phase A
132 = Phase AB
64 = Phase BN
64 = Phase B
32 = Phase CN
32 = Phase C
224 = Phase ABC
66 = Phase BC
40 = Phase CA
512 = Phase S1
256 = Phase S2
768 = Phase S1S2
513 = Phase S1N
257 = Phase S2N
769 = Phase S1S2N</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt16"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:simpleType name="PowerOfTenMultiplierType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = None
1 = deca=x10
2 = hecto=x100
-3 = mili=x10-3
3 = kilo=x1000
6 = Mega=x106
-6 = micro=x10-3
9 = Giga=x109</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:byte"/>
  </xs:simpleType>
  <xs:complexType name="QualityOfReading">
```

```
    <xs:annotation>
      <xs:documentation>List of codes indicating the quality of the reading, using
specification:

0 – valid (validated)
7 - manually edited
8 - estimated
10 - questionable
11 - derived
12 – projected (forecasted)
13 - mixed
14 - raw
15 - normalized for weather
16 - other </xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="ServiceKind">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 - electricity
1 - gas
2 - water
4 - pressure
5 - heat
6 - cold
7 - communication
8 - time</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="TimeAttributeType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
1 = 10-minute
2 = 15-minute
4 = 24-hour
5 = 30-minute
7 = 60-minute
11 = Daily
13 = Monthly
15 = Present
16 = Previous
24 = Weekly
32 = ForTheSpecifiedPeriod
79 = Daily30minuteFixedBlock</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:simpleType name="TimeType">
    <xs:annotation>
      <xs:documentation>Time is a signed 64 bit value representing the number of seconds since
0 hours, 0 minutes, 0 seconds, on the 1st of January, 1970.</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:long"/>
  </xs:simpleType>
  <xs:complexType name="TOUType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = NotApplicable
1 = TOU A
2 = TOU B
3 = TOU C
4 = TOU D
5 = TOU E
6 = TOU F
7 = TOU G
8 = TOU H
9 = TOU I
10 = TOU J
11 = TOU K
12 = TOU L
13 = TOU M
14 = TOU N
15 = TOU O</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="UomType">
    <xs:annotation>
      <xs:documentation>The only valid values are:

0 = Not Applicable
5 = A (Current)
29 = Voltage
31 = J (Energy joule)
33 = Hz (Frequency)
38 = Real power (Watts)
42 = m3 (Cubic Meter)
61 = VA (Apparent power)
63 = VAr (Reactive power)
65 = Cos? (Power factor)
67 = V² (Volts squared)
69 = A² (Amp squared)
71 = VAh (Apparent energy)
72 = Real energy (Watt-hours)
73 = VArh (Reactive energy)
106 = Ah (Ampere-hours / Available Charge)
119 = ft3 (Cubic Feet)
122 = ft3/h (Cubic Feet per Hour)
125 = m3/h (Cubic Meter per Hour)
128 = US gl (US Gallons)
```

```
129 = US gl/h (US Gallons per Hour)
130 = IMP gl (Imperial Gallons)
131 = IMP gl/h (Imperial Gallons per Hour)
132 = BTU
133 = BTU/h
134 = Liter
137 = L/h (Liters per Hour)
140 = PA(gauge)
155 = PA(absolute)
169 = Therm</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="UInt8"/>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="SummaryMeasurement">
    <xs:annotation>
      <xs:documentation>An aggregated summary measurement reading.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="powerOfTenMultiplier" type="PowerOfTenMultiplierType"
minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The multiplier part of the unit of measure, e.g. "kilo"
(k)</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="timeStamp" type="TimeType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The date and time (if needed) of the summary
measurement.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="uom" type="UomType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The units of the reading, e.g. "Wh"</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="value" type="UInt48" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The value of the summary measurement.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="BatchItemInfo">
    <xs:annotation>
      <xs:documentation>Includes elements that make it possible to include multiple
transactions in a single (batch) request.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="Object">
        <xs:sequence>
          <xs:element name="name" type="HexBinary16" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>An identifier for this object that is only unique within the
containing collection.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="operation" type="UInt8" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Specifies the operation requested of this item.

0=Create
1=Read
2=Update
3=Delete</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="statusCode" type="UInt16" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Indicates the status code of the associated transaction.

200 - Ok
201 - Created
204 - No Content
301 - Moved Permanently
302 - Redirect
304 - Not Modified
400 - Bad Request
401 - Unauthorized
403 - Forbidden
404 - Not Found
405 - Method Not Allowed
410 - Gone
500 - Internal Server Error</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="statusReason" type="String32" minOccurs="0" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>Indicates the reason for the indicated status
code.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="Object">
    <xs:annotation>
      <xs:documentation>Superclass of all object classes to allow
extensions.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="extension" type="xs:anyType" minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>Contains an extension.</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
```

```
    </xs:complexType>
    <xs:complexType name="ServiceStatus">
      <xs:annotation>
        <xs:documentation>Contains the current status of the service.</xs:documentation>
      </xs:annotation>
      <xs:complexContent>
        <xs:extension base="Object">
          <xs:sequence>
            <xs:element name="currentStatus" type="UInt8" minOccurs="1" maxOccurs="1">
              <xs:annotation>
                <xs:documentation>The current status of the service.

  0 = Unavailable
  1 = Normal, operational</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="Subscription">
      <xs:annotation>
        <xs:documentation>Defines the parameters of a subscription between authorized third
  party and data custodian</xs:documentation>
      </xs:annotation>
      <xs:complexContent>
        <xs:extension base="IdentifiedObject"/>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="ApplicationInformation" type="ApplicationInformation"/>
    <xs:element name="Authorization" type="Authorization"/>
    <xs:element name="IntervalBlock" type="IntervalBlock"/>
    <xs:element name="IntervalReading" type="IntervalReading"/>
    <xs:element name="MeterReading" type="MeterReading"/>
    <xs:element name="ReadingQuality" type="ReadingQuality"/>
    <xs:element name="ReadingType" type="ReadingType"/>
    <xs:element name="IdentifiedObject" type="IdentifiedObject"/>
    <xs:element name="ServiceCategory" type="ServiceCategory"/>
    <xs:element name="UsagePoint" type="UsagePoint"/>
    <xs:element name="ElectricPowerQualitySummary" type="ElectricPowerQualitySummary"/>
    <xs:element name="ElectricPowerUsageSummary" type="ElectricPowerUsageSummary"/>
    <xs:element name="AccumulationBehaviourType" type="AccumulationBehaviourType"/>
    <xs:element name="CommodityType" type="CommodityType"/>
    <xs:element name="ConsumptionTierType" type="ConsumptionTierType"/>
    <xs:element name="CurrencyCode" type="CurrencyCode"/>
    <xs:element name="DataQualifierType" type="DataQualifierType"/>
    <xs:element name="DateTimeInterval" type="DateTimeInterval"/>
    <xs:element name="FlowDirectionType" type="FlowDirectionType"/>
    <xs:element name="KindType" type="KindType"/>
    <xs:element name="PhaseCode" type="PhaseCode"/>
    <xs:element name="QualityOfReading" type="QualityOfReading"/>
    <xs:element name="ServiceKind" type="ServiceKind"/>
    <xs:element name="TimeAttributeType" type="TimeAttributeType"/>
    <xs:element name="TOUType" type="TOUType"/>
    <xs:element name="UomType" type="UomType"/>
    <xs:element name="SummaryMeasurement" type="SummaryMeasurement"/>
    <xs:element name="BatchItemInfo" type="BatchItemInfo"/>
    <xs:element name="Object" type="Object"/>
    <xs:element name="ServiceStatus" type="ServiceStatus"/>
    <xs:element name="Subscription" type="Subscription"/>
</xs:schema>
```

Appendices

This section contains informative descriptions, use cases and diagrams used by the committee in developing the Model Business Practices. These Appendices are not normative.

A. Overview

The scope of these Model Business Practices includes authorization by the Retail Customer and the Automatic Data Exchange of the EUI to the Authorized Third Party in accordance with parameters (e.g., term of access, type of data, quantity of data, etc.) determined by the Retail Customer subject to the Governing Documents and Applicable Regulatory Authority. The diagram below shows the logical components involved in this authorization and data exchange process. Note that while the authorization process shown in this figure is made using a web browser, the services provided by the Authorized Third Party are not required to use a web browser to deliver such services.

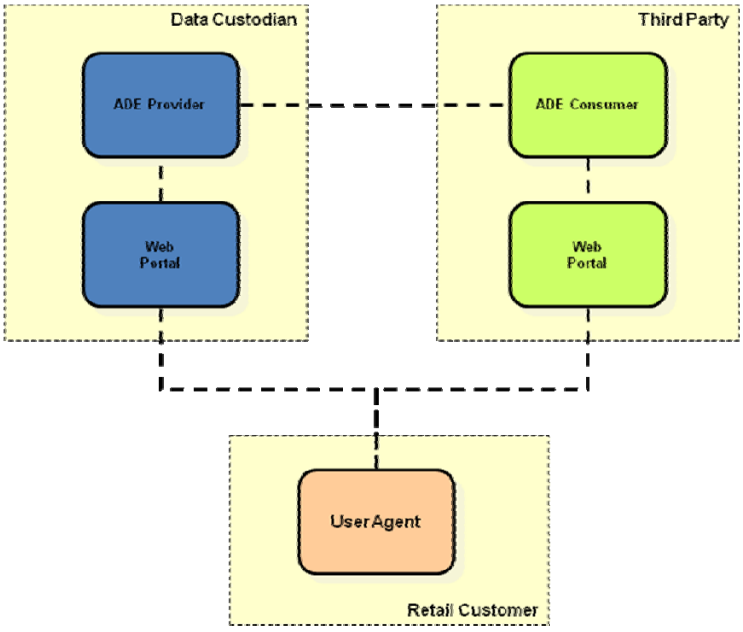

Figure 8: Overview of Logical Components [Change reference to "Third Party" in chart above to "Authorized Third Party".]

