



Energy Interoperation Common Transactive Services (CTS) Version 1.0

Committee Specification Draft 01

29 October 2021

This stage:

<https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/csd01/ei-cts-v1.0-csd01.pdf> (Authoritative)
<https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/csd01/ei-cts-v1.0-csd01.html>
<https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/csd01/ei-cts-v1.0-csd01.docx>

Previous stage:

N/A

Latest stage:

<https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/ei-cts-v1.0.pdf> (Authoritative)
<https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/ei-cts-v1.0.html>
<https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/ei-cts-v1.0.docx>

Technical Committee:

OASIS Energy Interoperation TC

Chairs:

David Holmberg (david.holmberg@nist.gov), NIST
William T. Cox ([wtcox@coxsoftwarearchitects.com](mailto:wtcx@coxsoftwarearchitects.com)), Individual

Editor:

Toby Considine (toby.considine@unc.edu), University of North Carolina at Chapel Hill

Related work:

This document replaces or supersedes:

- Common Transactive Services 1.0. The Energy Mashup Lab Specification. Edited by William T. Cox, Toby Considine 30 November 2020. <https://www.theenergymashuplab.org/s/cts-1-0-draft-20201130.pdf>

This document is related to:

- *Energy Interoperation Version 1.0*. Edited by Toby Considine, 11 June 2014. OASIS Standard. <http://docs.oasis-open.org/energyinterop/ei/v1.0/os/energyinterop-v1.0-os.html> Latest version: <http://docs.oasis-open.org/energyinterop/ei/v1.0/energyinterop-v1.0.html> and its TeMIX Profile
- *Energy Market Information Exchange (EMIX) Version 1.0*. Edited by Toby Considine. Latest version: <http://docs.oasis-open.org/emix/emix/v1.0/emix-v1.0.html>.
- *WS-Calendar Platform Independent Model (PIM) Version 1.0*. Edited by William Cox and Toby Considine. Latest version: <http://docs.oasis-open.org/ws-calendar/ws-calendar-pim/v1.0/ws-calendar-pim-v1.0.html>.
- *Schedule Signals and Streams Version 1.0*. Edited by Toby Considine and William T. Cox. Latest version: <http://docs.oasis-open.org/ws-calendar/streams/v1.0/streams-v1.0.html>.

Abstract:

Common Transactive Services (CTS) permits energy consumers and producers to interact through energy markets by simplifying actor interaction with any market. CTS is a streamlined and simplified

profile of the OASIS Energy Interoperation (EI) specification, which describes an information and communication model to coordinate the exchange of energy between any two Parties that consume or supply energy, such as energy suppliers and customers, markets and service providers.

Status

This document was last revised or approved by the OASIS Energy Interoperation TC on the above date. The level of approval is also listed above. Check the "Latest stage" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=energyinterop#technical.

TC members should send comments on this document to the TC's email list. Others should send comments to the TC's public comment list, after subscribing to it by following the instructions at the "Send A Comment" button on the TC's web page at <https://www.oasis-open.org/committees/energyinterop/>.

This document is provided under the [RF on Limited Terms](#) Mode of the [OASIS IPR Policy](#), the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this document, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC's web page (<https://www.oasis-open.org/committees/energyinterop/ipr.php>).

Note that any machine-readable content ([Computer Language Definitions](#)) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

Key words:

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [[RFC2119](#)] and [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

Citation format:

When referencing this document, the following citation format should be used:

[Energyinterop-CTS-v1.0]

Energy Interoperation Common Transactive Services (CTS) Version 1.0. Edited by Toby Considine. 29 October 2021. OASIS Committee Specification Draft 01. <https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/csd01/ei-cts-v1.0-csd01.html>. Latest stage: <https://docs.oasis-open.org/energyinterop/ei-cts/v1.0/ei-cts-v1.0.html>.

Notices:

Copyright © OASIS Open 2021. All Rights Reserved.

Distributed under the terms of the OASIS IPR Policy, [<https://www.oasis-open.org/policies-guidelines/ipr>]. For complete copyright information please see the Notices section in the Appendix.

Table of Contents

1	Introduction.....	9
1.1	Application of the Common Transactive Services	9
1.2	Support for Developers.....	10
1.3	Naming Conventions	10
1.4	Editing Conventions.....	11
1.5	Security and Privacy	11
1.5.1	Security Considerations	11
1.5.2	Privacy Considerations.....	11
1.6	Semantic Composition.....	11
1.6.1	Conformance with Energy Interoperation.....	12
1.6.2	Conformance with EMIX.....	12
1.6.3	Conformance with WS-Calendar Streams	12
1.6.4	Compatibility with Facilities Smart Grid Information Model	12
2	Overview of Common Transactive Services	14
2.1	Scope of Common Transactive Services	14
2.1.1	Applicability to Microgrids (Informative)	14
2.1.2	Specific scope statements.....	14
2.1.3	Resources, Products and Instruments	14
2.2	Common Transactive Services Architecture	16
2.2.1	Facets in CTS.....	16
2.2.2	Sides in Tenders and Transactions.....	17
2.2.3	Party and Counterparty in Tenders and Transactions	18
2.2.4	Responses.....	18
3	Common Semantic Elements of CTS.....	20
3.1	Semantic Elements from WS-Calendar	20
3.2	Semantic Elements from EMIX.....	20
3.2.1	Defining Resource	21
3.2.2	Defining Product.....	21
3.2.3	Market Characteristics from EMIX.....	22
4	Basic Interaction and Terminology	24
4.1	Structure of Common Transactive Services and Operations	24
4.2	Naming of Services and Operations	24
4.3	Payloads and Messages.....	24
4.4	Description of the Facets and Payloads	24
4.5	Responses	25
5	CTS Streams	27
5.1	CTS Streams Introduction	27
5.2	Information Model for CTS Streams	27
6	Market Facet.....	29
6.1	Market Context History	29
6.2	Interaction Pattern for the Market Facet	29
6.3	Information Model for the Market Facet.....	30
7	Party Registration Facet.....	33

8	Tender Facet	34
8.1	Tenders as a Pre-Transaction Payloads	34
8.2	Interaction Patterns for the Tender Facet	34
8.3	Information Model for the Tender Facet	35
8.4	Payloads for the Tender Facet	38
9	Transaction Facet.....	40
9.1	Transaction Services	40
9.2	Interaction Pattern for the Transaction Facet	40
9.3	Information Model for the Transaction Facet.....	40
9.4	Payloads for the Transaction Facet.....	42
9.5	Comparison of Transactive Payloads	42
10	Position Facet.....	44
10.1	Introduction	44
10.2	Position Definition	44
10.3	Interaction Pattern for the Position Facet	44
10.4	Information Model for the Position Facet.....	45
10.5	Payloads for the Position Facet	46
11	Delivery Facet.....	48
11.1	Interaction Pattern for the Delivery Facet	48
11.2	Information Model for the Delivery Facet.....	49
11.3	Payloads for the Delivery Facet.....	50
12	Market Information Facet—Quote and Ticker	52
12.1	Quotes	52
12.1.1	Interaction Pattern for Market Information Facet Quote	53
12.1.2	Information Model for the Quote	53
12.2	Tickers	53
12.2.1	Information Model for the Market Information Facet Ticker	54
12.2.2	Interaction Pattern for the Market Information Facet Ticker.....	54
13	Bindings.....	55
13.1	JSON	55
13.2	XML Schema	55
13.2.1	XML Namespaces	55
13.3	Simple Binary Encoding.....	55
14	Conformance	56
14.1	Introduction to Conformance	56
14.2	Claiming Conformance to Common Transactive Services.....	56
	Appendix A. References	57
	A.1 Normative References.....	57
	A.2 Informative References	58
	Appendix B. Security and Privacy Considerations.....	60
	B.1 CTS and Security Considerations	60
	B.2 CTS and Privacy Considerations	60
	Appendix C. Glossary of Terms and Abbreviations Used in this document	62
	Appendix D. Acknowledgments	63
	D.1 Participants.....	63

Appendix E. Revision History.....	64
Notices	65

Table of Tables

Table 2-1: Definitions used in CTS Markets	15
Table 2-2: Transactive Facets Defined in CTS.....	17
Table 2-3: Responses	18
Table 3-1: CTS Elements from WS-Calendar	20
Table 3-2: Defining the Resource	21
Table 3-3: Defining the Product	21
Table 3-4: Market-related elements from EMIX	22
Table 3-5: Standard Terms that define market interactions.....	22
Table 5-1: CTS Stream Attributes	28
Table 5-2: Stream Interval Attributes	28
Table 6-1: Standard Terms returned by Market Facet	30
Table 6-2: Elements that define Products in a Market or Marketplace	32
Table 8-1: Pre-Transaction Tender Facets	34
Table 8-2: EiTender Attributes	36
Table 8-3 EiCreateTenderPayload Attributes.....	39
Table 9-1: Transaction Management Service	40
Table 9-2: EiTransaction Attributes.....	41
Table 10-1: Position Facet	44
Table 10-2: Attributes of Position Facet Payloads	46
Table 11-1: Delivery Facet	48
Table 11-2: Attributes of Delivery Facet Payloads.....	50
Table 12-1: Quotation Facet	53
Table 12-2: Market Information Facet Ticker	54
Table C--1 Abbreviations and Terms used throughout this document	62

Table of Figures

Figure 4-1: Example of generic error response for a service operation.....	25
Figure 5-1: CTS Stream Definition.....	27
Figure 6-1: UML Sequence Diagram for the Market Facet.....	29
Figure 6-2: UML of Market Facet payloads	30
Figure 8-1: UML Sequence Diagram for the Tender Facet.....	35

Figure 8-2: Class EiTender	35
Figure 8-3: Enumeration TransactiveState	37
Figure 8-4: UML Class Diagram for the Operation Payloads for the EiTender Facet	38
Figure 9-1: UML Sequence Diagram for the EiTransaction Facet.....	40
Figure 9-2: UML Class Diagram of EiTransaction	41
Figure 9-3: UML Class Diagram of EiTransaction Facet Operation Payloads	42
Figure 9-4: UML Diagram comparing Tender and Transaction Facet Payloads.....	43
Figure 10-1: UML Sequence Diagram for the Position Facet	45
Figure 10-2: UML Class Diagram of Payloads for the Position Facet	46
Figure 11-1: UML Sequence Diagram for the Delivery Facet	49
Figure 11-2: UML Class Diagram of Payloads for the Position Facet	50
Figure 12-1: UML Sequence Diagram for the Quotation Facet	53
Figure 12-2 Market Information Facet Ticker Sequence Diagram.....	54

1 Introduction

The Common Transactive Services (CTS) enable actor interaction with any resource market.

CTS is an application profile of the OASIS Energy Interoperation 1.0 ([EI]) specification, with most optionality and complexity stripped away. CTS defines messages for a transactive energy profile specification, leaving communication details unspecified. The purpose of CTS is to enable broad semantic interoperation in multiple environments.

Transactive resource management coordinates resource supply and use between any two Parties using markets that trade instruments based on time. Transactive energy applies Transactive Resource Management [TRM] to energy markets.

TRM is a means to allocate resources including the delivery of commodities including but not limited to electrical energy, electrical power, natural gas, and thermal energy such as steam, hot water, or chilled water. The initial research in TRM used a market to allocate heat from a single furnace within a commercial building. A resource is defined as a tradable commodity whose value depends on price, location, and time of delivery [EMIX]. TRM balances supply and demand over time using automated voluntary transactions between market participants.

TRM applied to the distribution of power or energy is referred to as Transactive Energy (TE). The resource managed is energy or power, and the transmission rights to move these, or in order to maintain grid frequency and voltage.

The essence of TE is that an energy transaction for delivery of a quantity of an energy product during a time period at a location creates a position. This position may then be modified by additional buy and sell transactions. TE requires no information exchange other than that needed to offer and execute energy transactions.

The simplest application of TE assumes a constant flow of power for a known period of time. It is rarely acceptable to deliver all of a three-hour commitment during the initial five minutes. Delivery is historically measured as net energy at the end of an interval. The differences are computationally trivial, but may add system complexity.

Neither EI nor CTS specifies which technologies participants will use; rather CTS defines a technology-agnostic minimal set of messages to enable interoperation through markets of participants irrespective of internal technology. In a similar manner, CTS does not specify the internal organization of a market, but rather a common set of messages that can be used to operate any transactive energy market. The goal of CTS is to enable systems and devices developed today or in the future to participate in markets deployed today or in the future. The reader can find an extended discussion of Transactive Energy (TE) in the EI specification.

Autonomous market actors must be able to recognize patterns and make choices to best support their own needs. Actors need not share details of their internal operations with others.

CTS is a lightweight profile of the OASIS Energy Interoperation to support an actor model. An essential aspect of the actor model is to use a limited number of simple messages, with each message strongly typed. All CTS messages are simple and make no assumptions about the systems behind the messages.

1.1 Application of the Common Transactive Services

The purpose of this specification is to codify the common interactions and messages required for energy markets. Any system able to use CTS should be able to interoperate with any CTS-conforming market with minimal or no change.

Systems that can be represented by CTS actors include but are not limited to

- Smart Buildings/Homes/Industrial Facility
- Building systems/devices
- Business Enterprises
- Vehicles

- Microgrids
- Collections of IoT (Internet of Things) devices

TE demonstrations and deployments have seldom been interoperable—each uses its own message model and its own market dynamics. Many early implementations required the use of remote or cloud-based markets. Such markets discount local decision making while introducing new barriers to resilience such as network failure. Others rely on a single price-setting supplier. None are interoperable either at the system level or for the actors involved.

CTS is valuable for creating micromarkets **[Micromarkets]** to manage power within microgrids. Micromarkets support the capability for dynamic restructuring of grids for fault resilience and efficiency **[GridFaultResilience]**. CTS limits complexity by abstracting market interactions to the few common messages of CTS within a bounded scope.

A device, building, market, or microgrid implementing CTS can exchange information with any other market or system using CTS, meaning that an application need not be reimplemented or tailored to different CTS-enabled markets.

CTS does not presume a market with a single seller (e.g., a utility). CTS recognizes two parties to a transaction, and the role of any Party can switch from buyer to seller from one transaction to the next. Each Resource Offer (Tender) has a Side attribute (Buy or Sell). when each transaction is committed (once the product has been purchased) it is owned by the purchaser, and it can be re-sold as desired or needed.

A CTS-operated micromarket may balance power over time in a traditional distribution system attached to a larger power grid or it may bind to and operate a stand-alone autonomous microgrid **[SmartGridBusiness]**.

1.2 Support for Developers

The Common Transactive Services are defined in XML schemas **[XSD]** and described using Universal Modelling Language **[UML]**. Many software development tools can accept artifacts in UML or in XSD to enforce proper message formation.

This specification also provides **[JSON]** schemas compatible with JSON Abstract Data Notation **[JADN]** format.

The FIX Simple Binary Encoding **[SBE]** specification is used in financial markets. SBE is designed to encode and decode messages using fewer CPU instructions than standard encodings and without forcing memory management delays. SBE-based messaging is used when very high rates of message throughput are required. This specification will deliver schemas for generating SBE messages based on the common message content.

1.3 Naming Conventions

This specification follows some naming conventions for artifacts defined by the specification, as follows:

For the names of elements and the names of attributes within XSD files and UML models, the names follow the lowerCamelCase convention, with all names starting with a lower-case letter. For example,

```
<element name="componentType" type="ei:ComponentType"/>
```

For the names of types within XSD files, the names follow the UpperCamelCase convention with all names starting with an upper-case letter prefixed by "type-". For example,

```
<complexType name="ComponentServiceType">
```

For clarity in UML models the suffix "type" is not always used.

