8 Position Facet

8.1 Introduction

The purpose of the Position Facet is to allow actors including

- The Actor whose position is being requested
- An Actor who is authorized to request position information for other actors—in the nature of an auditor

8.2 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.

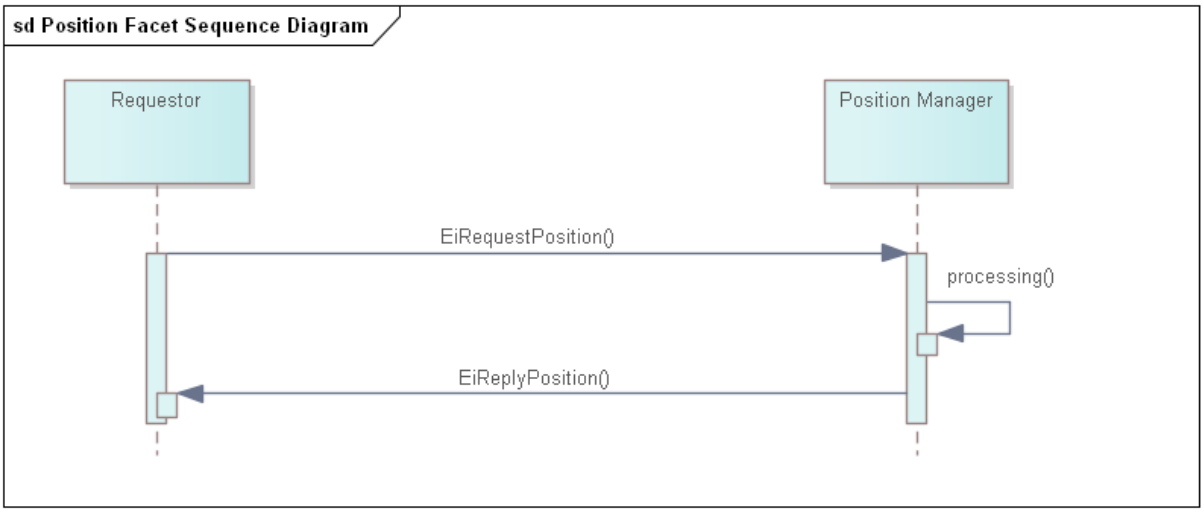
8.3 Interaction Pattern for the Position Facet

Table 8-1: Position Service

Facet	Request	Response	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.

EDITOR'S NOTE: input and creation of the logical Position database is out of scope; one approach using a SQL database is in EML-CTS.

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



774

775 *Figure 8-1: UML Sequence Diagram for the Position Facet*

776 8.4 Information Model for the Position Facet

777 EDITOR'S NOTE: This follows the EML-CTS implementation where a bounding interval is specified and
778 the position in each position interval contained in the closed bounding interval is returned.

779 An extension or alternate would be to return a CtsStream containing the positions for each individual
780 interval contained within the closed bounding interval.

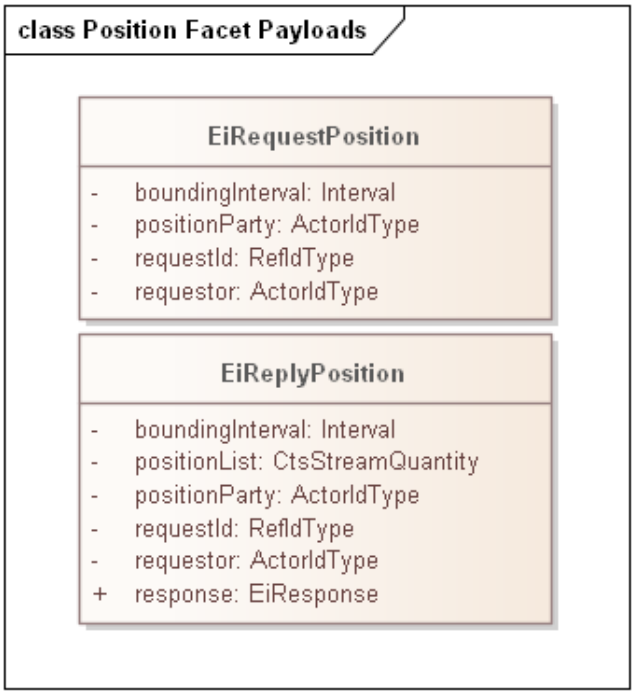
781 EDITOR'S NOTE: This draft shows a CtsStream with quantity as the attached value. Issues relating to
782 differing interval duration are not addressed—all elements of the stream share the duration and the
783 stream has the explicitly stated start time.

784 The attributes are shown in the following section.

785 *Table 8-2: Position Attributes*

786 8.5 Payloads for the Position Facet

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



788

789 *Figure 8-2: UML Class Diagram of Payloads for the Position Facet*

790

791 *Table 8-3: Attributes of Position Facet Payloads*

Attribute	Meaning	Notes
Bounding Interval	The [closed] time interval for which position information is requested. The first positionList Stream element starts at or after the start of the Bounding Interval	
Position Party	The Party whose position is being requested.	EDITOR'S NOTE: implicit market context; work in progress
Request ID	A reference to this payload	Standard throughout Energy Interoperation and its profiles
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
Position List	A CtsStreamQuantity containing the positions for Position Party.	Each CtsStream interval that is completely contained within the Bounding Interval will have a value associated (signed integer, zero permitted).

Attribute	Meaning	Notes
Response	An EiResponse. Will indicate failure if Requestor is not authorized to access position information for Position Party for any of the requested intervals.	

8.6 Notes on the Position Facet—EDITOR'S NOTE—Delete when integrated

```
Some notes on Position and Delivery:

A position is always about comparing what you got, to what you need to have.
As such, it must be similar to Delivery. A position is concerned with the
total amount under contract, not the prices. There are two options here:

Option 1, Duration specified
Get Position 1-3 PM, duration Hourly
Sum up all the Hourly contracts (buy or sell) between 1-2
Sum up all the Hourly contracts (buy or sell) between 2-3

(one could presumably do this for other Durations as well)

Option 2, no Duration specified
Get Position 1-3 PM, duration, no duration specified
Sum up all the contracts (buy or sell) between 1-3
Note granularity of smallest interval transacted (say 5 minutes)
Report as in Option 1 using derived granularity of 5 minutes

Presumably taking a greater delivery than contracted for in any interval must
be paid for. That is the primary reason for having a position: to compare
Position to Delivery. Markets will likely have some notion of a spot price
that will be charged. This may not be simple: if multiple Actors are taking
over-delivery, perhaps the spot price (and penalty if any) is more similar to
the price from a double auction. The last small transaction is likely under-
priced.

Is there a Delivery ALARM, "My position was 22 kW, but there were only 7 kW to
be had, and I am filing a complaint now?

I suspect that most markets will end up measuring delivery twice at each node,
goes-ins and goes-outs. Without that, the market will have trouble converging
on anything other than jitter.

The text in this section for WD09 is based on work done in EML-CTS at
https://github.com/EnergyMashupLab/eml-cts
```