B. Use Cases

This section presents a superset of the use cases that are informative of the ~~third party data~~EUI access relationship between Data Custodians and Authorized Third Parties. Alternative use cases are presented for certain activities that can be achieved in different ways, such as the delivery of shared resource information by push or by pull.

The concept of a Shared Resource Key is used throughout these use cases. A Shared Resource Key is a token used to uniquely identify an instance of a Third Party ~~data~~EUI access relationship (i.e., each Retail Customer-Data Custodian-Third Party combination for a particular resource will have a unique Shared Resource Key). A Shared Resource Key, in and of itself, contains no PII regarding the Retail Customer and so can be freely shared among all three roles without unnecessary disclosure of sensitive information. Once the relationship is established, inclusion of a Shared Resource Key in an interaction is sufficient to identify a specific Third Party ~~data~~EUI access relationship.

These Use Cases are illustrative, do not impose any obligations and are subject to the Governing Documents and the requirements of the Applicable Regulatory Authority. All statements of steps and preconditions should be interpreted to follow this constraint.

Each use case contains the following sections:

- Use Case Description: This is a summary of the use case, describing the overall purpose.
- Pre-Conditions: These are conditions that must be true for the use case to be successfully executed.
- Invariant Conditions: These are properties that will be true any time the use case is initiated, regardless of whether it terminates successfully.
- Post-Conditions: These are properties that will be true only if the use case terminates successfully. This requires that all preconditions and all condition checks (e.g., for validity of a request) be satisfied during execution of the use case.
- Basic Path Scenario: This defines the series of steps undertaken by each role during successful execution of the use case. The scenario is depicted graphically in a Unified Modeling Language (UML) sequence diagram and each step is summarized in text.

The following use cases are informative and not normative.

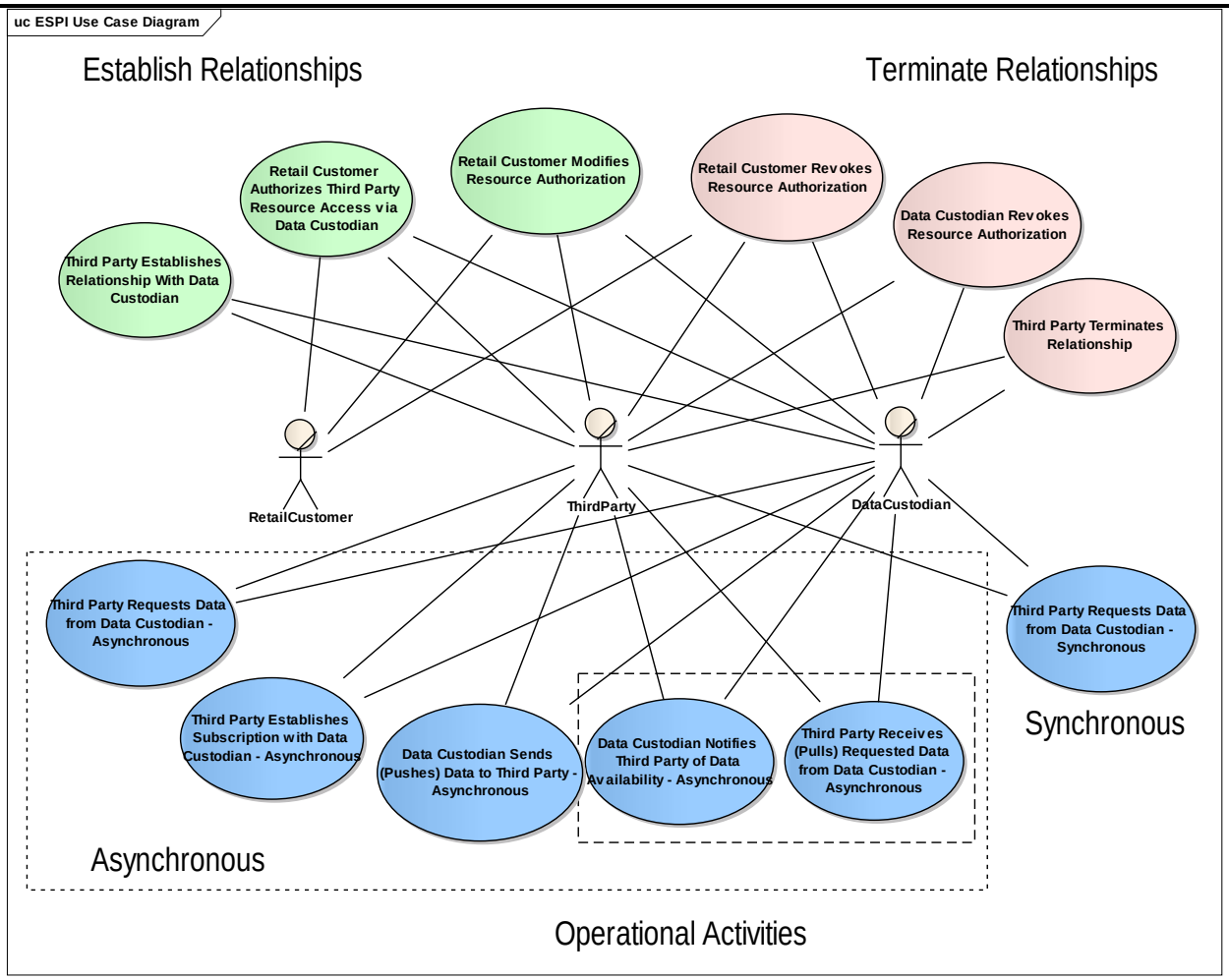

Figure 9**:** ESPI Use Case Diagram [Change reference to "Data" in chart above to "EUI".] [Change reference to "Third Party" in chart above to "Authorized Third Party".]

1: Authorized Third Party Establishes Relationship With Data Custodian

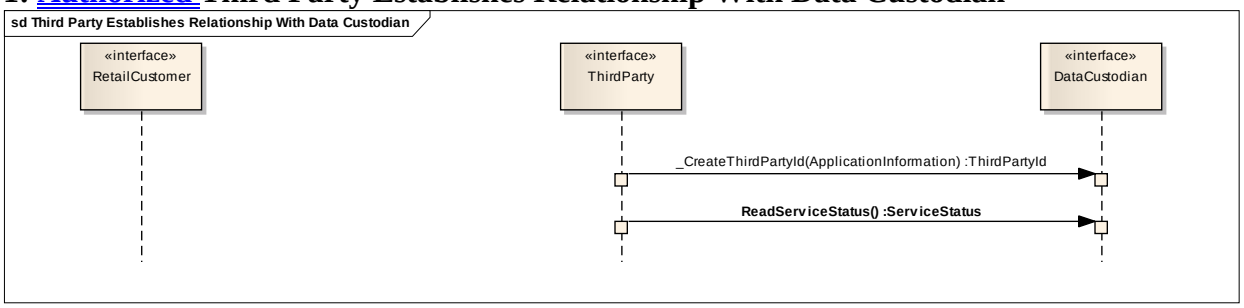

Figure 10**:** Authorized Third Party Establishes Relationship With Data Custodian [Change references to "Third Party" in chart above to "Authorized Third Party".]

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE
For Quadrant: Retail Electric Quadrant

Requesters: Open ADE Task Force
Request No.: R10008
Request Title: Energy Services Provider Interface Standard

Description

An Authorized Third Party service provider wants to register with a Data Custodian to provide services to Retail Customers with ~~data~~EUI stored ~~at~~by the Data Custodian.

Pre-Condition: Authorized Third Party had demonstrated that it meets eligibility, security and privacy requirements.

Invariant Constraint: No ~~resource data~~EUI or ~~personal data~~other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.

Post-Condition: A Shared Identity Key is generated to allow the Authorized Third Party to identify its identity to Data Custodian.

Post-Condition: The Authorized Third Party has permission to get specified ~~resource data~~EUI from the Data Custodian with permission of a Retail Customer.

Scenario: Basic Path

1. The Authorized Third Party wishes to provide value added services to Retail Customers with ~~data~~EUI stored by the Data Custodian.
2. Authorized Third Party requests that the Data Custodian establish relationship.
3. Authorized Third Party provides proof that they meet the applicable requirements for eligibility, data security and privacy protection. **[What proof will there be?]**
4. Authorized Third Party provides description of the services it wishes provide for Retail Customers.
5. The Data Custodian generates an Identity Key for the Authorized Third Party. Authorized Third Party will use this key to identify itself during Use Cases 2 through 12.