For the names of intents and for attributes in the UML models, names follow the lowerCamelCase convention, with all names starting with a lower-case letter, EXCEPT for cases where the intent represents an established acronym, in which case the entire name is in upper case.

JSON and where possible SBE names follow the same conventions.

1.4 Editing Conventions

For readability, element names in tables appear as separate words. The actual names are lowerCamelCase, as specified above, and as they appear in the UML models, and in the XML and JSON schemas.

All elements in the tables not marked as “optional” are mandatory.

Information in the **Meaning** column of the tables is normative. Information appearing in the **Notes** column is explanatory and non-normative.¹

Examples and Appendices are non-normative.

1.5 Security and Privacy

Service requests and responses are generally considered public actions of each interoperating system, with limitations to address privacy and security considerations (see Appendix B). Service actions are independent from private actions behind the interface (i.e., device control actions). A service is used without needing to know all the details of its implementation. Consumers of services generally pay for results, not for effort.

1.5.1 Security Considerations

Loose integration using the service-oriented architecture (SOA) style assumes careful definition of security requirements between partners. Size of transactions, costs of failure to perform, confidentiality agreements, information stewardship, and even changing regulatory requirements can require similar transactions be expressed within quite different security contexts. It is a feature of the SOA approach that security is composed in to meet the specific and evolving needs of different markets and transactions. Security implementation is free to evolve over time and to support different needs. The Common Transactive Services allow for this composition, without prescribing any particular security implementation.

1.5.2 Privacy Considerations

Detailed knowledge of offers to buy or sell or of energy inputs and outputs for an actor may reveal information on actions and operations.

For example, transactions or tenders may indicate whether a production line is starting or stopping, or anticipated energy needs, or who has been buying or selling power. Making such information public may be damaging to actors.

Similarly, an adverse party may be able to determine the likelihood that a dwelling is presently occupied.

Both security and privacy considerations are addressed in Appendix B.

1.6 Semantic Composition

The semantics and interactions of CTS are selected from and derived from [EI].

Energy Interoperation references two other standards, [EMIX] and [WS-Calendar], and uses an earlier Streams definition. We adapt, update, and simplify the use of the referenced standards, while maintaining conformance.

- EMIX describes price and product for electricity markets.
- WS-Calendar communicates schedules and sequences of operations. CTS uses the [Streams] optimization which is a standalone specification, rather than part of Energy Interoperation 1.0.

¹ In ISO and IEC standards, portions that are not normative are *informative*. OASIS uses the term *non-normative*.

- Energy Interoperation uses the vocabulary and information models defined by those specifications to describe the services that it provides. The payload for each Energy Interoperation service references a product defined using **[EMIX]**. EMIX schedules and sequences are defined using **[WS-Calendar]**. Any additional schedule-related information required by **[EI]** is expressed using **[WS-Calendar]**.
- Since **[EI]** was published, a semantically equivalent but simpler **[Streams]** specification was developed in the OASIS WS-Calendar Technical Committee. CTS uses that simpler **[Streams]** specification.

All terms used in this specification are as defined in their respective specifications.
Assumptions

1.6.1 Conformance with Energy Interoperation

OASIS Energy Interoperation **[EI]** defines an end-to-end interaction model. Energy Interoperation Transactive Services is the basis for CTS, which draws definitions of parties and transactive interactions primarily from the EI TEMIX profile.

This specification can be viewed as a minimal transactive profile of **[EI]**

1.6.2 Conformance with EMIX

This specification uses a simplified profile of the models and artifacts defined in OASIS Energy Market Information Exchange **[EMIX]** to communicate product definitions, quantities, and prices. EMIX provides a succinct way to indicate how prices, quantities, or both vary over time.

The EMIX product definition is the Transactive Resource in CTS 1.0.

EMIX also defines Market Context, a URI used as the identifier of the Market. EMIX further defines Standard Terms as retrievable information about the market that an actor can use to configure itself for interoperation with a given market. We extend and clarify those terms, provide an extension mechanism, and discuss the relationship of markets, marketplaces, and products.

1.6.3 Conformance with WS-Calendar Streams

WS-Calendar expresses events and sequences to support machine-to-machine (M2M) negotiation of schedules while being semantically compatible with human schedules as standardized in **[iCalendar]**. Schemas in **[WS-Calendar]** support messages that are nearly identical to those used in human schedules. We use a conformant but simpler and more abstract Platform Independent Model **[CAL-PIM]** and the **[Streams]** compact expression², to support telemetry (Delivery Facet) and series of Tenders.

By design and intent, the **[WS-Calendar]** schemas provide the capability of mapping between human and M2M schedules.

WS-Calendar conveys domain specific information in a per-event payload. An essential concept of WS-Calendar is inheritance, by which a starting time can be applied to an existing message, or by which all events in a sequence share common information such as duration. Inheritance is used to “complete” a partial message during negotiation. CTS makes use of this to apply common market product across a sequence, or to convey a specific starting time to a market product.

CTS messages conform to **[Streams]** format. See also Section 3.1.

1.6.4 Compatibility with Facilities Smart Grid Information Model

The Facilities Smart Grid Information Model **[FSGIM]** was developed to define the power capabilities and requirements of building systems over time. FSGIM addresses the so-called *built environment* and uses the semantics of WS-Calendar and EMIX to construct its information models for **[power]** use over time. These sequences of **[power]** requirements are referred to as load curves. Load curves can potentially be

² Simplified as CTS Streams in this specification.

177 relocated in time, perhaps delaying or accelerating the start time to get a more advantageous price for
178 [power].

179 Because FSGIM load curves use the information models of EMIX and WS-Calendar, conforming load
180 curves submitted by a facility could be the basis upon which a TE Agent would base its market decisions.

181 The Architecture of CTS is premised on distinct physical systems being able to interoperate by
182 coordinating their production and consumption of energy irrespective of their ownership, motivations, or
183 internal mechanisms. This specification defines messages and interactions of that interoperation.

184 CTS tenders and transactions can be used to express FSGIM load requests. CTS 1.0 uses single-
185 interval **[Streams]** to express single-interval tenders in anticipation of possible future use of Streams in
186 FSGIM-conformant communications.

2 Overview of Common Transactive Services

2.1 Scope of Common Transactive Services

CTS engages Transactive Resources, e.g. Distributed Energy Resources (DER), as well as any provider or consumer of energy, while making no assumptions as to their internal processes or technology.

This specification supports agreements and transactional obligations, while offering flexibility of implementation to support specific approaches and goals of the various participants.

No particular agreements are endorsed, proposed or required in order to implement this specification. Energy market operations are beyond the scope of this specification although interactions that enable management of the actual delivery and acceptance are within scope but not included in CTS 1.0.

As shown in [CTS2016] the Common Transactive Services with suitable product definitions can be used to communicate with essentially any market.

2.1.1 Applicability to Microgrids (Informative)

As an extended example, using the Common Transactive Services terminology, a microgrid is comprised of interacting nodes each represented by an actor (interacting as CTS parties). Those actors interact in a micromarket co-extensive in scope with the microgrid. No actor reveals any internal mechanisms, but only its interest in buying and selling power.

CTS can also be used for the fractal integration of microgrids. Any micromarket can be bound to or co-extensive with a node in a larger microgrid. A micromarket participating in this way exposes only its aggregate market position. Any participant in CTS effectively aggregates resources it logically contains.

Any participant in the original micromarket MAY itself represent a contained autonomous microgrid or any autonomous entity whether or not it is managed in turn by a market.

[StructuredEnergy][SmartGridBusiness]

2.1.2 Specific scope statements

Interaction patterns and facet definitions to support the following are in scope for Common Transactive Services:

- Interaction patterns to support transactive energy, including tenders, transactions, and supporting information
- Information models for price and product communication
- Information models for market characteristics
- Payload definitions for Common Transactive Services

The following are out of scope for Common Transactive Services:

- Requirements specifying the type of agreement, contract, product definition, or tariff used by a particular market.
- Computations or agreements that describe how power is sold into or sold out of a marketplace.
- Communication protocols, although semantic interaction patterns are in scope.

This specification describes standard messages, the set of which may be extended.

2.1.3 Resources, Products and Instruments

Systems use the common transactive services to operate transactive resource markets. A transactive resource market balances the supply of a resource over time and the demand for that resource by using a market specifying the time of delivery.

See Section 3.2 for formal definitions.

228 We define a Resource as any commodity whose value is determined by time of delivery. Transactable
 229 resources include, but are not limited to, energy, heat, natural gas, water, and transport as a support
 230 service for these. The ancillary services reactive power, voltage control, and frequency control are also
 231 transactable.

232 A Product names a transactive resource that has been “chunked” for market. These chunks define the
 233 market granularity in quantity and in time. For example, the product may be 1 MW of power delivered over
 234 an hour. Similarly, another Product may be 1 kW of power over a 5-minute period. Some transactive
 235 energy markets in North America today have durations as brief as two seconds. Temporal granularity is
 236 equally important as quantity for product definition.

237 An Instrument is a Product at a specific time. For example, the 1 MW of Power delivered over an hour
 238 beginning at 3:00 PM is a different Instrument than the same Product delivered starting at 11:00 PM. We
 239 use the semantics from financial markets to name the thing that is bought or sold is an Instrument.

240 A market considers all the tenders it has received offering to buy or sell an Instrument, using a Matching
 241 Engine to decide which can be cleared (*satisfied*) in full or in part. The 3:00pm instrument is traded
 242 independently from the 4:00pm instrument. This specification does not assume or require an Order Book,
 243 a Double Auction, or another mechanism in the Matching Engine.

244 The Resource definition is extensible using standard UML techniques (subclassing); however CTS 1.0
 245 uses only this base definition.

246 In future versions of CTS may permit any conforming resource definition to be used to define Products
 247 that can be traded using CTS.

248 These terms are summarized in Table 2-1: .

249 *Table 2-1: Definitions used in CTS Markets*

Transactive Entity	Definition
Resource	A measurable commodity, substance, service, or force, whose value is determined by time of delivery
Product	A Resource defined by size/granularity of the Resource and by the granularity of time. A market is defined by its product. Example 1: electric power in 10 kW units delivered over an hour of time. Example 2: electric energy in 1 kWh units delivered over a half hour.
Instrument	A Product instantiated by a particular begin time. Example: the Product beginning at 9:00 AM on April 3. An instrument is tendered to a market with specific quantity and price.
Party	A Party is an Actor that buys or sells Instruments in a CTS Marketplace. A Party may be described by a specific role in a specific interaction, such as Party or Counter Party. For semantic and privacy issues, see Section 2.2.3 below.
Market	Where Products are traded based on tenders submitted to buy or sell an Instrument
Marketplace	An actor wherein one or more Markets are conducted

Transactive Entity	Definition
Market Context	<p>In EMIX, the Market Context is a URI identifying a Marketplace. In CTS, the Market Context SHOULD be resolvable and available so an Actor can retrieve machine-readable information describing a Marketplace. Examples of information that might be associated with an EMIX Market Context include:</p> <ul style="list-style-type: none"> • A list of Products traded in this Marketplace • Specific details of market operation (e.g., rules for registration and qualification, product quality, penalties for non-delivery, etc.) • Currency used for market transactions
Matching Engine	<p>There are many market processes to exchange offers and reach agreements on transactions. Different parts of the same marketplace MAY employ different market processes. We term each of these processes a Matching Engine. The specific processes, structure, and algorithms of Matching Engines are out of scope.</p>

2.2 Common Transactive Services Architecture

The implied CTS architecture is drawn from and is a subset and simplification of the architecture presented in [EI]. Specifically, the Energy Interoperation architecture uses the Service-Oriented Architecture (SOA) model which has become the consensus view for energy-related interoperation. CTS refines and simplifies this to an Actor model.

The **[Actor Model]** names a style of system integration used for high scalability and resilience. The Actor Model uses a small number of simple messages to coordinate behavior among simple agents termed Actors. The Actor Model accomplishes complex behaviors through the fabric that hosts the Actors. This specification makes no assumptions about this fabric. Note that systems represented by Actors need not be actually simple; any modern facility incorporates a number of complex energy systems. This complexity is encapsulated within the Actors and the interactions are reduced to simple messages.

It is important to understand that an Actor may take on roles for its TE-related messages. In a Tender or Transaction, one Actor is the Party, the other is the Counterparty.

The Common Transactive Services are a lightweight profile of the OASIS Energy Interoperation specification, simplified into Actor-to-Actor messages. Each CTS message is simple and makes no assumptions about the systems behind the messages. The market receives tenders and announces contracts. Only the simple messages of CTS are used.

CTS is agnostic about how CTS messages are transported. In distinction, [EI] specifies transport (e.g. XML-based SOAP message exchanges). CTS messages may be thought of as the information exchange in a Service-Oriented Architecture environment, with the same implied message patterns.

Just as the market participants present simple messages, so too, does the market. The internals of a market contain a Matching Engine to match tenders and to declare contracts. The rules used to match tenders could be a continuously clearing order book, or a periodic double auction, or some other model. This complexity is hidden from the Actors.

2.2.1 Facets in CTS

Nearly all interactions implied in CTS (and described as payloads) are as defined in [EI]. That specification defines contracts between systems as services with defined messages and interactions.

This specification describes these roles taken on by actors as *facets* for that Actor, each distinct from other roles the Actor may perform. The facets are named and briefly described in Table 2-2. Each facet

includes several messages, as in submitting a Tender, acknowledging a Tender, and cancelling a Tender. Those familiar with [EI] will recognize that each facet is mappable to an EI service.

Each facet is discussed in detail starting in Section 5

Table 2-2: Transactive Facets Defined in CTS

Facet	Definition
Market	A Party to potential transactions needs to know what Products are traded in a Marketplace, the granularity (size and time and price), and other Marketplace information. When a Marketplace includes multiple products, the Party needs to know where to find the Market for each Product. While moving slowly over time, this can generally be viewed as static information about the Marketplace and its Products and Markets.
Tender	A Tender is an actionable offer to buy or to sell an instrument at a given price. Tenders go to the market and are generally private. It is possible to request that a Tender be advertised to all Actors in the Marketplace.
Transaction	A Transaction is created by the Market to record a contract when a tender to buy and a tender to sell are matched. Both parties are notified of contract creation.
Position	At any moment, a Party has a position which represents the cumulative amount of an Instrument that an actor has previously transacted for within a bounding time interval. A Position for an Instrument reflects the algebraic sum of all quantities previously bought or sold. See Section 10 Position Facet.
Delivery	<p>After the Product as represented by an Instrument is sold and delivered, in many system implementations there is an asynchronous validation of what was consumed or delivered, that it might be compared with what was purchased or sold.</p> <p>It is simplest to think of Delivery as a meter reading, although that meter may be virtual or computed. See Section 11 Delivery Facet.</p>
Quote	<p>A Quote is a non-actionable indication of a potential price or availability of an instrument. Different Markets may restrict which actors may issue Quotes, say from only Market Agents or from External Actors.</p> <p>[EI] defines the EiQuote service.</p>

Each of these facets includes multiple messages which are described starting in Section 4. Sometimes one facet precedes the use of another facet, as Tenders may initiate messages for the Transaction Facet.

2.2.2 Sides in Tenders and Transactions

A Party can take one of two Sides in a given Transaction:

- Buy, or
- Sell

A Party selling [an Instrument] takes the Sell Side of the Transaction. A Party buying [an Instrument] takes the Buy Side of the Transaction. The offering Party is called the Party in a Transaction; the other Party is called the Counterparty

From the perspective of the market, there is no distinction between a Party selling additional power and party selling from its previously acquired position. An Actor representing a generator would generally take the Sell side of a transaction. An Actor representing a consumer generally takes the Buy side of a transaction.