6. The Data Custodian adds the Authorized Third Party to its list of available services it presents to Retail Customers in Use Case 2.
7. Authorized Third Party adds Data Custodian to its list of Data Custodians it presents in Use Case 2.
8. Authorized Third Party persists the Identity Key.
9. As needed, Authorized Third Party checks their ability to connect to the service, and obtains the current status of the service.

2: Retail Customer Authorizes Third Party Resource Access via Data Custodian

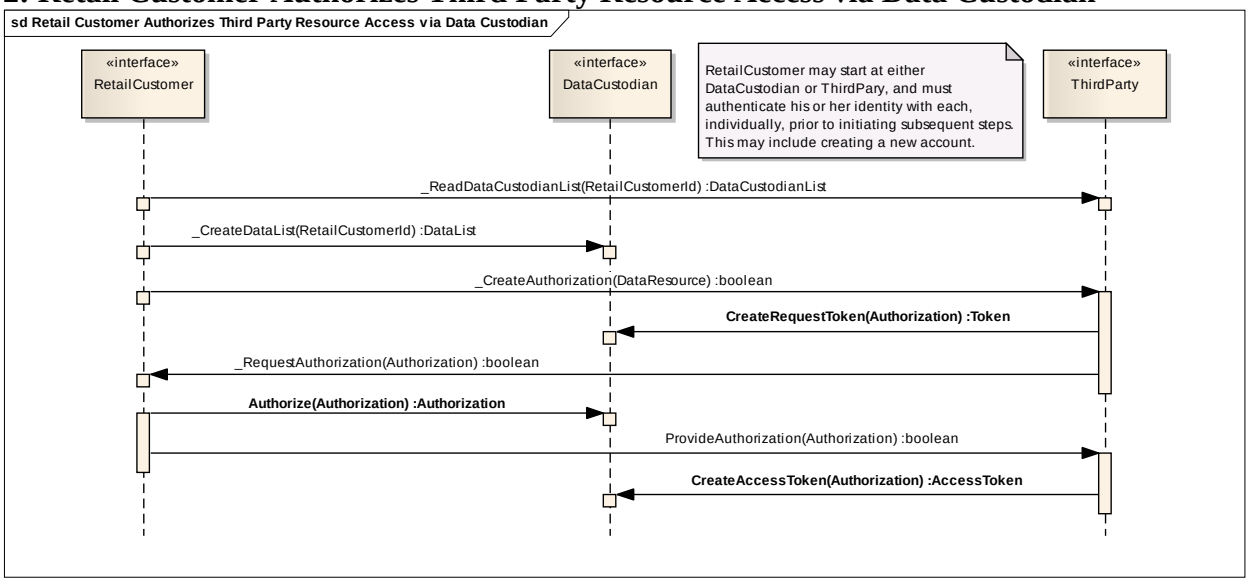

Figure 11**:** Retail Customer Authorizes Third Party Resource Access via Data Custodian [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

A Retail Customer wants to grant permission for a Data Custodian to share a particular ~~data~~EUI resource with an Authorized Third Party. The Retail Customer initiates the process through the Data Custodian.

Pre-Condition: Retail Customer has established accounts with Data Custodian and Authorized Third Party.
Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Data Custodian and Authorized Third Party have published and made Retail Customer aware of their privacy policy related to collection and handling of customer information.
Invariant Constraint: No ~~resource data~~EUI or ~~personal data~~other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.
Post-Condition: A Shared Resource Key is generated to allow all three roles to refer to the same shared resource without disclosing PII. This key is known to all three roles.
Post-Condition: The Authorized Third Party has the Retail Customer's permission to get the specified resource ~~data~~EUI from the Data Custodian.
Post-Condition: The Data Custodian sends the Retail Customer confirmation of establishment of the Authorized Third Party ~~data~~EUI access relationship.

Scenario: Basic Path

1. The Retail Customer decides to grant permission for the Data Custodian to share their resource ~~data~~EUI with the Authorized Third Party.
2. (Optional) Retail Customer finds their appropriate Data Custodian from Authorized Third Party, and navigates to the appropriate place to begin establishment of sharing relationship.
3. Retail Customer requests that the Data Custodian establish a new ~~data~~EUI access relationship.
4. Data Custodian presents the Retail Customer with a list of resources that can be shared with Third Parties. Any additional attributes (e.g., duration for which permission should be granted) that can be selected are also

presented.

5. Retail Customer selects a resource to share, sets any available attributes for the relationship, and specifies an Authorized Third Party ~~that is known to the Data Custodian~~. Selecting these parameters and completing the interaction indicates permission for the Data Custodian to grant the specified Authorized Third Party access to the specified shared resource.
6. The relationship will only be created if the Data Custodian accepts the selections for the Authorized Third Party (e.g., a Data Custodian may constrain access to certain resource attributes depending on resource sensitivity).
7. Data Custodian generates a Shared Resource Key (Request Token) to begin creation of this relationship and provides it to the Authorized Third Party. Each Shared Resource Key is unique to the relationship between a Retail Customer, Data Custodian, Authorized Third Party, and specific ~~data~~EUI resource.
8. Authorized Third Party requests authorization of the token by the Retail Customer, via the Data Custodian.
9. Retail Customer authenticates with Data Custodian and authorizes the Request Token.
10. Authorized Third Party exchanges the authorized Request Token for an Access Token from the Data Custodian.
11. Authorized Third Party and Data Custodian persist the ~~A~~authorization, associating it with its identity of the Retail Customer.

3: Retail Customer Modifies Resource Authorization

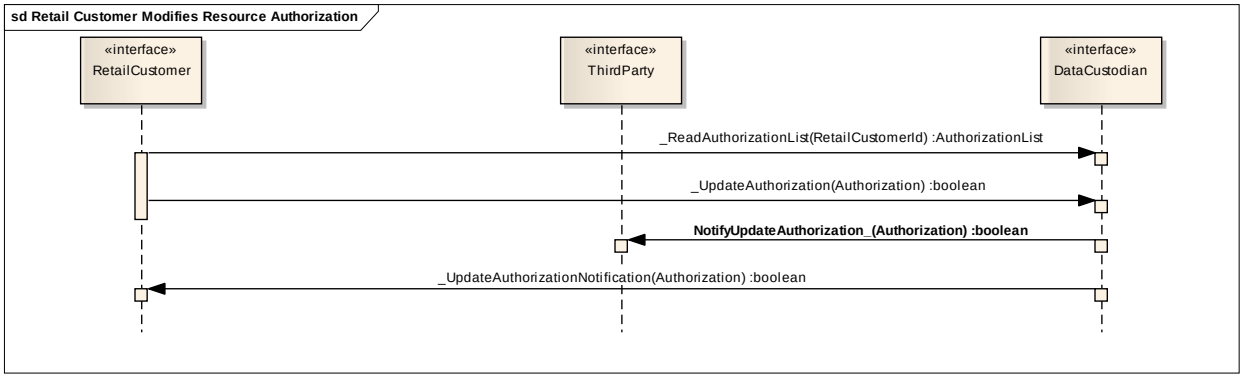

Figure 12**:** Retail Customer Modifies Resource Authorization [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party and wants to either extend or restrict the permissions associated with that relationship.

Pre-Condition: Retail Customer has established a Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Invariant Constraint: No ~~resource data~~EUI or ~~personal data~~other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.
Post-Condition: Future interactions between the Data Custodian and the Authorized Third Party with respect to the specified resource are constrained by the modified permissions.

Post-Condition: The Authorized Third Party handles any ~~data~~EUI not allowed by the termination of the relationship in the manner specified in any service agreements among the parties in the relationship (e.g., all instances of the ~~data~~EUI in control of the Authorized Third Party are deleted).
Post-Condition: The Data Custodian sends the Retail Customer confirmation of modification of the permissions of the Authorized Third Party ~~data~~EUI access relationship.

Scenario: Basic Path
1. Retail Customer chooses to modify relationship permissions with the Data Custodian.
2. Data Custodian presents the Retail Customer with a list of resources that are shared with Third Parties. If the Retail Customer may only grant access to one resource, S2 and S3 may be skipped.
3. Retail Customer chooses particular resource whose permissions he/she wishes to modify.
4. Data Custodian provides available resource attributes and current settings to Retail Customer.
5. Retail Customer chooses new settings.
6. The new permissions governing the relationship will apply only if the Data Custodian accepts the selections for the Authorized Third Party (e.g., a Data Custodian may constrain access to certain resource attributes depending on resource sensitivity).
7. Data Custodian persists the new permissions, which will be used from this point forward to constrain the relationship (until further changed or the relationship is terminated).
8. Data Custodian notifies Authorized Third Party that permissions have changed (identifying the resource by its Shared Resource Key). This notification may be immediate or deferred (e.g., as part of a resource push from Use Case 8, perhaps as part of a header).
9. Data Custodian notifies Retail Customer that permissions have been changed.
10. The Authorized Third Party handles any ~~data~~EUI not allowed by the modification of the resource authorization in the manner specified in any service agreements among the parties in the relationship.

4: Retail Customer Revokes Resource Authorization

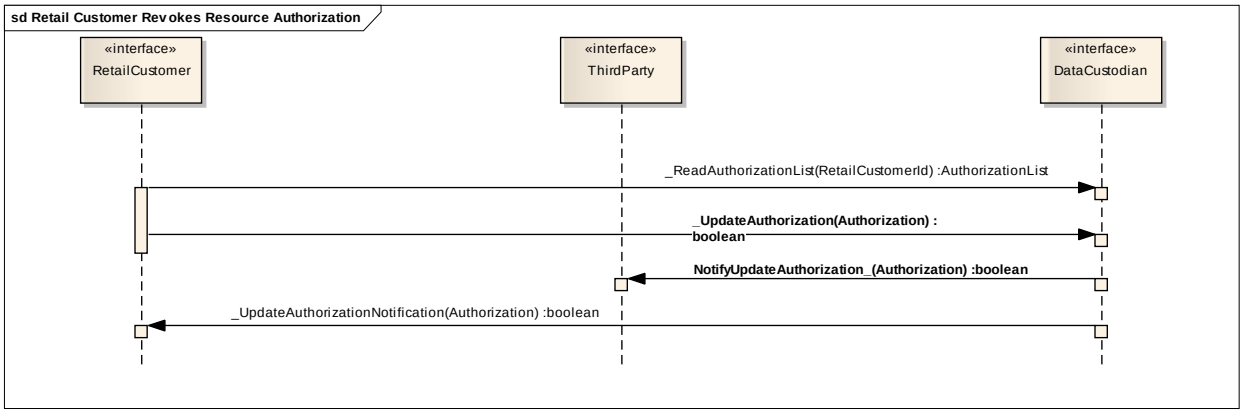

Figure 13**:** Retail Customer Revokes Resource Authorization [Wouldn't this occur on the utility's customer portal and then leverage ESPI to the third party?] [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party and wants to terminate that relationship.

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Invariant Constraint: No ~~resource data or personal data~~EUI or other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.
Post-Condition: Both the Authorized Third Party and the Data Custodian delete the Shared Resource Key for the relationship and no future interactions are permitted for that relationship.
Post-Condition: The Authorized Third Party handles any ~~data~~EUI not allowed by the termination of the relationship in the manner specified in any service agreements among the parties in the relationship (e.g., all instances of the ~~data~~EUI in control of the Authorized Third Party are deleted)
Post-Condition: The Data Custodian sends the Retail Customer confirmation of termination of the Authorized Third Party ~~data~~EUI access relationship.

Scenario: Basic Path
1. Retail Customer requests that Data Custodian terminate the ~~data~~EUI access relationship.
2. Data Custodian presents the Retail Customer with a list of resources for which there are valid relationships with Third Parties. If the Retail Customer only has one valid relationship, S2 and S3 may be skipped.
3. Retail Customer chooses a resource whose relationship is to be terminated.
4. Data Custodian terminates the relationship, deleting the appropriate Shared Resource Key from its list of valid relationships.
5. Data Custodian notifies Authorized Third Party that the relationship has been terminated (identifying the relationship by its Shared Resource Key).
6. Data Custodian notifies Retail Customer that the relationship has been terminated.
7. The Authorized Third Party handles any ~~data~~EUI not allowed by the termination of the relationship, in the manner specified in any service agreements among the parties in the relationship.

5: Data Custodian Revokes Resource Authorization

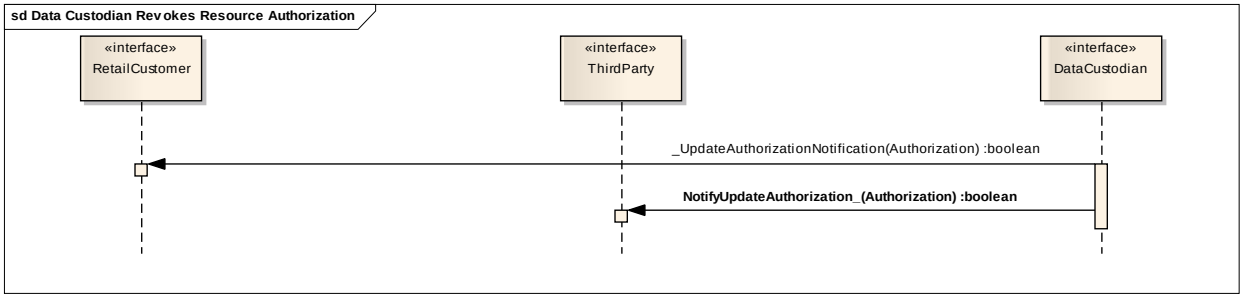

Figure 14**:** Data Custodian Revokes Resource Authorization [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~ EUI access relationship with a particular Data Custodian and Authorized Third Party. The Data Custodian wants to terminate the relationship (for whatever reason).

Pre-Condition: Authorized Third Party has an established account with Data Custodian.

Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Invariant Constraint: No ~~resource data or personal data~~ EUI or other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.
Post-Condition: Both the Authorized Third Party and the Data Custodian delete the Shared Resource Key for the relationship and no future interactions are permitted for that relationship.
Post-Condition: The Authorized Third Party handles any ~~data~~EUI or any other PII provided to the Authorized Third Party during the relationship not allowed by the termination of the relationship in the manner specified in any service agreements among the parties in the relationship (e.g., all instances of the ~~data~~EUI in control of the Authorized Third Party are deleted).
Post-Condition: The Data Custodian sends the Retail Customer notification of termination of the Authorized Third Party ~~data~~EUI access relationship.

Scenario: Basic Path
1. Data Custodian decides to terminate relationship with Authorized Third Party.
2. Data Custodian notifies Retail Customer of termination decision; no acknowledgement or confirmation is required.
3. Data Custodian notifies Authorized Third Party of termination of the relationship, identifying the relationship by a Shared Resource Key.
4. The Authorized Third Party handles any ~~data~~EUI or any other PII provided to the Authorized Third Party during the relationship not allowed by the termination of the relationship, in the manner specified in any service agreements among the parties in the relationship.

6: Authorized Third Party Terminates Relationship

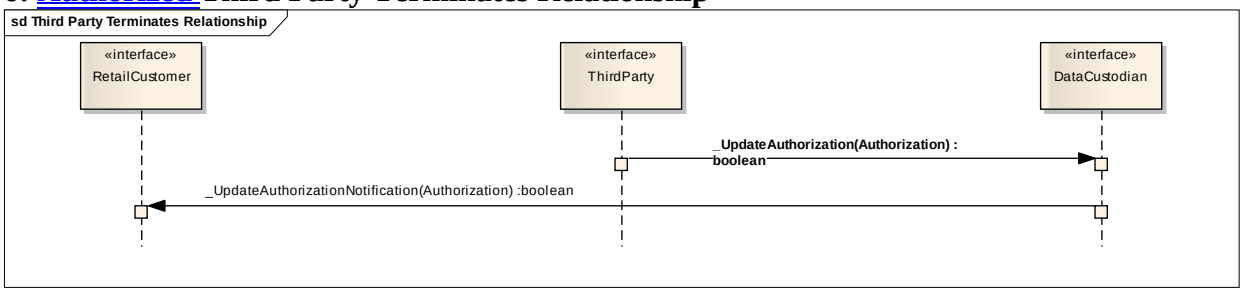

Figure 15**:** Authorized Third Party Terminates Relationship [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party determines that it no longer wants to provide services to the Retail Customer and terminates the relationship.

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established a Authorized Third Party ~~data~~ EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.

Invariant Constraint: No ~~resource data or personal data~~EUI or other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.
Post-Condition: Both the Authorized Third Party and the Data Custodian delete the Shared Resource Key for the relationship and no future interactions are permitted for that relationship.
Post-Condition: The Authorized Third Party handles any ~~data~~EUI or any other PII provided to the Authorized Third Party during the relationship not allowed by the termination of the relationship, in the manner specified in any service agreements among the parties in the relationship (e.g., all instances of the ~~data~~EUI in control of the Authorized Third Party are deleted within
Post-Condition: The Data Custodian sends the Retail Customer notification of termination of the Authorized Third Party ~~data~~EUI access relationship.

Scenario: Basic Path
1. Authorized Third Party decides to terminate an Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship.
2. Authorized Third Party notifies Data Custodian of termination of relationship, identifying the relationship by the Shared Resource Key.
3. An invalid request (e.g., specification of a Shared Resource Key not associated with the Authorized Third Party) will not be accepted.
4. Data Custodian deletes Shared Resource Key, terminating the relationship.
5. Data Custodian notifies the Retail Customer of termination of the relationship. No acknowledgement or confirmation is required.
6. The Authorized Third Party handles any ~~data~~EUI or any other PII provided to the Authorized Third Party during the relationship not allowed by the termination of the relationship, in the manner specified in any service agreements among the parties in the relationship.