However, a generator may take the Buy Side of a Transaction in order to reduce its own generation, in response either to changes in physical or market conditions or to reflect other commitments made by the actor.

A consumer may choose to sell from its current position if its plans change, or if it receives an attractive price. A power storage system actor may choose to buy or sell from interval to interval, consistent with its operating and financial goals.

We do not specify how the [Product related to the Instrument] is delivered.

2.2.3 Party and Counterparty in Tenders and Transactions

Which Party or Parties should be included in a Tender or Transaction payload? Who needs to know and be able to track a reference?

The Party in a Tender is offering to buy or sell.

Delegation may involve a sender (a delegate) that is not the party that is buying or selling. The *PartyID* should always reference the party that is tendering.

The Counterparty for a tender may reference either

- 1) The Market itself, or
- 2) A specific Party to which the Tender is made

The former suggests a market tender where the market will match tenders and create Transactions. The latter suggests a bilateral interaction not necessarily involving a market. Note that the behavior of the Actor creating a tender is the same, as the process to determine the Counterparty is not in scope.

In market interactions, the Counterparty SHOULD be the Party ID for the Market as described by the Market Characteristics Facet (and consistent with that described by the Market Characteristics Facet for a specific market). This value is accessible via the Market Characteristics Facet.

When a Transaction is created, a contract is created between the buyer and the seller.

2.2.4 Responses

This section re-iterates terms and simplifies models from [EI]. That specification is normative. The response types are common across all message categories.

Table 2-3: Responses

Attribute	Meaning
Request ID	A reference ID which identifies the artifact or message element to which this is a response. The Request ID uniquely identifies this request, and can serve as a messaging correlation ID ³ .

³ As an example of the *Correlation Pattern* for messages

Attribute	Meaning
Response Code	<p>The Response Code indicates success or failure of the operation requested. The Response Description is unconstrained text, perhaps for use in a user interface.</p> <p>The code ranges are those used for HTTP response codes,⁴ specifically</p> <p>1xx: Informational - Request received, continuing process</p> <p>2xx: Success - The action was successfully received, understood, and accepted</p> <p>3xx: Pending - Further action must be taken in order to complete the request</p> <p>4xx: Requester Error - The request contains bad syntax or cannot be fulfilled</p> <p>5xx: Responder Error - The responder failed to fulfill an apparently valid request</p>

323 The column labeled *Response* lists the name of the service operation payload (in Energy Interoperation
324 and its TEMIX profile, this includes the service operation as well) invoked in response. Most operations
325 have a response. The roles of *Service Consumer* and *Service Provider* are reversed for the *Response*.

⁴ See e.g. https://en.wikipedia.org/wiki/List_of_HTTP_status_codes

3 Common Semantic Elements of CTS

The messages of CTS use a few common elements. These elements are derived from definitions in [WS-Calendar], [EMIX], and in [EI].

3.1 Semantic Elements from WS-Calendar

Time and Duration are the essential elements of defining an instrument as well as for interacting with a market. A Stream **[Streams]** is a series of back-to-back intervals each with its own associated information. In Section 5, a *CTS Stream* is defined as a conformant specialization of **[Streams]**, integrating information that is outside of a Stream data structure but associated with a Stream.⁵

Table 3-1: CTS Elements from WS-Calendar

Attribute	Meaning
Duration	Duration is used to define Products, as in “Power can be purchased and there is a one-hour (duration) market for Power”. Duration is also used in Delivery to specify the period over which Delivery is measured, as in “How much Power was delivered in the 4 hours beginning with the Begin Date-Time.
Offset	An offset (expressed as a WS-Calendar Duration) that some markets MAY use to transfer trading off of hourly boundaries. A power distribution entity may experience disruption if there is a big price change on the hour. Offset enables a market to trade, for example, 3 minutes after the hour. See also Market Facet
Begin Date-Time	Begin Date-Time fully binds a Duration into an Interval. When applied to a Product, the Begin Date-Time defines an Instrument., i.e., something that is directly traded in the Market.
Expiration Date-Time	Expiration is used to limit the time a Tender is on the Market. There is an implicit expiration for every Tender equal to the Begin Date-Time of the instrument. Expiration Date-Time is needed only if the requested Expiration is prior to the Begin Date-Time of the Instrument.

3.2 Semantic Elements from EMIX

EMIX defines what is sold in a market, when it is sold, what the units are, what the standard trade size is, EMIX refers to this as the Item. EMIX also describes how the price for an Item varies over time. In CTS, we refactor the Item into the Resource (what is sold), the Product (how much of a Resource is sold and for how long), and the Instrument (a Product sold at a specific time).

CTS Markets consist of offers (Tenders) to buy and sell these Instruments.

⁵ Including Resource Designator, Stream Start, and Decimal Fraction

3.2.1 Defining Resource

Each Resource in a marketplace must be traded in a contained market. A given marketplace MAY have multiple products based on the same resource.

Table 3-2: Defining the Resource

Attribute	Meaning
Resource	Abstract base for describing all Resources. A Resource consists of a Designator, Name and a Description.
Item Description	The Item Description is a common name, as defined in EMIX
Item Unit	Item Unit is the unit of measure for the Resource.
Attributes	Optional elements that further describe the Resource, as in hertz and voltage

3.2.2 Defining Product

The product completes the re-factoring of the EMIX Item, adding the size and duration to a Resource

Table 3-3: Defining the Product

Attribute	Meaning
Product	Abstract Base for all defining all Products. The core of each Product is the Resource, as referenced by the Resource Designator.
Scale	Mantissa that specifies the size of the Resource Unit. For example, a Product denominated in megawatts has a mantissa of 6.
Size	An integer “chunking” the Product, i.e., the Product could be traded in units of 5 kW, a size of 5 and a scale of 3.
Warrant	Undefined element of a product that restricts the product beyond the Resource definition. For example, it is possible to trade in power designated to be Neighborhood Solar Power so long as the Product clears, that is, delivery is taken in the same interval as it is bought.

Products with differing Warrants are different Products.

For example, if an Actor wishes to buy energy with a *Green Warrant* (however defined) then the Actor is responsible for defining its trading strategies to buy the un-warranted Product if the warranted Product is not available.

As a further application example, Actors that wish to buy or sell Neighborhood Solar Power are responsible for submitting Tenders that expire in time to make alternate arrangements, or in cancelling Tenders before fulfillment.

Market implementers should consider carefully whether they wish to support Warrants, as excessive segmentation will lead to markets that to shallow for effective trading. Warrants add additional complexity of definition, i.e. such questions as “Is a Battery which stores power generated by Neighborhood Solar Power considered to be selling Neighborhood Solar Power when it discharges?” Alternately, if a market rule requires a Solar Panel to purchase a policy from other sources to insure its capability of Delivery, is that power considered Neighborhood Solar Power? This and similar questions would introduce the type of complexity that violates the design principles of CTS. Such complexity may also reduce interoperability of commodity Actors with specific Markets.

Warrants were defined in EMIX, and are permitted in CTS to support this complexity if desired.

3.2.3 Market Characteristics from EMIX

EMIX defines vocabulary used in market messages and interactions, which we simplify and extend. The CTS Market Facet is described in Section 6.

Table 3-4: Market-related elements from EMIX

Attribute	Meaning
PartyID	The market-based ID of an actor participating in a Market, particularly the actor originating a Tender, Quote, or Contract.
Counter PartyID	The market-based ID of an actor participating in a Market, particularly the actor taking the other side of a contract from the Party. See Section 2.2.3.
Side	An indication of what a Party offers in a tender or other message, i.e., “Buy” or “Sell”.
Expiration Date-Time	Expiration is used to limit the time a Tender is on the Market. There is an implicit expiration for every Tender equal to the Begin Date-Time of the instrument. Expiration Date-Time is needed only if the requested Expiration is prior to the Begin Date-Time of the Instrument.
Market Context	In EMIX, the Market Context is simply a URI to name a market, and need not be resolvable. CTS distinguishes between a Marketplace, where many products may be sold and the Market, where a specific Product is sold. See Section 6. “Market Facet”.
Standard Terms	Standard Terms are the machine-readable information about a Marketplace or Market, and the interactions it supports. In CTS, the Standard Terms include an enumeration of the Products and their respective Markets tradable in this Marketplace. See Section 6, “Market Facet”.

EMIX does not define how an Actor discovers the Standard Terms in a Marketplace. CTS defines the Market Facet to discover and expose Products and Standard Terms.

Table 3-5: Standard Terms that define market interactions

Attribute	Meaning
Market Context Name	Text providing a descriptive name for a Marketplace. While the Name MAY be displayed in a user interface, it may not be meaningful to the Actors.
Currency	String indicating how value is denominated in a market. If fiat currency, should be selected from current codes maintained by UN CEFACT. May also be cryptocurrencies or local currency.

Attribute	Meaning
Time Offset	A Duration that some markets MAY use to describe trading off of hourly boundaries. A power distribution entity may experience disruption if there is a big price change on the hour. For example, a distribution system operator (DSO) that operates multiple CTS markets could opt to set a different offset on each Market operated out of a given substation. In this model, a Marketplace could use an offset duration of 3 minutes to indicate that all tenders are based on three minutes after the hour.
Time Zone	A Time Zone indicates how all Times and Dates are expressed. The Marketplace Time Zone is a Standard Term.
Terms	<p>EMIX Terms are extrinsic to the product delivery but affect how each party interacts with others including a Market.</p> <p>Terms may be tied to basic operational needs, or state schedules of availability, or suggest limits on bids and prices acceptable. See <i>Section 6, "Market Facet"</i>.</p>
Products	The Products traded in this Marketplace. Note that similar products with and without Warrants are different products, each traded in their own Market.

4 Basic Interaction and Terminology

4.1 Structure of Common Transactive Services and Operations

The Common Transactive Services presented in this specification are described in the following sections, and are

- Market Facet—characteristics and to know what products and instruments can be traded
- Party Registration Facet—identification of actors within a market or marketplace
- Tender Facet—make offers to buy and sell products
- Transaction Facet—for expressing transactions (contracts)
- Position Facet—Describe what has been previously bought or sold
- Delivery Facet—Request data on actual deliveries
- Market Information Facet—Quotes and market Tickers

We include UML definitions for the standard payloads for service requests, rather than the service, communication, or other characteristics. In Section 13 we describe standard serialization for the CTS standard payloads; additional bindings may be used by conforming implementations.

4.2 Naming of Services and Operations

The naming of services and operations and service operation payloads follows the pattern defined in [EI]. Services are named starting with the letters **Ei** following the Upper Camel Case convention. Operations in each service use one or more of the following patterns. The first listed is a fragment of the name of the initial service operation; the second is a fragment of the name of the response message which acknowledges receipt, describes errors, and may pass information back to the invoker of the first operation.

Create—Created An object is created and sent to the other Party

Cancel—Canceled A previously created request is canceled

For example, to construct an operation name for the Tender facet, "Ei" is concatenated with the name fragment (verb) as listed. An operation to cancel an outstanding Tender is called *EiCancelTender*.⁶

Facets describe what would be called services in a full Service-Oriented Architecture implementation, as we do not define SOA services, but only imply and follow a service structure from [EI].

4.3 Payloads and Messages

We define only the payloads; the particular networking technique and message structure is determined by the applications sending and receiving CTS payloads.

While the payloads are logically complete with respect to the SOA interactions in [EI], the payloads may be exchanged by any means; such exchanges are below the semantic level of this specification.

4.4 Description of the Facets and Payloads

The sections below provide the following for each service:

- Facet description

⁶ This pattern was developed and is used by IEC Technical Committee 57 (Power Systems).

- Table of Payloads
- Interaction patterns for payload exchange in graphic form, using Energy Interoperation normative interactions and UML Sequence Diagrams [UML].
- Normative information model using [UML] for key artifacts used by the facet
- Normative operation payloads using [UML] for each interaction

4.5 Responses

Responses may need to be tracked to determine whether an operation succeeds or not. This may be complicated by the fact that any given transaction may involve the transmission of one or more information objects.

An EiResponse returns the success or failure of the entire operation, with possible detail included in responseTermsViolated (see Section 5).

It is MANDATORY to return responses.⁷ Indicating partial or complete success or failure.

The class diagram in Figure 4-1 shows the generic CTS response.

CTS uses EiResponseType is from Energy Interoperation, changing only the cardinality of responseDescription (to zero, that is, not passed).

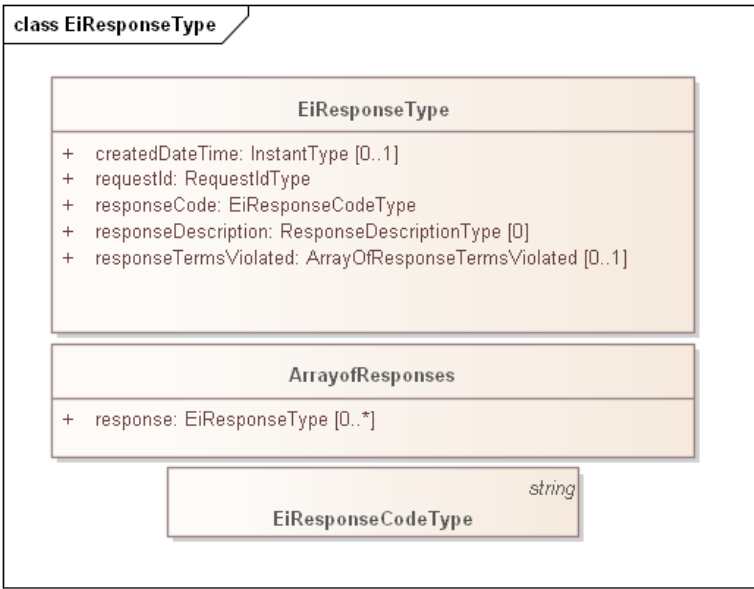


Figure 4-1: Example of generic error response for a service operation

There is no exhaustive list of all possible Response Codes. More detail on Response Codes is in Section 2.2.4.

The Response Codes are intended to enable even the smallest device to interpret Response. This specification uses a pattern consisting of a 3-digit code, with the most significant digit sufficient to interpret success or failure. This pattern is intended to support that smallest device, while still supporting more nuanced messages that may be developed.

While the only value after the leading digit the Response Code defined in Energy Interoperation is 00, conforming specifications may extend these codes to define more fine-grained response codes. These

⁷ This contrasts with Energy Interoperation, where it is not mandatory to return any responses if the entire EiCancelTender service operation was completed successfully. The pattern in Energy Interoperation is to return those that have failed (required) and those that succeeded (optional).

432 should extend the pattern above; for example, a response code of 403 should always indicate Requester
433 Error. Response codes not of the form x00 MAY be treated as the parallel x00 response.

5 CTS Streams

5.1 CTS Streams Introduction

CTS Streams are a conformant specialization of WS-Calendar Streams. Conformance can be established by mapping the elements to the **[Streams]** standard.

For CTS, the simplification contains all defining metadata within a single object, rather than that metadata potentially being found or queried in multiple places.

In addition, CTS Streams include elements that are outside the Streams standard but may be determined by examining referring type instances.

CTS Streams have neither interaction patterns nor payloads, as it is used in defining Facet Payloads.

5.2 Information Model for CTS Streams

The CTS Stream is defined as follows. The elements from **[Streams]** have been flattened into the CTS Stream, and the Stream Payload simplified into a streamPayloadValue and the internal local UID for the stream element.

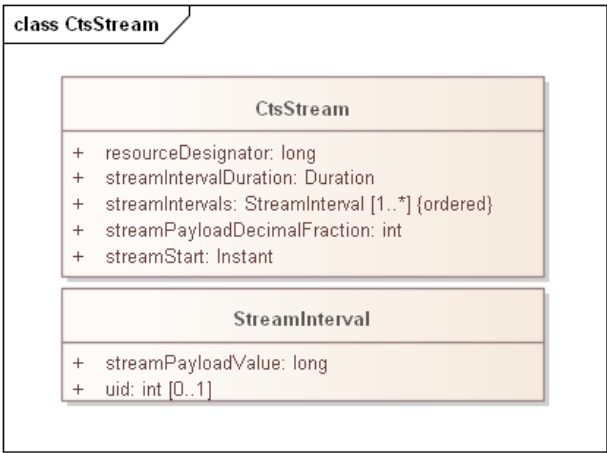


Figure 5-1: CTS Stream Definition

As with **[Streams]**, CTS Stream Intervals are ordered, that is the sequence of intervals is essential. Some serialization specifications, notably XML, do not require that order be preserved when deserializing a list. The UID enables proper ordering of the Stream Intervals if order is not preserved. Serializations of CTS that require preservation of order MAY omit the UID. See

The following tables describe the attributes for CTS Streams and Stream Intervals.

Attribute	Meaning	Notes
Resource Designator	A Long Integer that indicates the Resource for the Product and Market	The Resource Designator in a Market should match Resource Designator as enumerated in the Marketplace
Stream Interval Duration	The duration for each of the contiguous Stream Intervals	The Stream Interval UID enables proper ordering of the Stream Intervals if order is not preserved by the serialization. If the serialization used requires preservation of order MAY omit the UID.
Stream Intervals	The (ordered) set of Stream Intervals	
Stream Payload Mantissa	The Mantissa is to be used to determine actual value for each Stream Payload Value	The Mantissa allow integer comparisons in market implementations. For example, if the decimal fraction digits is -3, and the value (see Stream Interval below) is 1500, the price is 1.500 currency units.
Stream Start	The Start Time for a bound CTS Stream	See WS-Calendar Date-Time in Section 3.1

Attribute	Meaning	Notes
Stream Payload Value	The value for this specific Stream Interval	See note on Stream Payload Decimal Fraction in Table 5-1
UID	An optional “Local UID” for this Stream Interval.	See WS-Calendar Date-Time in Section 3.1. This is optional if Stream Intervals is ordered and/or the serialization used preserves order

6 Market Facet

An Actor interacts with a specific Market to trade a specific Product. A Market matches Tenders for all Instruments based on a given Product. The matching engine is logically contained within the Market and different matching engines have no visibility past the Market Facet.

All interactions in a Market are subject to common rules of engagement which are associated with a Market as identified by a Market Context. The Market Facet describes the behaviors that each Party can expect from the other.

6.1 Market Context History

Market Contexts in [EMIX] and [EI] are URIs and are used to express Market Information that rarely changes, so it is not necessary to communicate it with each message.

Note that a Market Context is associated with and identifies a collection of values and behaviors; while an implementation MAY use operations such as POST to a Market Context URI, that behavior is not required.

For any Market, there are standing terms and expectations about product offerings. If these standing terms and expectations are not known, many exchanges may need to occur before finding products and tenders that meet those expectations. If all market information were to be transmitted in every information exchange, messages would be overly repetitive.

The Market Context for CTS is simplified from that in Energy Interoperation and extended for access to standard terms.

Note that each Market is contained in a Marketplace, and that each Market trades a single Product. Marketplace Characteristics are in progress for a future version of the Market Facet.

6.2 Interaction Pattern for the Market Facet

An Actor interacts with a specific Market to trade a specific Product. A Market matches Tenders for all Instruments based on a given Product. The matching engine is contained within the Market and different matching engines have no visibility past the Market Facet.

The Market Facet enables a Party to request the details of a Marketplace. Using the Market Facet, Parties MAY be able to request and compare Market Contexts to select which markets to participate in.

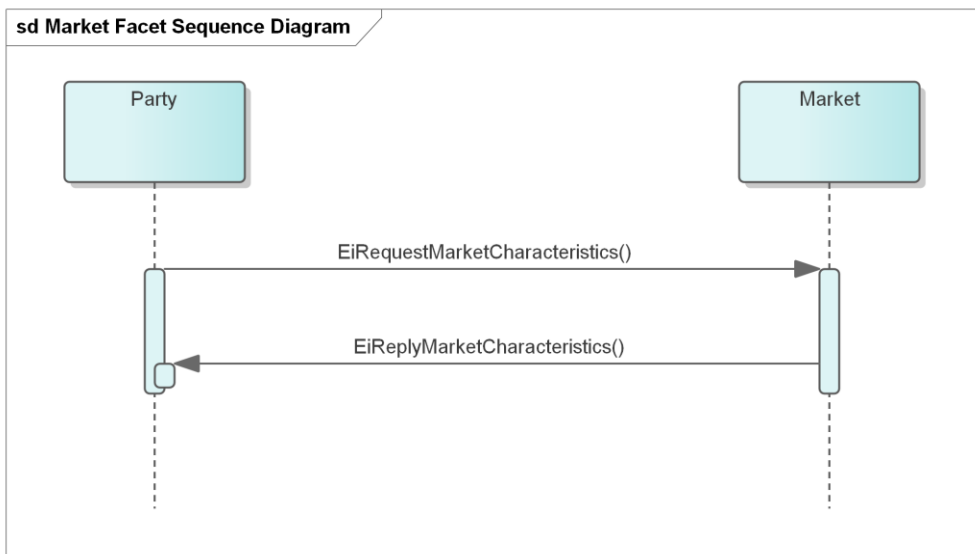
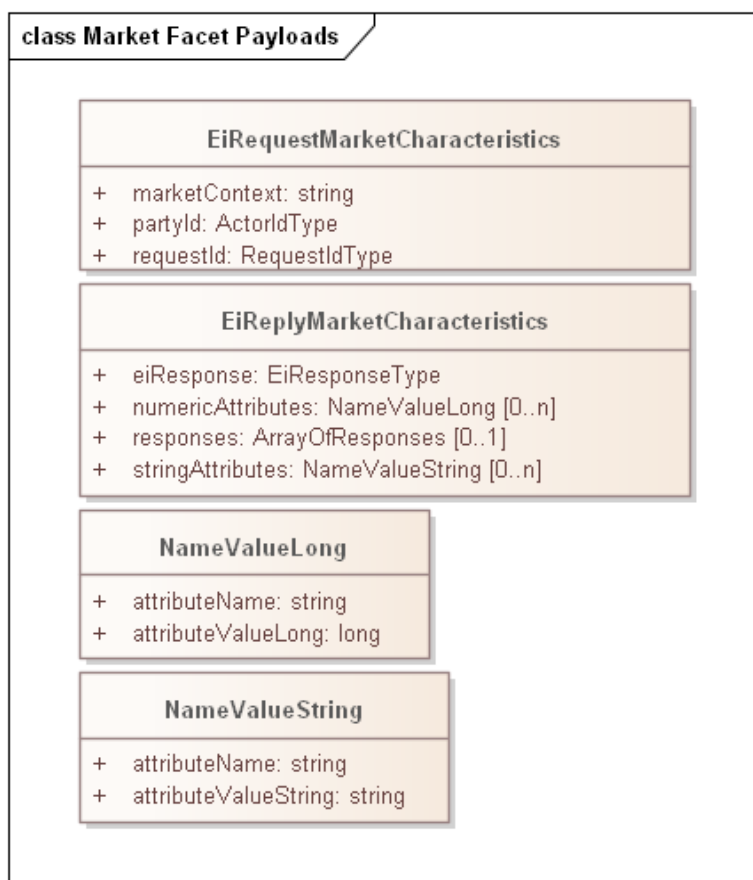


Figure 6-1: UML Sequence Diagram for the Market Facet

The Market Facet can retrieve the standard terms associated with a Market.

488 Delivering an EiRequestMarketCharacteristics payload requests the standard terms for a Market; the
 489 reply payload EiReplyMarketCharacteristics returns those terms as name-value pairs.



490
 491 *Figure 6-2: UML of Market Facet payloads*

492 6.3 Information Model for the Market Facet

493 Sending an EiRequestMarketCharacteristics payload referencing a Market (by containing a market
 494 context) requests standard terms as given in Table 6-1: Standard Terms .

495 These are derived and extended from EMIX Terms; those are extrinsic to the product delivery but effect
 496 how each party interacts with others. Terms may be tied to basic operational needs, or schedules of
 497 availability, or limits on bids and prices acceptability.

498 The CTS Standard Terms MAY be extended to reflect additional capabilities and description.

499 *Table 6-1: Standard Terms returned by Market Facet*

Attribute	Attribute Name	Attribute Type	Meaning
Market Name	NAME	String	Text providing a descriptive name for a Market. While the Name MAY be displayed in a user interface; it is not meaningful to the Actors.

Attribute	Attribute Name	Attribute Type	Meaning
Currency	CURRENCY	String	String indicating how value is denominated in a market. If fiat currency, should be selected from current codes maintained by UN CEFACT. May also be cryptocurrencies or local currency.
Time Offset	T_OFFSET	Long	A Duration that some marketplaces MAY use to describe trading where a first interval is not on an hourly boundary. ⁸
Time Zone	TZ	String	A Time Zone indicates how all Times and Dates are expressed.
Price Decimal Fraction Digits	PRICE_FRAC	Long	Some market implementations use a market-wide indication of how many decimal fraction digits are used. ⁹
Market Party ID	MPARTYID	String	The PartyID to use in a market tender (reference 2.2.3)
Bilateral OK	BILATERAL OK	Long	Boolean expressed as integer 1 - True—bilateral transactions with identified parties are permitted. 2 - False—bilateral transactions not permitted, only market tenders
Resource Designator	R_ID	Long	The Resource traded in this market. This establishes the Resource Designator used in Product definitions and in messages.
Containing Marketplace	MPLACE	String	URI for Marketplace Context
Product	PRODUCT	Array of Ordered Pairs	See Product Definition, Table 2-1: Definitions used in CTS Markets. It SHALL match the Product Definition indicated in the Marketplace for this Market.

⁸ A power distribution entity may experience disruption if there is a big price change on the hour. For example, a distribution system operator (DSO) that operates multiple CTS marketplaces could opt to set a different offset on each Marketplace operated out of a given substation. In this model, a Market could use an offset duration of 3 minutes to indicate that all tenders are based on three minutes after the hour.

⁹ Integer operations are typically much more efficient than fixed or floating point, so it is likely to be much faster to apply decimal shift on input and output rather than for more frequent comparison operations in the matching engine implementation

500

501 Each Product in a Marketplace is defined using attributes as below

502 *Table 6-2: Elements that define Products in a Market or Marketplace*

Attribute	Attribute Name	Meaning
Resource Designator	R_ID	Reference to the required Resource
Time Granularity	T_GRAIN	The interval duration in seconds for the specific Product definition
Quantity Scale	Q_SCALE	The mantissa of the Quantity Scale. For example, a product denominated in kilowatts has a Q_SCALE of 3.
Quantity Granularity	Q_GRAIN	The allowed quantity unit size, e.g. Q_GRAIN == 10 means that a tender for 9 units will be rejected but any multiple of 10 will be accepted.
Price Granularity	PRICE_GRAIN	The allowed price unit, e.g. Price Granularity == 10 means that that any multiple of 10 CURRENCY units is acceptable, but any price not matching, say a price of 9 CURRENCY units, is rejected.
Market	MARKET	The message endpoint to access the market where this Product is traded.
Warrants	WARRANT	Optional further specificity of Product

7 Party Registration Facet

Background (adapted from [EI])

A valid Party ID is required to interact with a market and is included in most payloads.

Party Registration is described in Energy Interoperation. This facet describes messages necessary for an actor (Party) to join a market and to leave or be removed from a market.

- Create Party associates an actor with an ID and informs the marketplace of that ID. CTS makes no representation on whether that ID is an immutable characteristic, such as a MAC address, a stable network address, such as an IP, or assigned during registration,
- Register Party names the exchange of information about an actor that enables full participation in a CTS marketplace. It may exchange information needed for financial transfers including, perhaps, reference to an existing customer or vendor ID, or proof of financial bond for large participants, or issuance of crypto-tokens, or any other local market requirements. A Registered Party is ready to be a full participant in the local market.
- Cancel Party Registration removes a party from the market. It may include final settlement, cancellation of outstanding tenders, backing out of future contracts, or other activities as defined in a particular CTS Marketplace.

Aside from the business services as described, Party Registration may have additional low-level requirements tied to the protocol itself used in a particular implementation based on CTS.

This specification does not attempt to standardize these interactions and messages beyond naming the Register Party facet. A more complete discussion can be found in the [EI] specification.

Some Marketplaces MAY wish to associate one or more measurement points with a Party. Such measurement points could be used to audit transaction completion, to assess charges for using uncontracted for energy, etc. Measurement points are referenced in *Section 11 "Delivery Facet"*, Markets that require this functionality may want to include an enumeration of Measurement Points in Party Registration.

8 Tender Facet

Transactive Services in Energy Interoperation define and support the lifecycle of transactions from initial Tender to final settlement. The phases described in Energy Interoperation are

- Registration—to enable further phases. (Not part of CTS)
- Pre-Transaction —binding tenders for transactions and Quote/Ticker (Part of CTS)
- Transaction Services—execution and management of transactions. (Part of CTS)
- Post-Transaction—settlement, energy used or demanded, payment, position. (Not part of CTS)

For transactive services, the roles are **Parties** and **Counterparties**. The specific actor is identified by its PartyID; see Section 2.2.3.

The terminology of this section is that of business agreements: tenders and transaction. The Service descriptions and payloads are simplified and updated from those defined in Energy Interoperation.

8.1 Tenders as a Pre-Transaction Payloads

Pre-transaction interactions are those between parties that may prepare for a transaction. The pre-transaction facet in CTS is the Tender Facet (and including EiDistributeTender, with payloads shown in Table 8-1.

Tenders and transactions are artifacts based on [EMIX] artifacts suitably flattened and simplified, and which contain schedules and prices in varying degrees of specificity or concreteness.

Table 8-1: Pre-Transaction Tender Facets

Facet	Request Payload	Response Payload	Notes
EiTender	EiCreateTender	EiCreatedTender	Send a CTS-Stream of one or more Tenders. Create and emit Request Payload
EiTender	EiCancelTender	EiCanceledTender	Cancel one or more Tenders
EiTender	EiDistributeTender	None	Describe a list of Tenders to be notified to a set of parties

8.2 Interaction Patterns for the Tender Facet

Figure 8-1 presents the [UML] sequence diagram for the EiTender Facet. Note that EiDistributeTender is not part of CTS 1.0 at present, but is being considered for a future release.