7: Authorized Third Party Establishes Subscription with Data Custodian - Asynchronous

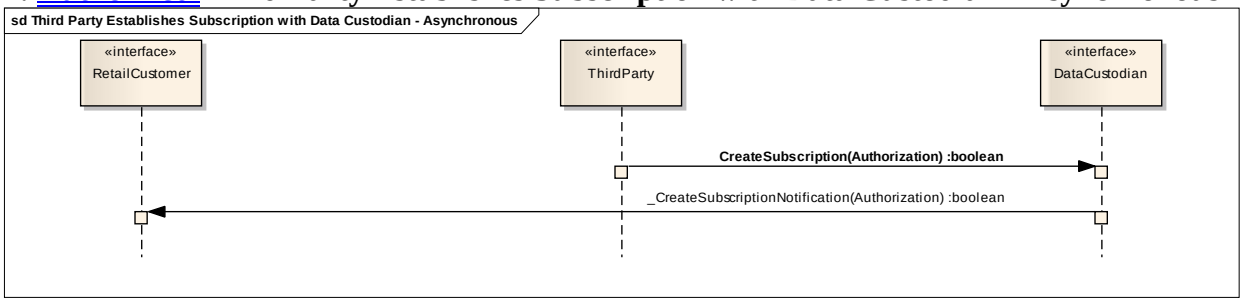

Figure 16**:** Authorized Third Party Establishes Subscription with Data Custodian - Asynchronous [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~ EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party establishes a *subscription* indicating the circumstances (i.e., an agreed-upon schedule and/or specification of special events) under which the Data Custodian should provide the Authorized Third Party with the relevant resource ~~data~~ EUI.
Depending on the services offered by a Data Custodian, the subscription may indicate the circumstances under which the Data Custodian will send ~~resource data~~EUI or only notification that ~~resource data~~ EUI is available (i.e., whether the Data Custodian supports a push or pull model). Subscriptions may be parameterized, if supported by the Data Custodian, to define preferred delivery criteria (e.g., new ~~data~~EUI whenever available or only once per day).

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Invariant Constraint: No ~~resource data~~EUI or ~~personal data~~other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.
Post-Condition: The Data Custodian records a valid subscription on behalf of the Authorized Third Party. Future ~~data~~EUI availability triggers satisfying the subscription will result in the appropriate information being sent to the Authorized Third Party.
Post-Condition: Data Custodian sends the Authorized Third Party confirmation of its subscription request
Post-Condition: Data Custodian sends the Retail Customer notification of the Authorized Third Party's subscription request

Scenario: Basic Path
1. Authorized Third Party requests that the Data Custodian establish a new subscription.
2. Authorized Third Party provides Data Custodian with information defining the subscription request. At a minimum, this information includes a Shared Resource Key identifying the resource whose ~~data~~EUI is to be shared. The information may include additional subscription parameters, as supported by the Data Custodian.
3. The subscription will not be accepted if the Shared Resource Key is invalid.
4. The Data Custodian saves the subscription information, associating the subscription with the Shared Resource Key and the Authorized Third Party.
5. The Data Custodian notifies the Authorized Third Party that the subscription request was successful. No acknowledgement or confirmation is required.
6. The Data Custodian notifies the Retail Customer that the Authorized Third Party has completed a subscription for their ~~resource data~~EUI. No confirmation is required, as the Authorized Third Party already has permissions as indicated by the valid Shared Resource Key. If the subscription is not acceptable to the Retail Customer, Use Case 3 can be exercised to modify permissions for the Authorized Third Party.

8: Authorized Third Party Requests Data from Data Custodian - Asynchronous

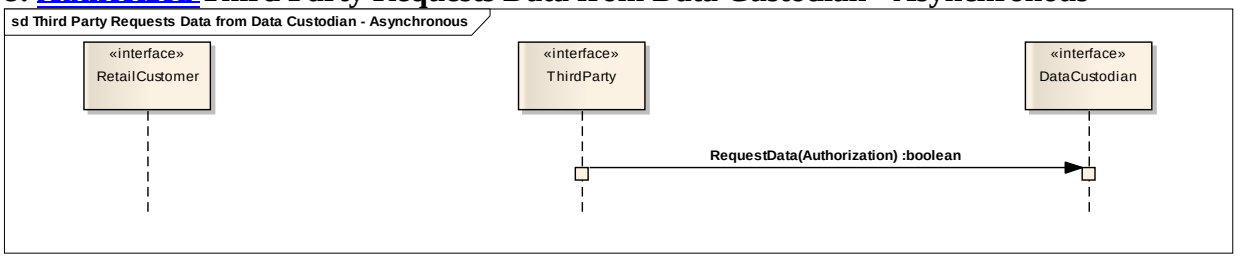

Figure 17**:** Authorized Third Party Requests ~~Data~~EUI from Data Custodian - Asynchronous [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party requests specific ~~resource data~~EUI to be delivered with next transfer.

Pre-Condition: Authorized Third Party has an established account with Data Custodian.

Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship

Invariant Constraint: No ~~resource data~~EUI or ~~personal data~~other PII is provided to the Authorized Third Party by the Data Custodian as part of this interaction.

Post-Condition: The Data Custodian records the request on behalf of the Authorized Third Party. Future ~~data~~EUI availability triggers will result in the appropriate information being sent to the Authorized Third Party. **[What auditing requirements?]**

Post-Condition: Data Custodian sends the Authorized Third Party confirmation of its ~~data~~EUI request.

Scenario: Basic Path

1. Authorized Third Party decides to request ~~resource data~~EUI from the Data Custodian. [The third party should be limited in the number of times it can ask for data within the same period. A Subscription to data is preferred rather than adhoc requests. It is assumed that multiple retail customer usage data can be sent in a batch packet that meets the subscription request?]
2. Data Custodian ~~C~~checks validity of request.
3. Data Custodian queues request for next asynchronous transfer.
4. Data Custodian sends confirmation to Authorized Third Party.

9: Data Custodian Sends (Pushes) ~~Data~~EUI to Authorized Third Party - Asynchronous

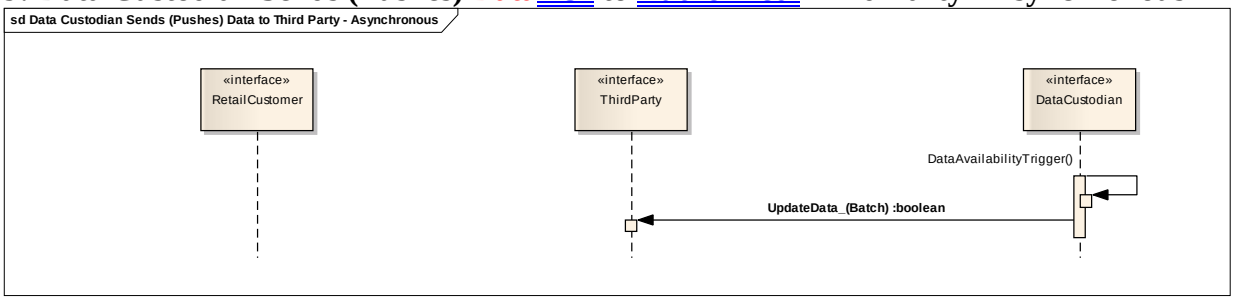

Figure 18**:** Data Custodian Sends (Pushes) ~~Data~~EUI to Authorized Third Party - Asynchronous [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party has established a subscription for receiving the relevant ~~resource data~~EUI from the Data Custodian. This ~~resource data~~EUI is sent (pushed) to the subscribed Authorized Third Party by the Data Custodian when an event triggers indicates a need to push new ~~resource data~~EUI.

Conditions observable to the Data Custodian change, causing an ~~data~~EUI availability trigger to be checked to see if there is a need to push ~~resource data~~EUI to the Authorized Third Party. Such triggers can be caused by any of the following observable changes

- New ~~resource data~~EUI is received by the Data Custodian
- A new subscription is received by the Data Custodian
- A pre-defined interval has elapsed
- A request for ~~resource data~~EUI has been received from a Authorized Third Party

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Pre-Condition: A subscription by the Authorized Third Party to receive ~~resource data~~EUI from the Data Custodian has been established.
Invariant Constraint: No ~~personal information~~PII other than EUI is provided to the Authorized Third Party by the Data Custodian.
Post-Condition: The Data Custodian sends ~~resource data~~EUI to the subscribed Authorized Third Party.
Post-Condition: Only ~~data~~EUI specifically requested or modified and in a subscription is sent to the Authorized Third Party.

Scenario: Basic Path
1. An ~~data~~EUI availability trigger is received by the Data Custodian.
2. Data Custodian determines the Shared Resource Keys associated with the ~~data~~EUI availability trigger. It then determines if there are any subscriptions associated with the Shared Resource Key and whether the conditions of the subscription are satisfied (i.e., if it is time to send out ~~resource data~~EUI). If so, it proceeds to S3.
3. Data Custodian determines the Authorized Third Party associated with the subscriptions. This includes a check that the Authorized Third Party is still in a valid relationship with the Data Custodian and any other relevant checks prior to releasing ~~resource data~~EUI to that Authorized Third Party.
4. Data Custodian provides ~~data resources~~EUI to Authorized Third Party.
5. Authorized Third party ~~persists~~retains ~~data~~EUI for the period specified by ~~data~~ EUI retention requirements.