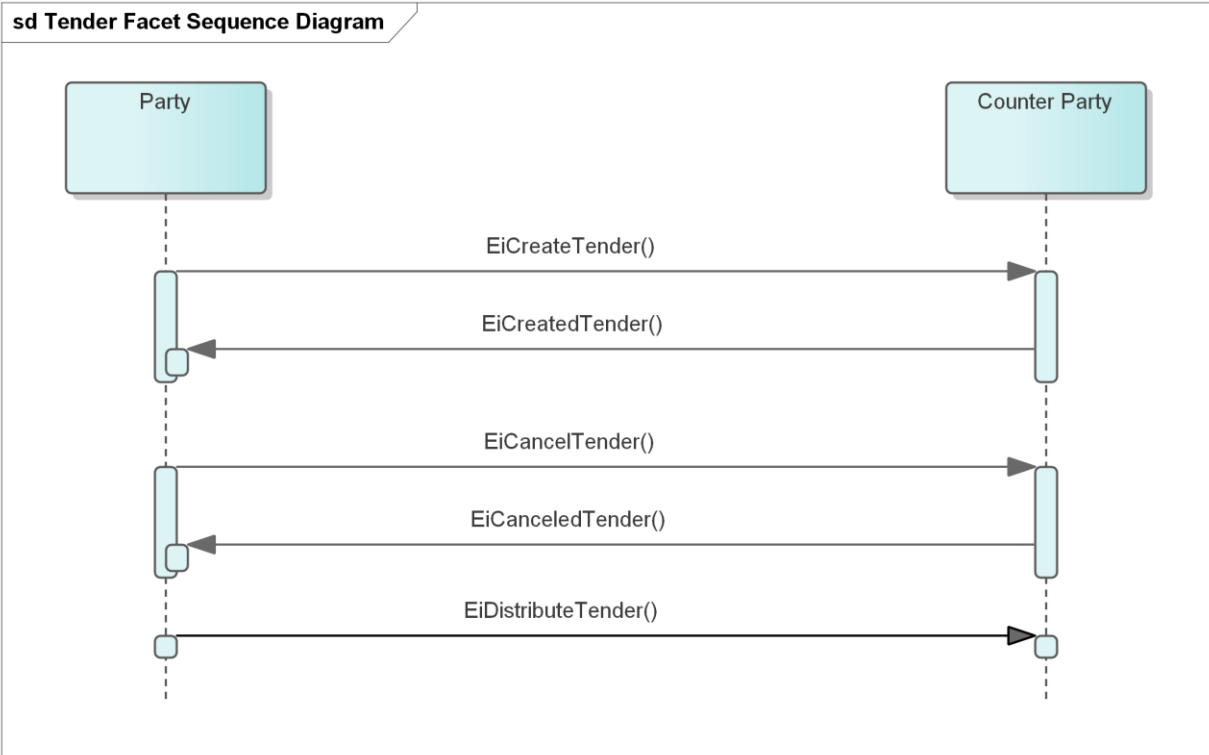


Figure 8-1: UML Sequence Diagram for the Tender Facet

8.3 Information Model for the Tender Facet

The information model for the EiTender Facet artifacts follows that of **[EMIX]**, but flattened and with product definition implied by the implementation. See Section Payloads for the Tender Facet below.

Time interval, price, and quantity are key elements for a product; the other aspects of product definition (e.g. energy and units) are implicit as described in Section 3.2.

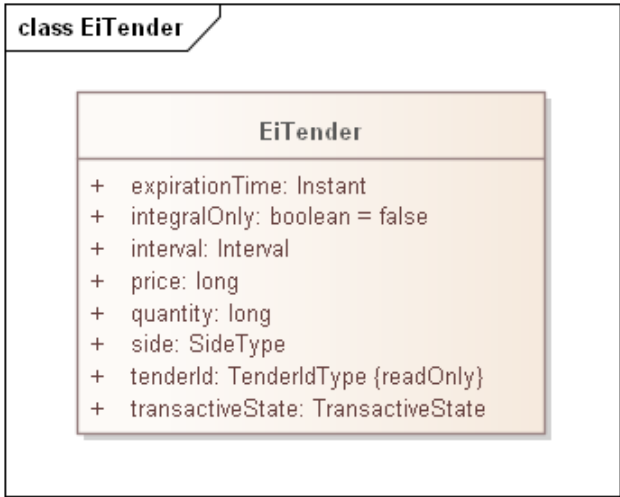


Figure 8-2: Class EiTender

The attributes of EiTender are shown in the following table.

560 Table 8-2: *EiTender Attributes*

Attribute	Meaning	Notes
Expiration Time	The date and time after which this Tender is no longer valid.	
Integral Only	All of the Tender must be bought or sold at once; no partial sale or purchase	In CTS set to False. Partial sale or purchase is always allowed. The attribute is present for possible future evolution.
Interval	The time interval for the product being offered	
Price	The unit price for the product being offered	Total price is the product of price and quantity. Note that price is subject to the Price Decimal Fraction value.
Quantity	The quantity of the product being offered	Total price is the product of price and quantity
Side	Whether the tender is to buy or to sell the product	
Tender ID	An ID for this tender	
Transactive State	The transactive state of this payload is <i>tender</i>	See below

561 Transactive State **[EMIX]** describes the state of a transactive artifact. For CTS 1.0, only the following
562 states are used:

- 563 • *tender*
- 564 • *transaction*
- 565 • *delivery*
- 566 • *publication*

567

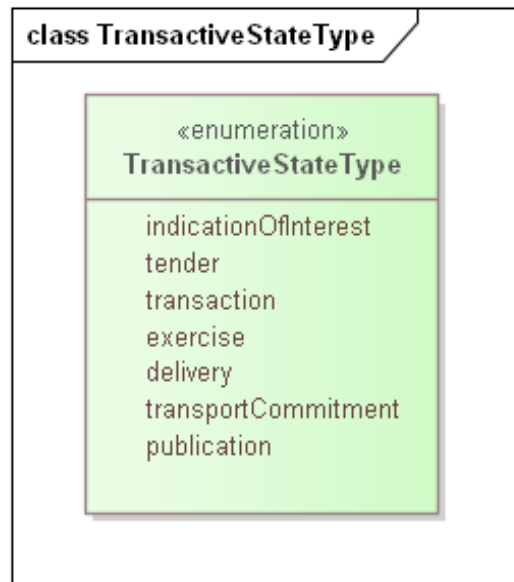


Figure 8-3: Enumeration TransactiveState

8.4 Payloads for the Tender Facet

The [UML] class diagram describes the payloads for the EiTender facet operations.

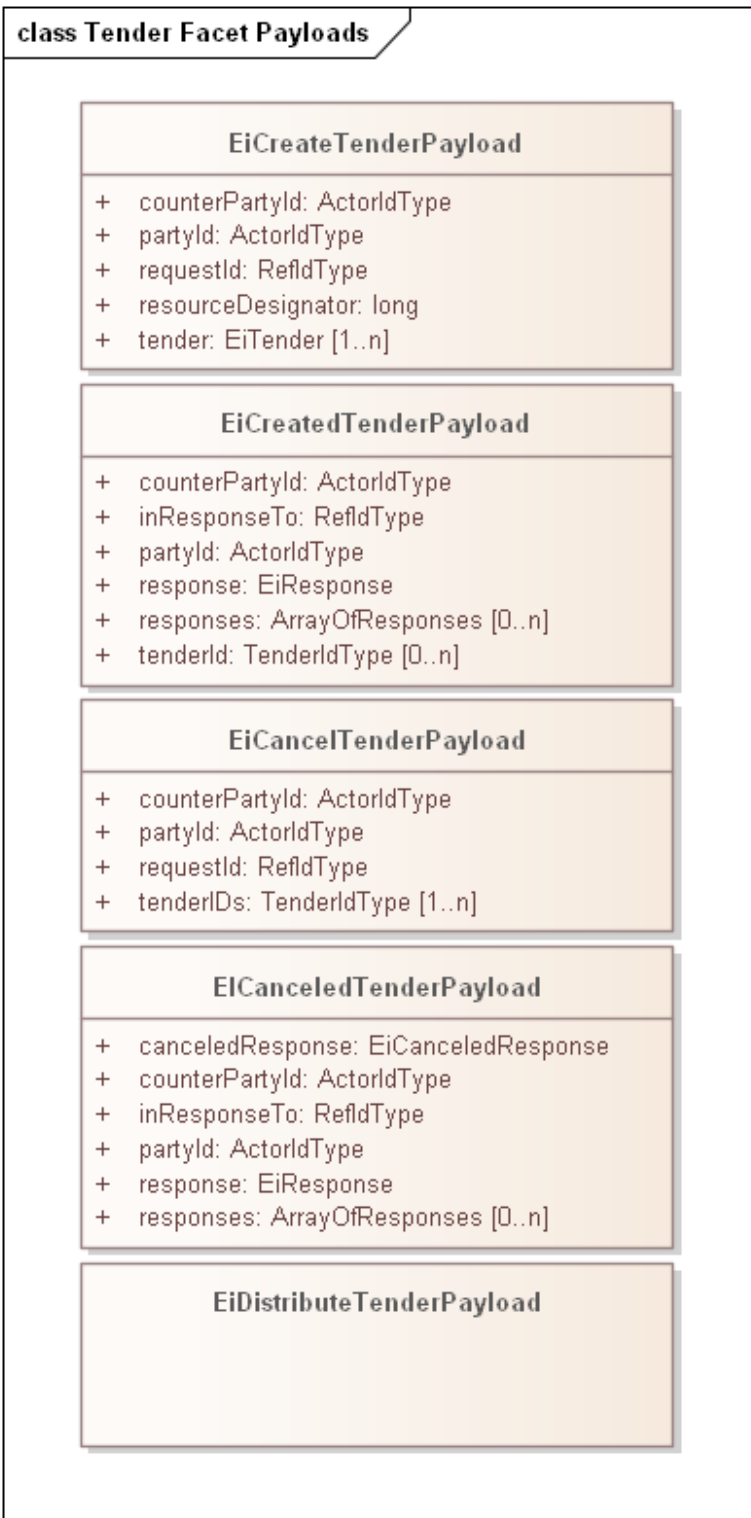


Figure 8-4: UML Class Diagram for the Operation Payloads for the EiTender Facet

The following table describes the attributes for EiCreateTenderPayload

576 Table 8-3 EiCreateTenderPayload Attributes

Attribute	Meaning	Notes
Counter Party ID	The Actor ID for the CounterParty for which the tender is created.	Each market in a Marketplace has a standard term which is the Counter PartyID to use to indicate the expectation that the market will match and clear the tender if possible. In the alternative, for a bilateral tender/transaction, an Actor's PartyID may be used.
Party ID	The Actor ID for the Party on whose behalf this Tender is made.	Indicates which Actor proposes the buy or sell side EiCreateTender.
EiTender	One or more EiTenders to be created.	CTS uses CTS stream of EiTenders. In CTS an object describing a Tender is instantiated then sent; the latter is a consequence of processing an EiCreateTender payload.
Resource Designator	The Resource being tendered	Must match the Market Resource Designator on receipt at the Market
Request ID	A reference ID which identifies the artifact or message element. The Request ID uniquely identifies this request, and can serve as a messaging correlation ID ¹⁰ .	
Responses	Responses for each attempted EiTender creation	Array Of Responses [EI]

577 EiCreateTenderPayload with more than one EiTender SHALL be treated as a shorthand for sending each
578 EiTender in a separate payload.

579 Note that if more than one EiTender is requested to be created, there is no implication that there be an all
580 or none meaning. This avoids the complexity of database-style transaction processing consistency, and
581 simplifies implementations.

¹⁰ As an example of the *Correlation Pattern* for messages

9 Transaction Facet

9.1 Transaction Services

This section presents the Transaction Facet payloads, used by Actors in the role of creating and responding to Transactions.

This section makes them explicit, consistent with the definitions in Section 3.

Canceling or modifying transactions is not permitted.¹¹ Following the approach of distributed agreement protocols¹², compensating tenders and transactions SHOULD be created as needed to compensate for any effects.¹³

Table 9-1: Transaction Management Service

Service	Request Payload	Response Payload	Notes
EiTransaction	EiCreateTransaction	EiCreatedTransaction	Create and acknowledge creation of a Transaction

9.2 Interaction Pattern for the Transaction Facet

This is the [UML] sequence diagram or the EiTransaction Facet:

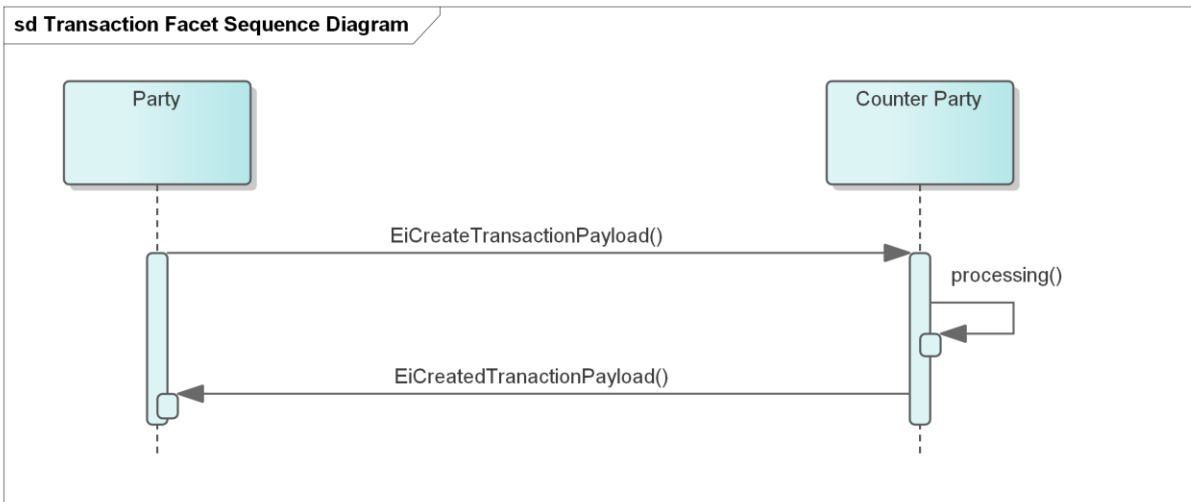


Figure 9-1: UML Sequence Diagram for the EiTransaction Facet

9.3 Information Model for the Transaction Facet

Transactions are a CTS artifact evolved from EMIX including a Stream with time, quantity, and price. Flattening similar to that in the Tender Facet) is used.

¹¹ Canceling transaction is not permitted in either CTS or Energy Interoperation

¹² See, e.g., WS-Transaction and WS-BusinessActivity.

¹³ This is consistent with the way that distributed agreement protocols such as [WS-BusinessActivity] manage compensation rather than cancellation.

Although an EiTransaction object includes the original EiTender, the EiTransaction carries its own Transactive State.

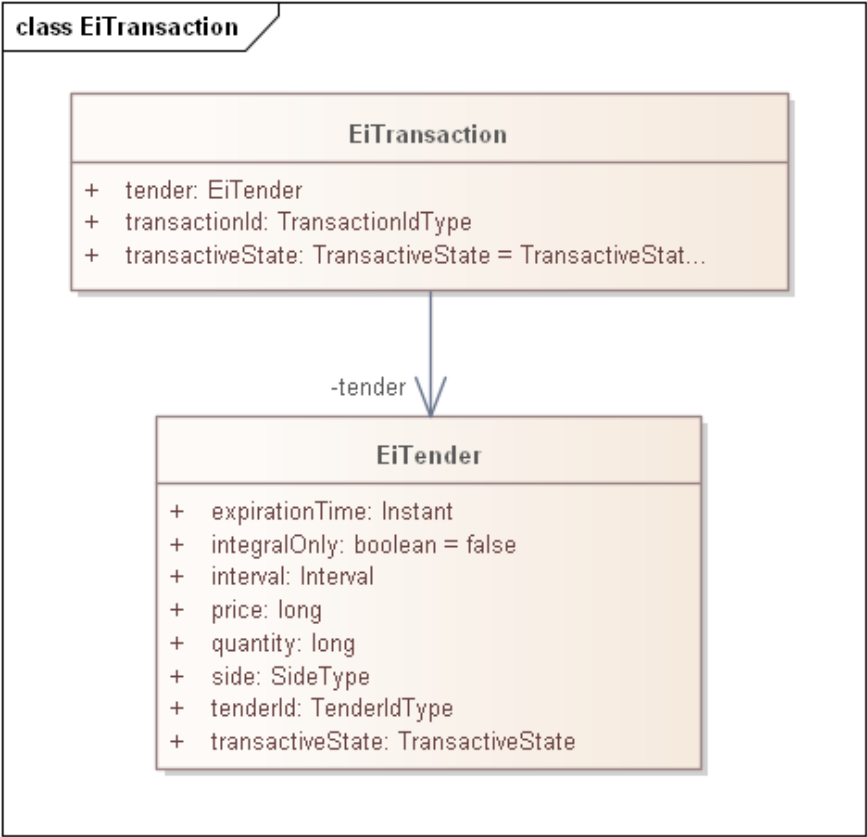


Figure 9-2: UML Class Diagram of EiTransaction

The attributes of EiTransaction are shown in the following table.

Table 9-2: EiTransaction Attributes

Attribute	Meaning	Notes
Tender	The tender (Fig. 4-2) that led to this Transaction.	The ID, quantity and price may differ from that originally tendered due to market actions.
Transaction ID	An ID for this Transaction	The contained Tender has its own Tender Id
Transactive State	The transactive state of this payload is <i>transaction</i>	See Figure 8-3: Enumeration TransactiveState

9.4 Payloads for the Transaction Facet

The [UML] class diagram describes the payloads for the EiTransaction facet operations.

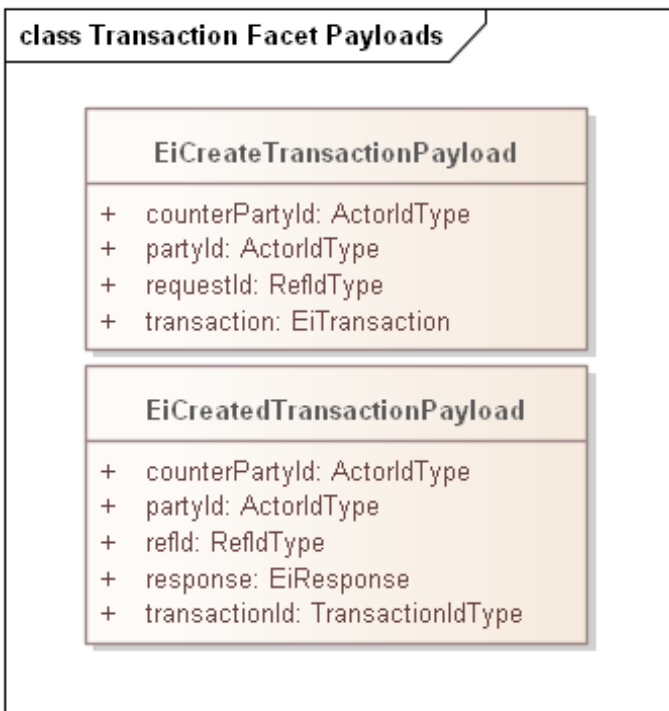
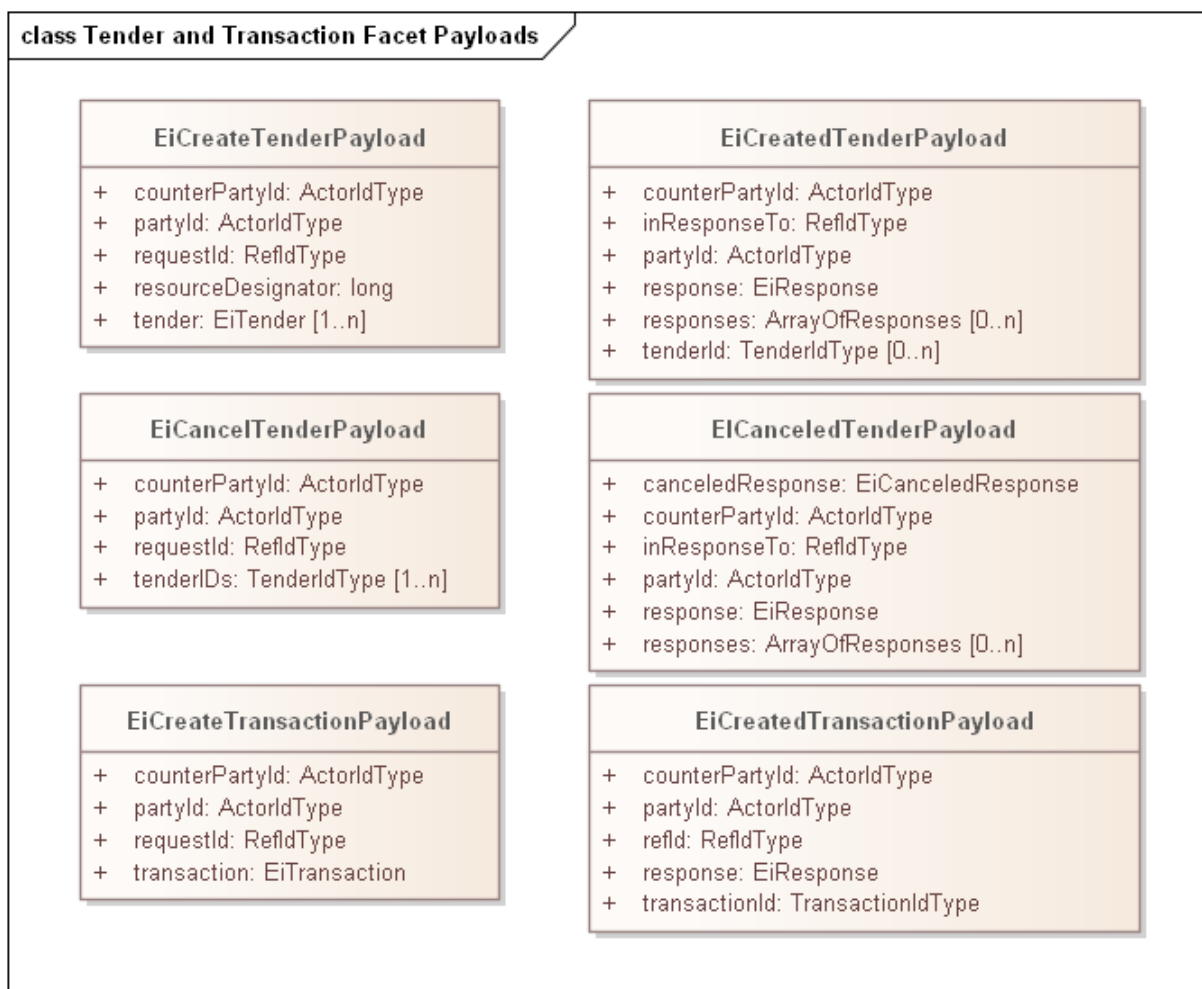


Figure 9-3: UML Class Diagram of EiTransaction Facet Operation Payloads

9.5 Comparison of Transactive Payloads

In this section we show the payloads for the Tender and Transactive Facets



10 Position Facet

10.1 Introduction

The purpose of the Position Facet is to allow access to the accumulated position for actors.

Roles in using the Position Facet include

- The Actor whose position is being requested—the *position party*
- An Actor who is authorized to request position information for other actors—in the nature of an auditor—the *requestor*
- The Market and Product for which the Position is being requested.

10.2 Position Definition

A Party's **Position** for a time period is the algebraic sum of committed supply or sale typically represented as purchases and sales.

The time period for position intervals SHOULD be the same as for the underlying market used to buy and sell, but need not be; conversion of differing time granularity is programmatic and not required by this specification.

A Party needs to know both

- The Party's projected needs for a time interval (not in scope)
- The Party's committed net inflow and outflow for the interval

Note that committed inflow and outflow may be outside a market, e.g. local generation or battery interaction.

An Actor may, with appropriate authorization, request positions for other parties. This permits the specification and implementation of an auditor Actor.

An Actor sees its own Tenders and Transactions, and can maintain its own position. This facet allows the offloading of that data management, but could in fact be a request to a local Position manager.

10.3 Interaction Pattern for the Position Facet

Table 10-1: Position Facet

Facet	Request Payload	Response Payload	Notes
Position	EiRequestPosition	EiReplyPosition	Request an Actor's Position(s) for a specific time interval, and reply with those Position(s) if access is authorized.

This is the [UML] sequence diagram for the Position Facet:

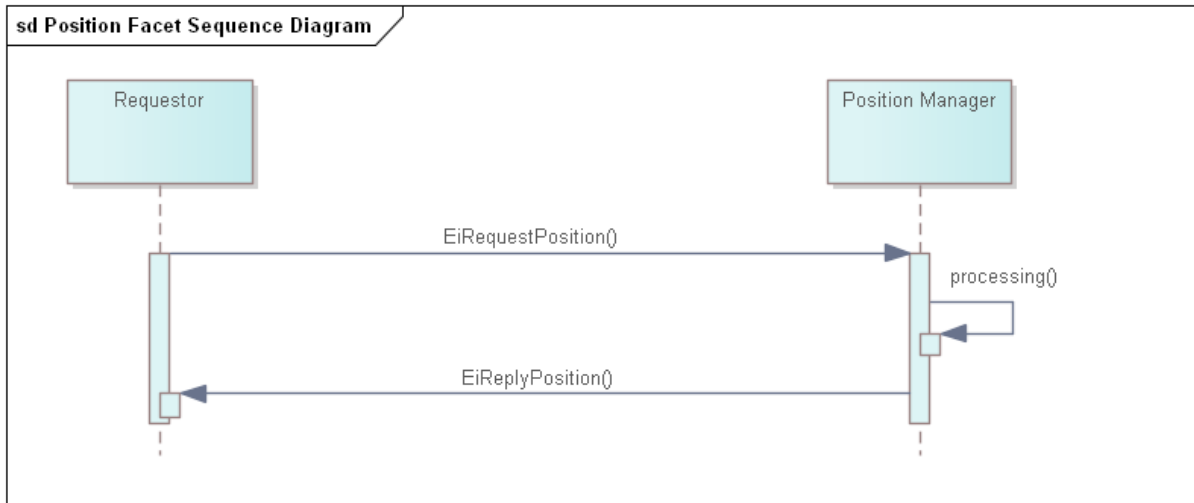


Figure 10-1: UML Sequence Diagram for the Position Facet

10.4 Information Model for the Position Facet

For Position a bounding interval is specified and the position in each interval contained in the closed bounding interval is returned. An Actor has a position in a product, and a product specifies a temporal granularity or Interval duration. This Product duration defines the Interval duration for the returned CTS Stream. All elements of the stream share the duration and the stream has an explicitly stated start time.

A position is concerned with the total amount under contract, not the prices. If an Actor has positions in more than one product, say, in a one-hour product and in a one-minute product, then that requires two requests for position, and the two replies have different interval durations. The integration of these two Positions into a single combined Position is the responsibility of the Requestor.

The attributes are shown in the following section.

10.5 Payloads for the Position Facet

The [UML] class diagram describes the payloads for the Position facet.

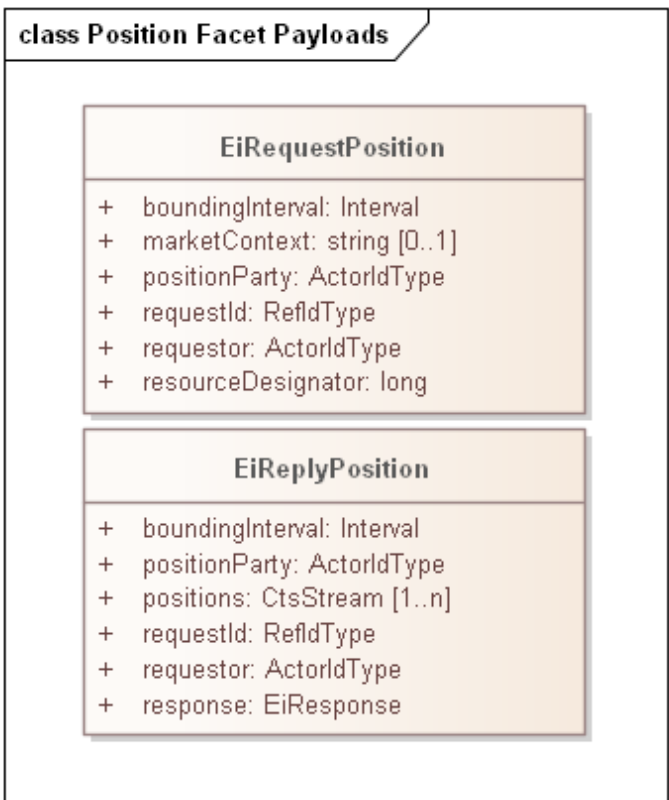


Figure 10-2: UML Class Diagram of Payloads for the Position Facet

Table 10-2: Attributes of Position Facet Payloads

Attribute	Meaning	Notes
Bounding Interval	The [closed] time interval for which position information is requested. The first Positions Stream element starts at or after the start of the Bounding Interval. The last Stream element ends at or before the start of the Bounding Interval.	
Position Party	The Party whose position is being requested.	Allows a request for another Party's position, with appropriate privacy and security constraints
Market Context	The market context of interest	Used to determine the Resource for position. If not present, any resource of which the responder is aware, with no claim to completeness, will be used
Request ID	A reference to this payload	May be used as a correlation ID

Attribute	Meaning	Notes
Requestor	The Party requesting the position.	A failure indication will be returned if the Requestor is not authorized to access position information for Position Party. Addresses the auditor use case.
Positions	CTS Streams containing the positions for Position Party for each Resource. Positions are signed or zero.	Each CTS Stream interval that is contained within the Bounding Interval will have a value associated (signed integer, zero permitted). Note that a CTS Stream contains a Resource Designator
Response	An EiResponse. Will indicate failure if Requestor is not authorized to access position information for Position Party for any of the requested intervals.	

658 The following system-specific requirements are out of scope:

- 659
- 660
- 661
- 662
- 663
- Different systems may support Position requests for different purposes. An Actor MAY request its own position(s) to recover from failure.
 - Positions MAY be used to compute Actor reliability.
 - A supplier of last resort MAY compare Positions to Delivery to impute transactions for unpurchased power delivered. (See 11 Delivery Facet)

11 Delivery Facet

The CTS Delivery Facet can be considered as a telemetry facet. A CTS Delivery payload contains a CTS Stream that conveys the flow of a specific Resource through a particular point on the product's delivery network between particular times.

CTS Delivery is used to report and power flow from a node (as represented and associated with an Actor) into or out of a microgrid. Every contract involves a includes a party that promises to buy as well as a party that promises to sell. Consider an actor that performs temporal arbitrage, i.e., buys one-hour products and sells one-minute products during the same hour. The Actor MAY report that it took delivery in each minute of that Interval, and the sales to other Actors would be visible only as reductions in Delivery.

In most TE markets, taking a greater delivery than contracted for in any interval must be paid for. An auditor, however defined, could sum all positions (See section 10, *Position Facet*) and compare the result to Delivery. The Auditor can then impute a transaction for the over-delivery. This may not be a simple "spot price": if multiple Actors are taking over-delivery then the last small transaction is likely underpriced. Systems that track "actor reputation" may lower the reputation score. These comments are offered to explain the thinking behind this facet, and not to dictate any particular business rule or system model.

A CTS Delivery payload reports on the flow of a resource because the temporal granularity MAY not match that of any particular product. The payload may (e.g.) report the sum of a one-hour market and of a one-minute market for the same Resource.

A CTS marketplace MAY have expectations about levelized load—as do many of today's tariffed markets. Exceeding the limiting bounds for Delivery may result in a market penalty. It is outside the scope of this specification to define the bounds or the nature of the penalty.

A request for delivery specifies a Resource, physical granularity, and temporal granularity. While the physical granularity and temporal granularity need to be within the capabilities of the telemetry node, they need not match any particular Product.]