10: Data Custodian Notifies Authorized Third Party of Data Availability - Asynchronous

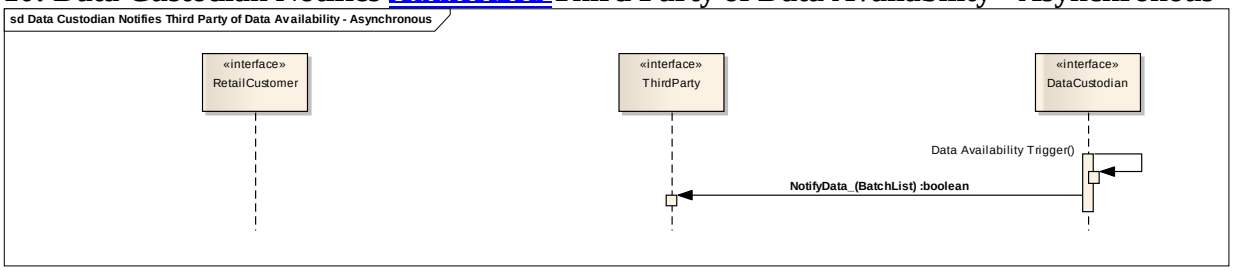

Figure 19**:** Data Custodian Notifies Authorized Third Party of ~~Data~~EUI Availability - Asynchronous [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party has established a subscription for receiving the relevant ~~resource data~~EUI from the Data Custodian. An Authorized Third Party is notified when new ~~data~~EUI satisfying its subscription parameters is available.
Conditions observable to the Data Custodian change, causing an ~~data~~EUI availability trigger to be checked to see if there is a need to notify an Authorized Third Party of ~~resource data~~EUI availability. Such triggers can be caused by any of the following observable changes:

- New ~~resource data~~EUI is received by the Data Custodian

- A new subscription is received by the Data Custodian
- A pre-defined interval has elapsed
- A request for ~~resource data~~EUI has been received from an Authorized Third Party

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established an Authorized Third Party EUI~~data~~ access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Pre-Condition: Data Custodian has ~~resource data~~EUI relevant to the Authorized Third Party
Invariant Constraint: No ~~personal information~~PII other than EUI is provided to the Authorized Third Party by the Data Custodian
Post-Condition: The Data Custodian has ~~resource data~~EUI (e.g., electricity usage data) that is available for access by the Authorized Third Party
Post-Condition: The Data Custodian sends the Authorized Third Party notification of availability of EUI~~resource data~~.

Scenario: Basic Path
1. An ~~data~~EUI availability trigger event is received by the Data Custodian.
2. Data Custodian determines the Shared Resource Keys associated with the ~~data~~EUI availability trigger. The Data Custodian then determines if there are any subscriptions associated with the Shared Resource Key and whether the conditions of the subscription are satisfied (i.e., if it is time to notify a Authorized Third Party). If so, it proceeds to S3.
3. Data Custodian determines the Authorized Third Party associated with subscriptions. This includes a check that the Authorized Third Party is still in a valid relationship with the Data Custodian and any other relevant checks prior to determining that it is appropriate to send ~~resource data~~EUI to that Authorized Third Party
4. Data Custodian notifies the Authorized Third Party of the availability of ~~resource data~~EUI associated with the Shared Resource Key. Note that notification can take different forms. Notification could be sent asynchronously as soon as the trigger is evaluated. Notification for several resources could be bundled for delivery to a common Authorized Third Party. Notification could be queued, awaiting the next scheduled interaction with the Authorized Third Party (e.g., as part of a response to a regular pull from the Authorized Third Party). No mechanism or timing is specified.

11: Authorized Third Party Receives (Pulls) Requested ~~Data~~EUI from Data Custodian - Asynchronous

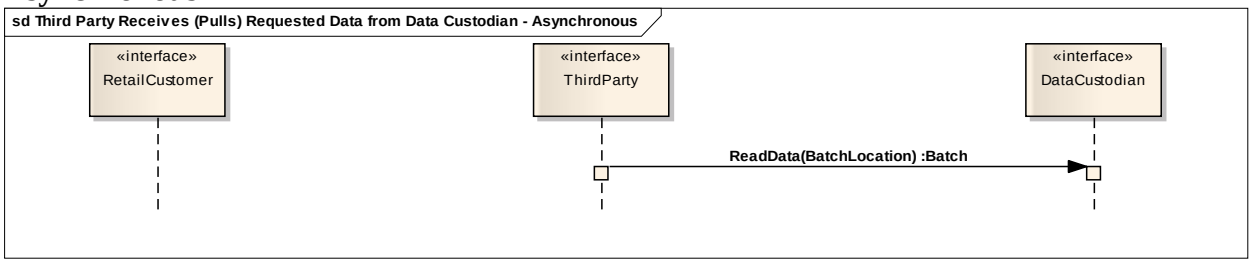

Figure 20**:** Authorized Third Party Receives (Pulls) Requested ~~Data~~EUI from Data Custodian - Asynchronous
[Change reference to "Third Party" in chart above to "Authorized Third Party".]

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE
For Quadrant: Retail Electric Quadrant

Requesters: Open ADE Task Force
Request No.: R10008
Request Title: Energy Services Provider Interface Standard

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party requests the relevant subscribed and requested ~~resource data~~EUI from the Data Custodian, who replies with the ~~data~~EUI if the request is valid.

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Pre-Condition: Data Custodian has ~~resource data~~EUI relevant to the Authorized Third Party.
Invariant Constraint: No ~~personal data~~PII other than EUI is provided to Third Parties by the Data Custodian.
Post-Condition: The Data Custodian replies with the requested ~~data~~EUI.
Post-Condition: Only the requested ~~resource data~~EUI is provided by the Data Custodian.

Scenario: Basic Path
1. Authorized Third Party receives notification or periodically attempts to pull ~~resource data~~EUI from the Data Custodian.
2. Data Custodian checks validity of request.
3. Data Custodian replies with requested and subscribed ~~resource data~~EUI to Authorized Third Party.
4. Authorized Third Party persists ~~resource data~~EUI for use in performing services for Retail Customer.

12: Authorized Third Party Requests ~~Data~~EUI from Data Custodian - Synchronous

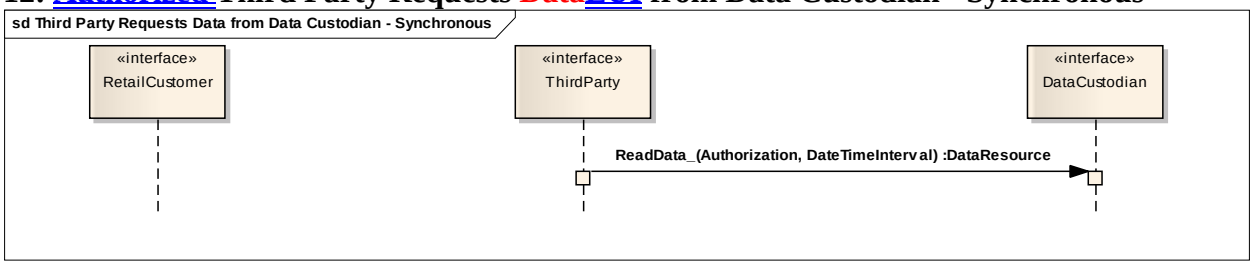

Figure 21**:** Authorized Third Party Requests ~~Data~~EUI from Data Custodian - Synchronous [Change reference to "Third Party" in chart above to "Authorized Third Party".]

Description

The Retail Customer has an existing Authorized ~~t~~Third ~~p~~Party ~~data~~EUI access relationship with a particular Data Custodian and Authorized Third Party. The Authorized Third Party directly requests specific resource ~~data~~EUI from the Data Custodian, who replies with the requested ~~data~~EUI synchronously if the request is valid.

Pre-Condition: Authorized Third Party has an established account with Data Custodian.
Pre-Condition: Retail Customer has established an Authorized Third Party ~~data~~EUI access relationship with the Data Custodian and the Authorized Third Party with respect to a particular resource, resulting in a unique Shared Resource Key identifying the relationship.
Pre-Condition: Authorized Third Party requests authorized ~~resource data~~EUI.
Invariant Constraint: No ~~personal data~~PII other than EUI is provided to Third Parties by the Data Custodian.
Post-Condition: The Data Custodian replies with the requested ~~data~~EUI.

Post-Condition: Only the requested ~~resource data~~EUI is provided by the Data Custodian.

Scenario: Basic Path

1. Authorized Third Party decides to pull ~~resource data~~EUI from the Data Custodian.
2. Authorized Third Party specifies the ~~resource data~~EUI being requested. The request must contain the Shared Resource Key. It may also contain parameters (e.g., the period over which the specified ~~data~~EUI is requested), if permitted by Data Custodian.
3. Data Custodian checks validity of request (e.g., Shared Resource Key is still valid and registered with this Authorized Third Party or validity of any additional parameters).
4. Data Custodian sends requested ~~resource data~~EUI to Authorized Third Party.
5. Authorized Third Party persists ~~resource data~~EUI for use in performing services for Retail Customer.

C. ESPI Abstract Services

This section provides definition of the abstract services used in the use cases. These are the services that will be specified fully for the ESPI ~~m~~Model ~~b~~Business ~~p~~Practice specification. The services are named using the following conventions, since not all are intended to be fully standardized.