11.1 Interaction Pattern for the Delivery Facet

Table 11-1: Delivery Facet

Facet	Request Payload	Response Payload	Notes
Delivery	EiRequestDelivery	EiReplyDelivery	Request Delivery through a specific Measurement Point

Figure 4-1 is the [UML] sequence diagram for the Delivery Facet:

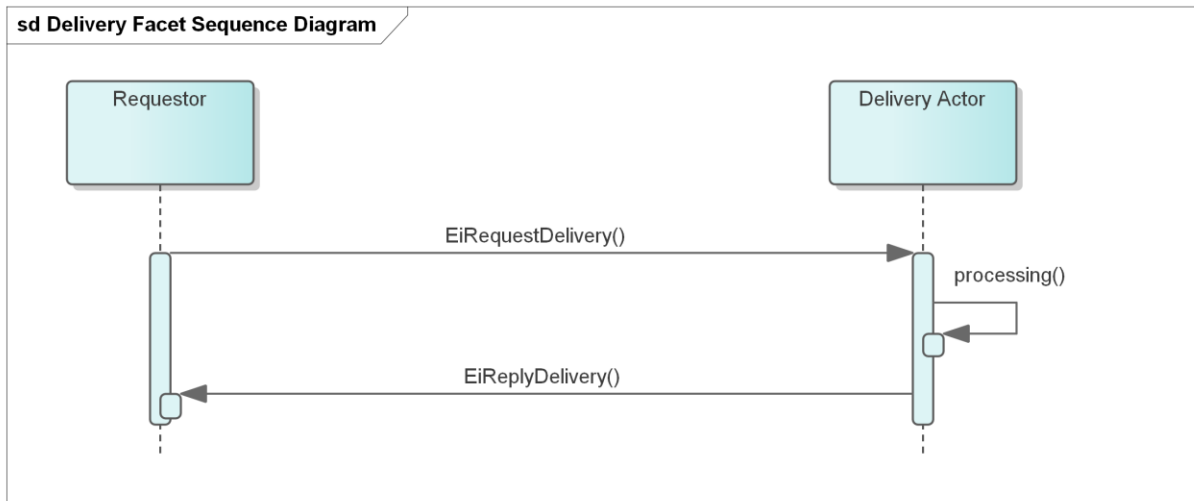


Figure 11-1: UML Sequence Diagram for the Delivery Facet

11.2 Information Model for the Delivery Facet

A Delivery response returns a single CTS Stream of intervals of the requested Duration, with a quantity in each.

As with the Position Facet a bounding interval is specified and the delivery in each interval contained in the closed bounding interval is returned. The granularity as requested MAY not be available, or the Delivery Actor may convert and combine—for example a request for one hour delivery intervals could be responded to using information from 1 minute or 5-minute measurement cycles.

The attributes are shown in the following section.

11.3 Payloads for the Delivery Facet

The [UML] class diagram describes the payloads for the Delivery facet.

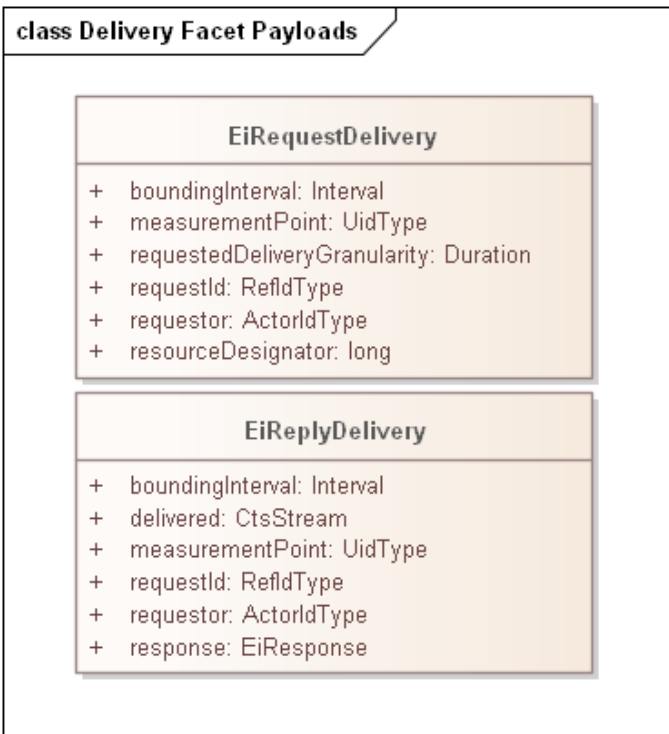


Figure 11-2: UML Class Diagram of Payloads for the Position Facet

Table 11-2: Attributes of Delivery Facet Payloads

Attribute	Meaning	Notes
Bounding Interval	The [closed] time interval for which position information is requested. The first Positions Stream element starts at or after the start of the Bounding Interval. The last Stream element ends at or before the start of the Bounding Interval.	
Measurement Point	The Point for which telemetry is provided about the flow of the resources..	Allows a request to any Measurement Point for information on Resource flow at that point over time. Information should be secured in conformance with appropriate privacy and security constraints
Request ID	A reference to this payload	May be used as a correlation ID

Attribute	Meaning	Notes
Requestor	The Party requesting the position.	A failure indication will be returned if the Requestor is not authorized to access position information for Position Party. Addresses the auditor use case.
Delivered	A CTS Stream containing the Delivery information for the Resource. Delivery value is signed or zero.	Each CTS Stream interval that is contained within the Bounding Interval will have a value associated (signed integer, zero permitted). Note that a CTS Stream contains a Resource Designator which SHOULD match that in the requested Resource Designator
Response	An EiResponse. Will indicate failure if Requestor is not authorized to access position information for Position Party for any of the requested intervals. If the Requested Delivery Granularity cannot be used, MAY indicate what granularity can be used.	

12 Market Information Facet—Quote and Ticker

Tenders are typically private in a market, whether the market matches tenders using an order book, a double auction, or some other means to match buy and seller to award contracts. Markets generate order by enabling price knowledge to emerge from the tenders of independent actors. If all tenders are public, then this price cannot emerge. No seller would ever offer a price less than the highest outstanding tender to buy; no buyer would ever offer a price higher than the lowest outstanding tender to sell. Moreover, analysis of tenders can reveal detailed information about the market participant beyond that necessary to balance supply and demand. (See *Appendix B.2, CTS and Privacy Considerations*.)

Even so, some Actors may wish to advertise specific tenders. In a transitional environment, a utility may wish to publish day ahead prices for each hour of the day. An Actor may wish to draw others into the market quickly in response to a system failure or unplanned-for need—and may offer an unusually high or low price to attract sellers or buyers. Others may wish to quickly dispose of a previous position. A distribution operator in TE markets may wish to advertise short term deals temporal price boundaries to protect grid components by smoothly ramping power delivery requirements. Whatever the reason, [EI] specifies the EiQuote service for advertising tenders.

12.1 Quotes

[EI] defines a quotation as a market price or possible price, which needs a tender and acceptance to reach a transaction. An advertisement of an attractive price for limited amount of power might only be available to the first to respond. That said, a Quote looks very much like a tender.

Different CTS-based systems may want to distribute Quotes in different ways. Some may permit an Actor to broadcast Quotes to all other Actors. Others will require that a Quote be submitted to the Market which will then distribute the Quote to all subscribers. A market MAY choose to protect privacy by indicating its own Actor Id as the originator of all quotes.

12.1.1 Interaction Pattern for Market Information Facet Quote

This is the [UML] sequence diagram for the Market Information Facet Quote:

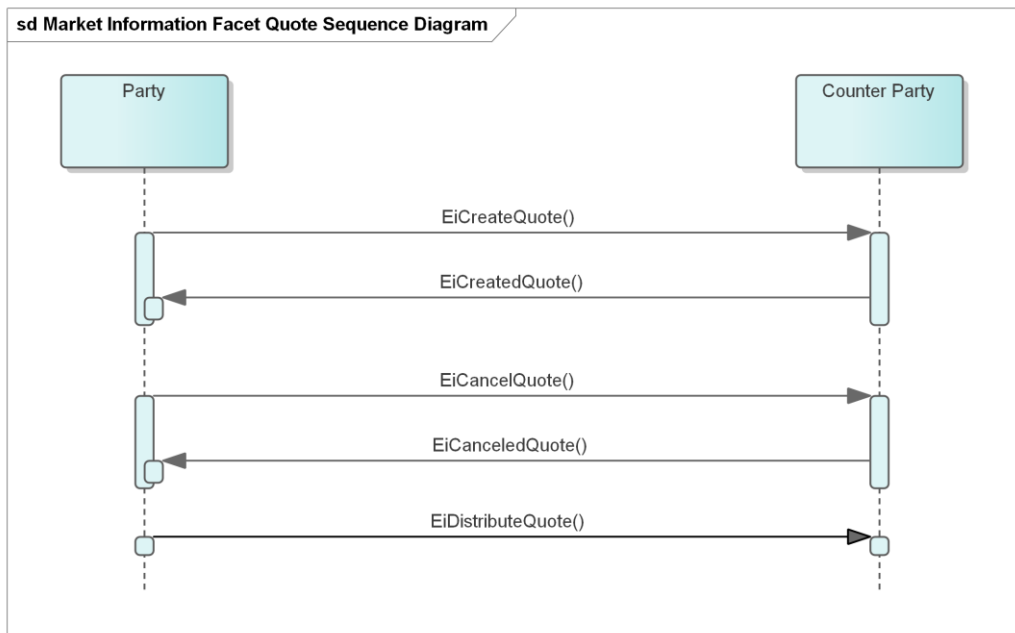


Figure 12-1: UML Sequence Diagram for the Quotation Facet

12.1.2 Information Model for the Quote

An Actor may submit quotes for a number of consecutive Intervals, a set of Instruments for an identical product. An example is a load serving entity quoting 24 prices for the next day. All elements of the stream share the duration and the stream has the explicitly stated start time.

Table 12-1: Quotation Facet

Facet	Request	Response	Notes
Market Information	EiCreateQuote	EiCreatedQuote	Creates a Quote
Market Information	EiDistributeQuote		Used for a broadcast of a Quote. Depending on system business rules, MAY be only to subscribers
Market Information	EiCancelQuote	EiCanceledQuote	This can be point to point or broadcast per system design

12.2 Tickers

EDITOR'S NOTE Comments welcome. This section is incomplete.

While Tenders may be private, the existence of contracts are expected to be public (although typically without party identification). Subscribing Actors are continuously informed of executed contracts by means of Tickers. This facet is named by analogy to the earliest financial communications medium, which

transmitted stock price information to a machine called a stock ticker, which printed the information on a continuous paper strip. The term "ticker" came from the sound made by the machine as it printed.

EDITOR'S NOTE Ticker needs a Product; there is some confusion between functions of a messaging

12.2.1 Information Model for the Market Information Facet Ticker

The information model for the Ticker is the same as that for a Transaction. Depending on specific system and privacy requirements, the ticker may replace one or both of the Party ID and Counter-Party ID may be absent from the Ticker. If a requirement for strong message typing requires their inclusion, the Party ID of the market can be substituted for either or both.

Table 12-2: Market Information Facet Ticker

Facet	Request	Response	Notes
Market Information	EiCreateTicker	EiCreatedTicker	Create a ticker for a specific Market and reply with a reference.
Market Information	EiCancelTicker	EiCanceledTicker	This can be point to point or broadcast as per system design
Market Information	EiDistributeTicker		Distribute a reference to a Ticker. Similar conditions apply as for EiDistributeTender

12.2.2 Interaction Pattern for the Market Information Facet Ticker

This is the [UML] sequence diagram for the Market Information Facet Ticker:

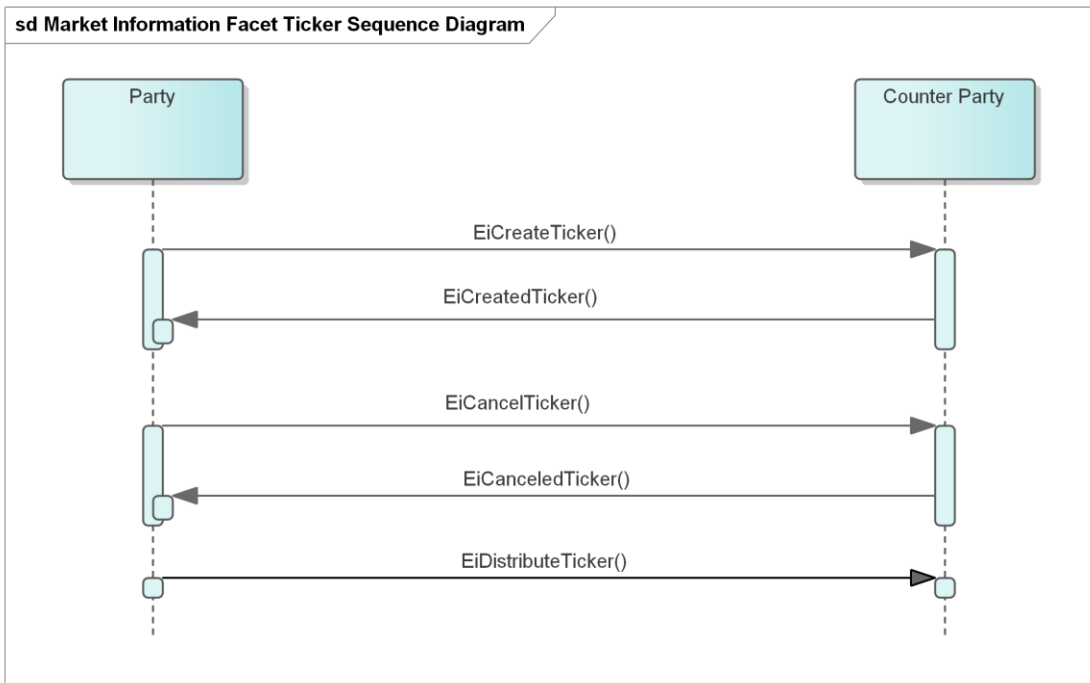


Figure 12-2 Market Information Facet Ticker Sequence Diagram

13 Bindings

Payloads and interaction patterns are described in [UML] in Sections 6 through 12 above. This section contains bindings for the payloads in three encoding schemes:

- JSON [JSON]
- XML Schema [XSD]
- FIX Simple Binary Encoding [SBE]

13.1 JSON

PENDING—JSON Schema awaiting stable payload definitions

13.2 XML Schema

PENDING—XML Schema awaiting stable payload definitions

13.2.1 XML Namespaces

PENDING—XML Namespaces awaiting XML Schema

13.3 Simple Binary Encoding

TODO—SBE Schema awaiting stable payload definitions

14 Conformance

14.1 Introduction to Conformance

By design, CTS is a simplified and restricted subset profile of TeMIX.

Portions of CTS conform to and use updated and simplified versions of the specifications consumed by Energy Interoperation, specifically

- OASIS WS-Calendar **[WS-Calendar]**
- OASIS WS-Calendar Schedule Streams and signals **[Streams]**

This draft specification uses the WS-Calendar **[CALMIN]** interval directly (as IntervalType). This specification simplifies WS-Calendar Schedule Streams and Signals [Streams] as CTS Streams.

14.2 Claiming Conformance to Common Transactive Services

Implementations claim conformance to Common Transactive Services 1.0 by asserting conformance statements on the numbered items below.

1. The conformance statement MUST list all Facets which it supports in full or and in part.
2. The conformance statement MUST describe all extensions to payloads described in this specification.
3. The conformance statement MUST describe the Binding(s) which it supports along with any extensions. If the implementation does not use a standard binding as defined in Section 13, the conformance statement MUST define the binding used, at a similar level to detail to Section 13.
4. The conformance statement MUST describe how each payload definition conforms to the UML and/or profiled definitions for each payload unless it uses only standard Bindings in Section 13.
5. The conformance statement MUST indicate cardinality for message payload attributes where there is flexibility in this specification.
6. The conformance statement MUST describe any facets it defines to extend this specification.

Appendix A. References

This appendix contains the normative and informative references that are used in this document. Normative references are specific (identified by date of publication and/or edition number or Version number) and Informative references may be either specific or non-specific.

While any hyperlinks included in this appendix were valid at the time of publication, OASIS cannot guarantee their long-term validity.

A.1 Normative References

The following documents are referenced in such a way that some or all of their content constitutes requirements of this document.

NOTE: INSERT AS FORMATTED REFERENCES. Consider [EI]

[CAL-MIN]

WS-Calendar Minimal PIM-Conformant Schema Version 1.0. Edited by William Cox and Toby Considine. 26 August 2016. OASIS Committee Specification. <http://docs.oasis-open.org/ws-calendar/ws-calendar-min/v1.0/ws-calendar-min-v1.0.html>

[CAL-PIM]

OASIS WS-Calendar Platform-Independent Model version 1.0, Committee Specification 02 Edited by William T. Cox and Toby Considine, 21 August 2015. <http://docs.oasis-open.org/ws-calendar/ws-calendar-pim/v1.0/cs02/ws-calendar-pim-v1.0-cs02.html> Latest version: <http://docs.oasis-open.org/ws-calendar/ws-calendar-pim/v1.0/ws-calendar-pim-v1.0.html>

[EI]

Energy Interoperation Version 1.0. Edited by Toby Considine, 11 June 2014. OASIS Standard. <http://docs.oasis-open.org/energyinterop/ei/v1.0/os/energyinterop-v1.0-os.html> Latest version: <http://docs.oasis-open.org/energyinterop/ei/v1.0/energyinterop-v1.0.html>. and its TeMIX Profile

[EMIX] OASIS Energy Market Information Exchange (EMIX) Version 1.0 Committee Specification 02 Edited by Toby Considine, 11 January 2012. <http://docs.oasis-open.org/emix/emix/v1.0/cs02/emix-v1.0-cs02.html> Latest version: <http://docs.oasis-open.org/emix/emix/v1.0/emix-v1.0.html>

[JSON]

JavaScript Object Notation and JSON Schema. <https://cswr.github.io/JsonSchema/>

[RFC8174]

Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<http://www.rfc-editor.org/info/rfc8174>>.

[RFC2119]

Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.

[RFC2246]

T. Dierks, C. Allen *Transport Layer Security (TLS) Protocol Version 1.0*, <http://www.ietf.org/rfc/rfc2246.txt>, IETF RFC 2246, January 1999.

[SBE]

Simple Binary Encoding Technical Specification 1.0. FIX Trading Community, June 16, 2016. <https://www.fixtrading.org/standards/sbe/>

[Streams]

Schedule Signals and Streams Version 1.0. Edited by Toby Considine and William T. Cox. 18 September 2016. OASIS Committee Specification. <http://docs.oasis-open.org/ws-calendar/streams/v1.0/streams-v1.0.html>.

A.2 Informative References

The following referenced documents are not required for the application of this document but may assist the reader with regard to a particular subject area.

[Actor Model]

C. Hewitt, "Actor Model of Computation: Scalable Robust Information Systems," arxiv.org, 2010.

[Framework]

National Institute of Standards and Technology, *NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0*, January 2010, http://nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf

[CTS2016]

W.T. Cox, E. Cazalet, E., A Krstulovic, W Miller, & W.Wijbrandi *Common Transactive Services*. TESC 2016. Available at <http://coxsoftwarearchitects.com/Resources/TransactiveSystemsConf2016/Common%20Transactive%20Services%20Paper%2020160516.pdf>

[EML-CTS]

Energy Mashup Lab Common Transactive Services (open-source software) <https://github.com/EnergyMashupLab/eml-cts>

[FSGIM]

Facility smart grid information model. ISO 17800. <https://www.iso.org/standard/71547.html> 2017

[iCalendar]

B. Desruisseaux, *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*, <https://tools.ietf.org/html/rfc5545>. 2009,
See also
C. Daboo & M. Douglas. *Calendar Availability*, <https://tools.ietf.org/html/rfc7953>, 2016

[GridFaultResilience]

W.T. Cox & T. Considine. *Grid Fault Recovery and Resilience: Applying Structured Energy and Microgrids..* IEEE Innovative Smart Grid Technologies 2014. Available at http://coxsoftwarearchitects.com/Resources/ISGT_2014/ISGT2014_GridFaultRecoveryResilienceStructuredMicrogrids_Paper.pdf

[Micromarkets]

W.T. Cox & T. Considine, *Energy, Micromarkets, and Microgrids*. GridInterop 2011, https://www.gridwiseac.org/pdfs/forum_papers11/cox_considine_paper_gi11.pdf

[RFC3552]

E Rescorla & B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <<https://www.rfc-editor.org/info/rfc3552>>.

[SmartGridBusiness]

T. Considine & W.T. Cox, *Smart Loads and Smart Grids—Creating the Smart Grid Business Case*. Grid-Interop 2009. Available at <http://coxsoftwarearchitects.com/Resources/Grid-Interop2009/Smart%20Loads%20and%20Smart%20Grids.pdf>

[StructuredEnergy]

Structured Energy: Microgrids and Autonomous Transactive Operation, http://coxsoftwarearchitects.com/Resources/ISGT_2013/ISGT-Cox_StructuredEnergyPaper518.pdf. Innovative Smart Grid Technologies 2013 (IEEE).

[TeMIX] TeMIX Transactive Energy Market Information Exchange [TeMIX] an approved Note of the EMIX TC. Ed Cazalet et al., 23 May 2010. <http://www.oasis-open.org/committees/download.php/37954/TeMIX-20100523.pdf>

[TRM] (Transactive Resource Management)

B. Huberman and S. H. Clearwater, *Thermal markets for controlling building environments*, Energy Engineering, vol. 91, no. 3, pp. 26- 56, January 1994.

893 **[UML]**
894 Object Management Group, *Unified Modeling Language (UML)*, V2.4.1, August 2011.
895 <http://www.omg.org/spec/UML/2.4.1/>

896 **[XSD]**
897 *W3C XML Schema Definition Language (XSD) 1.1*. Part 1: Structures, S Gao, C. M. Sperberg-McQueen,
898 H Thompson, N Mendelsohn, D Beech, M Maloney <http://www.w3.org/TR/xmlschema11-1/>, April 2012,
899 Part 2: Datatypes, D Peterson, S Gao, A Malhotra, C. M. Sperberg-McQueen, H Thompson, P Biron,
900 <http://www.w3.org/TR/xmlschema11-2/> April 2012
901

Appendix B. Security and Privacy Considerations

This specification defines message payloads only. Security must be composed in. Privacy considerations must be decided when implementing specific systems for specific purposes.

B.1 CTS and Security Considerations

Procuring energy for local use and selling energy for remote use are each at the cusp of finance and operations.

- A price that is falsely low may cause the buyer to operate a system when there is inadequate power, potentially harming systems within a facility, or harming other facilities on the same circuit.
- A price that is falsely low may cause the seller to leave the market.
- A price that is falsely high may cause the buyer to shut down operation of systems or equipment.
- A price that is falsely high may cause the seller increase operations when there is neither a ready consumer or perhaps even grid capacity to take delivery.

For these reasons, it is important that each system guard the integrity of each message, assure the sender and of the receiver, and prove whether a message was received or not.

Messages should be encrypted to prevent eavesdropping. Any node should be able detect replay, message insertion, deletion, and modification. A system must guard against and detect man-in-the-middle attacks wherein an intermediary node passes of messages as originating from a known and trusted source.

B.2 CTS and Privacy Considerations

The United Nations has defined privacy as “the presumption that individuals should have an area of autonomous development, interaction and liberty, a ‘private sphere’ with or without interaction with others, free from state intervention and excessive unsolicited intervention by other uninvited individuals. The right to privacy is also the ability of individuals to determine who holds information about them and how that information is used” (UN General Assembly 2013:15).

Electrical usage data inherently creates a privacy risk. Published work has demonstrated that simple usage data can be used to reveal the inner operations and decisions in a home. Other research has demonstrated that anonymous electrical usage data can be “de-anonymized” to identify an individual electricity user. The more fine-grained the data, the more intimate the details that can be garnered from meter telemetry.

In an amicus brief in a case on smart metering, the Electronic Freedom Foundation testified that that aggregate smart meter data collected from someone’s home in 15-minute intervals could be used to infer, for example, whether they tend to cook meals in the microwave or on the stove; whether they make breakfast; whether and how often they use exercise equipment, such as a treadmill; whether they have an in-home alarm system; when they typically take a shower; if they have a washer and dryer, and how often they use them; and whether they switch on the lights at odd hours, such as in the middle of the night. And these inferences, in turn, can permit intimate deductions about a person’s lifestyle, including their occupation, health, religion, sexuality, and financial circumstances. These privacy concerns are linked to increased security risks criminals may be able to access the data and use the information to enable inferences about what people are doing in their home or if they are away from home.

This specification describes how to share communications beyond mere electrical usage telemetry. Communications reveal what the user would like to buy, how much they would be willing to spend, and future intents and plans.

System developers using this specification should consider legal requirements under the Fair Practice Principles and the European Union’s General Data Protection Regulation. These include:

- 1) The Collection Limitation Principle. There should be limits to the collection of personal data and any such data should be obtained by lawful and fair means and, where appropriate, with the knowledge or consent of the data subject.

- 949 2) The Data Quality Principle. Personal data should be relevant to the purposes for which they are
950 to be used and, to the extent necessary for those purposes, should be accurate, complete and
951 kept up-to-date.
- 952 3) The Purpose Specification Principle. The purposes for which personal data are collected should
953 be specified not later than at the time of data collection and the subsequent use limited to the
954 fulfillment of those purposes or such others as are not incompatible with those purposes and as
955 are specified on each occasion of change of purpose.
- 956 4) The Use Limitation Principle. Personal data should not be disclosed, made available or otherwise
957 used for purposes other than those specified, except a) with the consent of the data subject, or b)
958 by the authority of law.
- 959 5) The Security Safeguards Principle. Personal data should be protected by reasonable security
960 safeguards against such risks as loss or unauthorized access, destruction, use, modification or
961 disclosure of data.
- 962 6) The Openness Principle. There should be a general policy of openness about developments,
963 practices and policies with respect to personal data. Means should be readily available of
964 establishing the existence and nature of personal data and the main purposes of their use, as
965 well as the identity and usual residence of the data controller.
- 966 7) The Individual Participation Principle. An individual should have the right:
- 967 a. to obtain from a data controller, or otherwise, confirmation of whether or not the data
968 controller has data relating to him;
- 969 b. to have data relating to him communicated to him, within a reasonable time, at a charge,
970 if any, that is not excessive; in a reasonable manner, and in a form that is readily
971 intelligible to him;
- 972 c. to be given reasons if a request made under subparagraphs (a) and (b) is denied and to
973 be able to challenge such denial; and
- 974 d. to challenge data relating to him and, if the challenge is successful, to have the data
975 erased, rectified, completed or amended;
- 976 8) The Accountability Principle. A data controller should be accountable for complying with
977 measures which give effect to the principles stated above.

978 In developing this specification, the Technical Committee has kept in mind the need to support a
979 developer wishing to support privacy. Actors representing an up-stream electrical serving entity, say a
980 distribution system operator or traditional utility, use the same messages as anyone else—no actor is
981 inherently privileged. Messages to provide market information or “ticker-tape” functions do not include
982 party IDs. General advertising of tenders, while necessary to draw matching tenders quickly to market,
983 may be anonymous.

984 The system developer should keep the privacy principals in mind when making specific technology
985 choices. For example, messages between an actor and the market MAY be encrypted to protect the
986 privacy of people represented by individual actors. While the transactive energy market must know both
987 buyers and sellers to support transaction contracts and settlements, the developer should take steps to
988 guard that information. A developer may opt that each notice of contract sent to an actor always has a
989 counterparty of the market, so as to protect the sources and uses of electricity.

990 It is beyond the scope of this specification to specify security practices and system design form markets
991 built using this specification.

Appendix C. Glossary of Terms and Abbreviations Used in this document

Throughout this document, abbreviations are used to improve clarity and brevity, especially to reference specifications with long titles.

Table C--1 Abbreviations and Terms used throughout this document

Attribute	Meaning
CTS	Common Transactive Services
EI	Energy Interoperation, an OASIS specification as per the normative references, CTS is a conforming profile of EI.
EMIX	Energy Market Information Exchange, an OASIS specification used to describe products and markets for resources, particularly those traded in power grids.

Appendix D. Acknowledgments

This work is derived from the specification Common Transactive Services 1.0 , contributed by The Energy Mashup Lab, written by William T. Cox and Toby Considine.

Portions of models and text is derived from The Energy Mashup Lab open source project, EML-CTS and is used under terms of the Apache 2.0 License for that project.¹⁴

D.1 Participants

The following individuals were members of this Technical Committee during the creation of this document and their contributions are gratefully acknowledged:

Rolf Bienert, OpenADR Alliance
Toby Considine, University of North Carolina at Chapel Hill
William T. Cox, Individual Member
Pim van der Eijk, Sonnenglanz Consulting
David Holmberg, National Institute for Standards & Technology (NIST)
Elysa Jones, Individual
Chuck Thomas, Electric Power Research Institute (EPRI)

¹⁴ <https://github.com/EnergyMashupLab/eml-cts>

Appendix E. Revision History

Revision	Date	Editor	Changes Made
WD01	2/15/2021	Toby Considine	Initial reformatting and conversion of the specification contributed by The Energy Mashup Lab to create a document for committee work.
WD02	3/3/2021	Toby Considine	Added prose definitions of Resource, Product, and Instrument
WD03	4/5/2021	Toby Considine	Simplified introductory material, raised message type to earlier in document. Removed some repetitive material. Revised UML required.
WD04	5/7/2021	Toby Considine David Holmberg William T Cox	Reordered intro material to reduce repetition, Reference Actor Model more consistently, Revise and re-factor Resource/Product/Instrument Add Section 3 to elevate common semantic elements
WD05	5/25/2021	Toby Considine David Holmberg William T Cox	Continues clean-up and condensation of sections 1, 2
WD06	6/7/2021	Toby Considine	Refines Item language into Resource and Products. Explains Message Groups as a conforming descendant of EI Services.
WD07	6/21/2021	Toby Considine William T Cox	Clarified terminology and relationship to implied Service-Oriented Architecture. Structured CTS facets for clearer explanation
WD08	8/5/2021	Toby Considine William T Cox David Holmberg	Clarify and simplify actor facets descriptions, including Tender, Transaction, and Configuration. Reduce redundant and less relevant content.
WD09	9/14/2021	William T Cox Toby Considine David Holmberg	Added Facet descriptions for Position, Market Characteristics, CTS Streams, and drafts of Privacy Consideration, Delivery and Party Registration Facets. Numerous edits for clarity and conciseness.
WD10	10/4/2021	Toby Considine William T Cox David Holmberg	Extended Market Facets. Defined Position and Delivery facets. Made references more consistent. Updated UML model and diagrams.
WD11	10/22/2021	David Holmberg William T Cox Toby Considine	Corrections for clarity. Improved UML diagrams. Flagged requests for comments in Public Review

Notices

Copyright © OASIS Open 2021. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website: [\[https://www.oasis-open.org/policies-guidelines/ipr\]](https://www.oasis-open.org/policies-guidelines/ipr).

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OASIS AND ITS MEMBERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THIS DOCUMENT OR ANY PART THEREOF.

As stated in the OASIS IPR Policy, the following three paragraphs in brackets apply to OASIS Standards Final Deliverable documents (Committee Specifications, OASIS Standards, or Approved Errata).

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

The name "OASIS" is a trademark of OASIS, the owner and developer of this document, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, documents, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.