- Underscore before the method name means "to be done, but not standardized"
- Underscore after the method name means "optional"

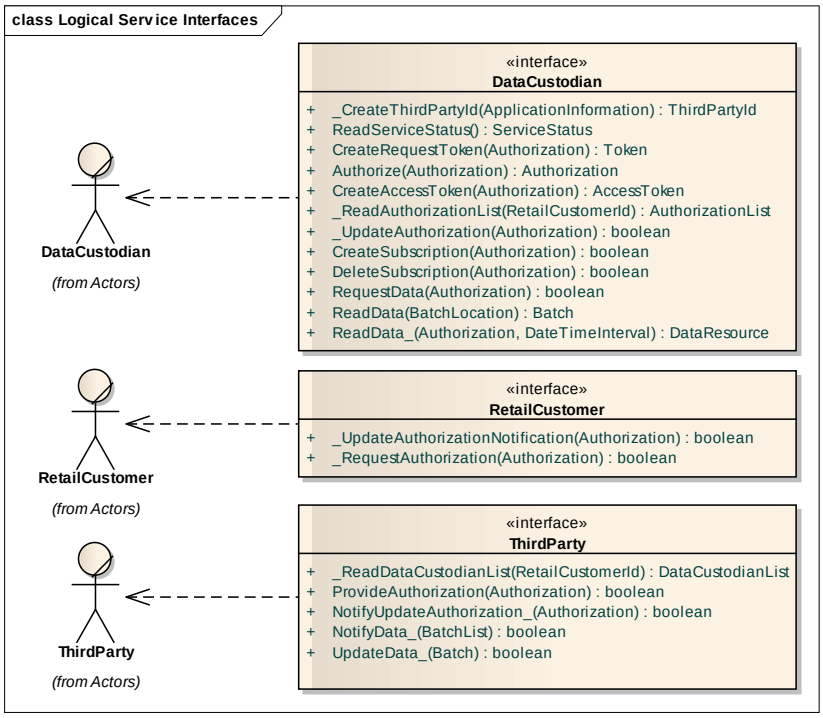

Figure: 22 - Logical Service Interfaces [Change reference to "Third Party" in chart above to "Authorized Third Party".]

DataCustodian

The DataCustodian service interface contains methods to be called by Authorized Third Party in order to authorize and receive ~~data~~EUI.

Operations

Method	Notes	Parameters
_CreateThirdPartyId() ThirdPartyId	Allows Data Custodian to obtain agreement to terms of service, contact information and application details about a Authorized Third Party application. Provides Authorized Third Party with service key and consumer secret.	**ApplicationInformation** [in] ApplicationInformation
ReadServiceStatus() ServiceStatus	Allows Authorized Third Parties to check their ability to access the Data Custodian service,	

Method	Notes	Parameters
	and its current status.	
CreateRequestToken() Token	Allows Authorized Third Party to request an unauthorized request token.	**Authorization** [in] authRequest
Authorize() Authorization	Provides ability for Retail Customer to authenticate and verify desire to authorize an Authorized Third Party request token. This results in a verifier to be used with CreateAccessToken.	**Authorization** [in] authorization
CreateAccessToken() AccessToken	Allows Authorized Third Party to exchange an authorized request token for an access token.	**Authorization** [in] authRequest
_ReadAuthorizationList() AuthorizationList	Allows Retail Customer to choose an existing Authorization. Is not standardized, since the method involves user input.	**RetailCustomerId** [in] customerID
_UpdateAuthorization() boolean	Provides ability to update an existing Authorization.	**Authorization** [in] authorization
CreateSubscription() boolean	Allows Authorized Third Party to request ongoing updates to the ~~data resources~~ EUI associated with the specified Authorization, to be delivered asynchronously.	**Authorization** [in] authorization
DeleteSubscription() boolean	Removes the ~~data resources~~EUI associated with the specified Authorization from the subscriptions.	**Authorization** [in] authorization
RequestData() boolean	Allows Third Parties to request initial transfer of existing authorized ~~data~~EUI, or re-transfer of same. Results are delivered asynchronously.	**Authorization** [in] authorization
ReadData() Batch	Allows Authorized Third Party request ("pull") of asynchronously requested and subscribed data.	**BatchLocation** [in] batch
ReadData_() DataResource	Allows "on demand" (synchronous) access to authorized ~~data~~EUI. Some providers may choose not to make this method available. [Not suggested.]	**Authorization** [in] authorization **DateTimeInterval** [in] requestedInterval

RetailCustomer

The Retail Customer service interface represents methods used to make requests of the Retail Customer.

Operations

Method	Notes	Parameters
_UpdateAuthorizationNotification() boolean	This method is not standardized, but provides notification to the Retail Customer that an Authorization was updated. May be optional, based on policies of Data Custodian.	**Authorization** [in] authorization
_RequestAuthorization() boolean	This method represents the delivery of the request to authorize an access grant. The normal flow implements this using a URL redirect, but other methods may be possible.	**Authorization** [in] authorization

Authorized ThirdParty

The Authorized Third Party service interface contains methods to be called by the Data Custodian.

Operations

Method	Notes	Parameters
_ReadDataCustodianList() DataCustodianList	This is a non-standardized method to allow the Retail Customer to find their Data Custodian.	**RetailCustomerId** [in] reatilCustomerID
ProvideAuthorization() boolean	This method represents the callback after authorization of a request token.	**Authorization** [in] authorization
NotifyUpdateAuthorization_() boolean	This method allows for a Data Custodian to notify an Authorized Third Party when an authorization has been modified, so that timely requests to extend authorizations can be provided to Retail Customer, and so that the Authorized Third Party can differentiate between problems and lack of authorization.	**Authorization** [in] authorization
NotifyData_() boolean	This optional method can be implemented in order to avoid having to poll for new ~~data~~EUI. It is called by the Data Custodian to indicate that requested authorized ~~data~~EUI is available via ReadData.	**BatchList** [in] batchList
UpdateData_() boolean	This method can be implemented by the Authorized Third Party to allow asynchronous transfers to use the "push" model for delivery. If used, notify is not used, and polling is avoided.	**Batch** [in] data

The following table contains a listing of these Logical Interface Operations, along with the expected Physical Operation for informative purposes. Physical operation names in parentheses are passed programmatically

Actor	Description	Logical	Physical
Data Custodian	Ability to get service status	ReadServiceStatus	ServiceStatus
Data Custodian	Initiate signed request_token request per RFC 5849	CreateRequestToken	request_token
Data Custodian	Initiate signed authorize request per RFC 5849	Authorize	authorize
Data Custodian	Initiate signed access_token request per RFC 5849	CreateAccessToken	access_token
Data Custodian	Update existing authorization	NotifyUpdateAuthorization_	Authorization
Data Custodian	Revoke existing authorization (Retail Customer)	NotifyUpdateAuthorization_	Authorization
Data Custodian	Terminate existing authorization via service	UpdateAuthorization	Authorization
Data Custodian	Request subscription to authorized resource	CreateSubscription	Subscription (from config)
Data Custodian	Request authorized data resource(s)	RequestData	(request_token scope)

Actor	Description	Logical	Physical
Data Custodian	Receive requested and subscribed ~~data resources~~ EUI	ReadData	(dataCustodianDefaultBatchResource)
Data Custodian	Request and receive authorized ~~data resource(s)~~EUI	ReadData_	(request_token scope)
Third Party	Initiate callback specified in request_token per RFC 5849	ProvideAuthorization	(request_token callback)
Third Party	Revoke existing authorization (Data Custodian)	NotifyUpdateAuthorization_	Authorization
Third Party	Send requested and subscribed ~~data resources~~ EUI	UpdateData_	(thirdPartyDefaultBatchResource)
Third Party	Notify requested and subscribed ~~data~~EUI is available	NotifyData_	(thirdPartyDefaultNotifyResource)

Logical Information Model

This section contains descriptions of the data elements used in the abstract services.

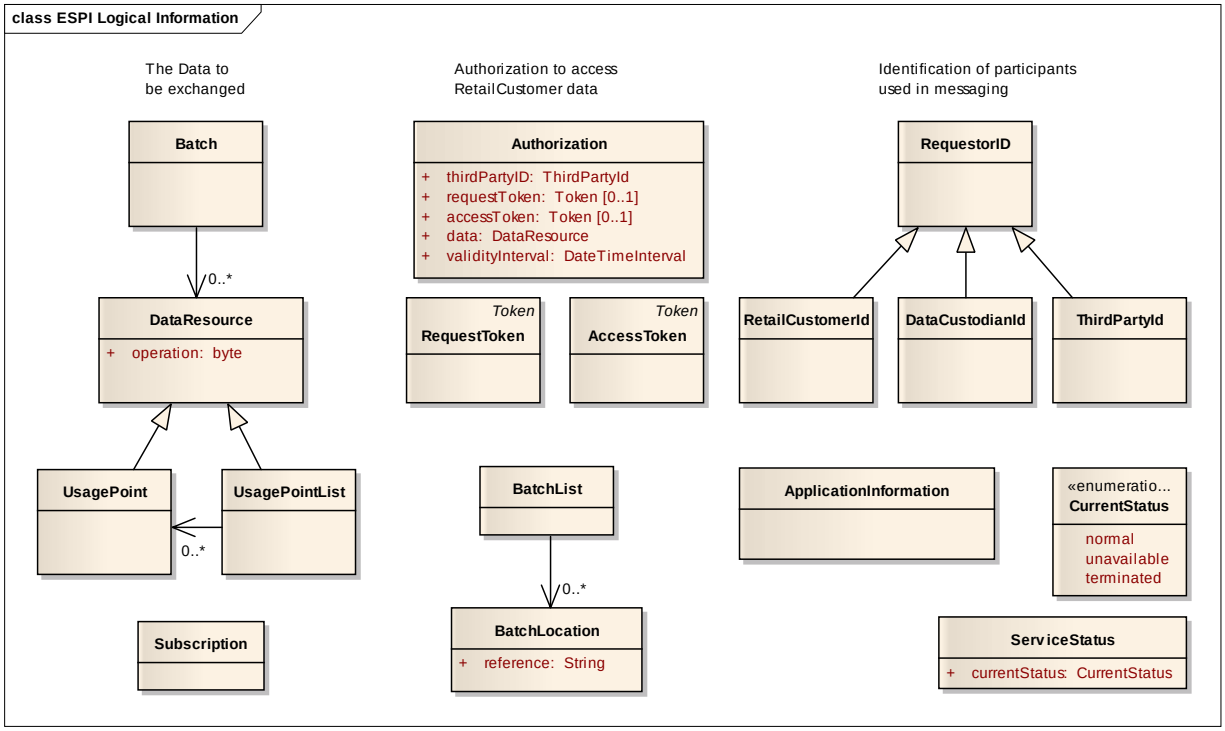

Figure 23: ESPI Logical Information Model

AccessToken

AccessToken is a shared key representing the relationship between a RetailCustomer, DataCustodian, and Authorized ThirdParty for ~~a~~ particular ~~data resource~~EUI.

ApplicationInformation

Includes (non-standardized) information about the Authorized Third Party Application requesting access to the DataCustodian services. Information requested may include items such as Organization Name, Website, Contact Info, Application Name, Description, Icon, Type, default Notification and Callback endpoints, and may also include agreement with terms of service.

Authorization

Description of an authorization. Includes the information constraining and defining access to the ~~Data~~EUI. May include additional security elements, such as signature, timestamp, nonce, etc. as well as callback to allow redirection of the user agent.

Name	Type	Description
thirdPartyID	*ThirdPartyId*	The identifier issued to the Authorized Third Party by the Data Custodian.
requestToken	*Token*	An unauthorized token, issued by Data Custodian.
accessToken	*Token*	An access token that has been authorized by the user at the Data Custodian.
data	*DataResource*	The data resource to which the authorization was granted.
validityInterval	*DateTimeInterval*	The date time interval during which this access is authorized.

Batch

This object is simply a container to hold a number of data resources.

BatchList

BatchList is a container to refer to a multiple batches of data.

BatchLocation

Specifies the location of a specific batch of data.

Name	Type	Description
reference	*String*	

CurrentStatus «enumeration»

Valid values for current status.

Name	Type	Description
normal		
unavailable		
terminated		

DataCustodianId

An identifier for the Data Custodian.

DataResource

Generalization of any ~~data~~EUI object that can be exchanged.

Name	Type	Description
operation	*byte*	Create, Update, Delete

RequestToken

RequestToken is the first step toward obtaining an authorized AccessToken.

RequestorID

Generic superclass for identifiers.

RetailCustomerId

An identifier for the Retail Customer.

ServiceStatus

Contains the current status of the service.

Name	Type	Description
currentStatus	*CurrentStatus*	

Subscription

Defines the parameters of a subscription between Authorized ~~t~~Third ~~p~~Party and ~~d~~Data ~~c~~Custodian

ThirdPartyId

An identifier for the Authorized Third Party.

Token

A Token is a key associated with a secret used for securing exchanges.

UsagePointList

Container to hold multiple UsagePoint objects.

D. Model Conformance Information

The following table provides information about the elements included in ESPI and their relation to the NAESB PAP10 Energy Usage Information Model as well as the IEC TC57 CIM Model. Harmonization across these models

is a goal of this recommendation, as is aligning with other usage information interfaces, including Smart Energy Profile 2.0.

PAP10 EUI Model Element	Type	ESPI Model Element	Type	CIM Notes
CustomerAuthorisation.name	String	(OAuth) access_token		N/A
CustomerAuthorisation.validityInterval	DateTimeInterval	(OAuth 2.0) expires_in		N/A
UsagePoint.name	String	UsagePoint.mRID	HexBinary128	Same
UsagePoint.description	String	UsagePoint.description	String32	Same
		UsagePoint.status	UInt8	connectionState
ServiceCategory.kind	ServiceKind	ServiceCategory.kind	ServiceKind	Same
ServiceKind.electricity		ServiceKind 0		(encoded)
ServiceKind.gas		ServiceKind 1		(encoded)
ServiceKind.water		ServiceKind 2		(encoded)
MeterReading.name	String	MeterReading.mRID	HexBinary128	Same
		MeterReading.description	String32	Same
ReadingType.name	String	ReadingType.mRID	HexBinary128	Same
		ReadingType.description	String32	Same
ReadingType.defaultQuality	QualityOfReading	ReadingType.defaultQuality	QualityOfReading	Recommended extension
ReadingType.direction	ReadingDirection	ReadingType.flowDirection	FlowDirectionType	Same
ReadingType.intervalLength	Duration	ReadingType.intervalLength	UInt32	Recommended extension
ReadingType.kind	ReadingKind	ReadingType.kind	KindType	measurementKind
ReadingType.multiplier	UnitMultiplier	ReadingType.powerOfTenMultiplier	PowerOfTenMultiplierType	Recommended extension
ReadingType.unit	UnitSymbol	ReadingType.uom	UomType	unit
		ReadingType.accumulationBehaviour	AccumulationBehaviourType	accumulation
		ReadingType.dataQualifier	DataQualifierType	Recommended extension
		ReadingType.tou	TOUType	Same
		ReadingType.currency	CurrencyCode	Same
		ReadingType.commodity	CommodityType	Same
		ReadingType.consumptionTier	ConsumptionTierType	Same
		ReadingType.phase	PhaseCode	phases
		IntervalBlock.mRID	HexBinary128	Recommended extension
		IntervalBlock.description	String32	Recommended extension
		IntervalBlock.interval	DateTimeInterval	Recommended extension
Reading.cost	Float	Reading.cost	UInt48	Recommended extension
Reading.timeStamp	AbsoluteDateTime	Reading.timePeriod	DateTimeInterval	Same
Reading.value	Float	Reading.value	UInt48	Same
ReadingQuality.quality	QualityOfReading	ReadingQuality.quality	QualityOfReading	Recommended extension
DateTimeInterval.start	AbsoluteDateTime	DateTimeInterval.start	TimeType	Same
DateTimeInterval.duration	Duration	DateTimeInterval.duration	UInt32	Uses "end" instead of "duration"
QualityOfReading.estimated		QualityOfReading *.8.0		(encoded)
QualityOfReading.raw		QualityOfReading 1.*		(encoded)
QualityOfReading.validated		QualityOfReading *.0.1		(encoded)

RECOMMENDATION TO NAESB EXECUTIVE COMMITTEE
For Quadrant: Retail Electric Quadrant

Requesters: Open ADE Task Force
Request No.: R10008
Request Title: Energy Services Provider Interface Standard

PAP10 EUI Model Element	Type	ESPI Model Element	Type	CIM Notes
ReadingDirection.forward		FlowDirectionType 1		(encoded)
ReadingDirection.reverse		FlowDirectionType 19		(encoded)
ReadingKind.energy		FlowDirectionType 12		(encoded)
ReadingKind.power		FlowDirectionType 37		(encoded)
ReadingKind.demand		FlowDirectionType 8		(encoded)
UnitMultiplier.micro		PowerOfTenMultiplierType -6		(encoded)
UnitMultiplier.m		PowerOfTenMultiplierType -3		(encoded)
UnitMultiplier.c		PowerOfTenMultiplierType -2		(encoded)
UnitMultiplier.d		PowerOfTenMultiplierType -1		(encoded)
UnitMultiplier.k		PowerOfTenMultiplierType 3		(encoded)
UnitMultiplier.M		PowerOfTenMultiplierType 6		(encoded)
UnitMultiplier.G		PowerOfTenMultiplierType 9		(encoded)
UnitMultiplier.T		PowerOfTenMultiplierType 12		(encoded)
UnitMultiplier.none		PowerOfTenMultiplierType 0		(encoded)

4. SUPPORTING DOCUMENTATION

a. Description of Request:

b. Description of Recommendation:

c. Business Purpose:

d. Commentary/Rationale of Subcommittee(s)/Task Force(s):

Please see the ESPI Minutes posted on the ESPI web page: http://www.naesb.org/espi_task_force.asp