

eMarket Data Exchange Specification

This document is the data exchange specification for the eMarket web service application. This document is designed to assist ISO New England (ISO-NE) Market Participants in the development of their web services interface to the eMarket application.

Version 10.3
July 2015

Trademarks

"**ESCA**" and "**HABITAT**" are registered trademarks of ALSTOM Grid Inc. "**e-terra**" is a registered trademark and/or service mark of E-Terra, LLC, licensed for use by ALSTOM Grid Inc. in connection with its e-terra family of products and services.

Other product and company names in these materials may be trademarks or registered trademarks of other companies, and are the property of their respective owners. They are used only for explanation and to the respective owners' benefit, without intent to infringe.

Contents

Purpose of this Document.....	viii
Scope and Prerequisite Knowledge	viii
Structure of this Document.....	viii
References and Additional Information	ix
Change Summary	x
1. Web Service Overview.....	1
1.1 Web Service Design.....	1
1.2 Accessing the eMarket Web Service.....	1
1.3 Participant Roles.....	2
1.4 Demand Bid Subaccounts.....	3
2. SOAP Messages	5
2.1 Submit and Query Responses	5
2.2 Format and Construction.....	6
2.3 Delete Operations	8
2.3.1 Explicit Web Service Delete Operations.....	8
2.3.2 Explicit Hourly Data Delete Operations	9
2.3.3 Implicit Delete Operations	10
2.4 Optionality and Nillability	11
2.4.1 Submit Messages with Nillable Elements.....	12
2.4.2 Submit Messages with Optional (Response Only) Elements	12
2.4.3 Submit Messages with Optional Attributes.....	12
2.4.4 Query Messages with Optional Elements	12
2.4.5 Query Response Messages with Optional Elements	13
2.4.6 Query Response Messages with Optional Attributes	13
2.4.7 Query Response Messages with Nillable Elements.....	13
2.5 Query Filters.....	13
2.6 Sample SOAP Message Format	14
2.7 Submittal and Query Response Symmetry	15
2.8 Query Response Format.....	17
3. Data Restrictions and Validation	20
3.1 Data Type Validation.....	21
3.1.1 Data Types.....	21
3.1.2 Handling Times	24
3.2 Submission Deadlines.....	26
3.2.1 Energy Market Timeline	26
3.2.2 Regulation Market Timeline	27
3.2.3 Fuel Price Adjustments Submittal Timeline	28
3.2.4 Rules for Timeline Checks	29

3.3 Mitigation Threshold Checking	30
3.4 Effective Dates of Data	31
4. Common Web Services	32
4.1 Messages.....	32
4.1.1 Query Message.....	32
4.2 Node	35
4.2.1 Query Message.....	35
4.3 Node Definition.....	37
4.3.1 Query Message.....	37
4.4 Portfolio.....	39
4.4.1 Submit Message	39
4.4.2 Delete Message.....	41
4.4.3 Query Message.....	42
4.5 Schedule Type	44
4.5.1 Query Message.....	44
5. Asset-Related Demand (ARD) Web Services	46
5.1 ARD	50
5.1.1 Submit Message	50
5.1.2 Query Message.....	52
5.2 ARD By Participant	56
5.2.1 Query Message.....	56
5.3 ARD Hourly	59
5.3.1 Submit Message	59
5.3.2 Query Message.....	61
5.4 ARD Portfolio	65
5.4.1 Submit Message	65
5.4.2 Delete Message.....	65
5.4.3 Query Message.....	65
5.5 ARD Ramp Rate	66
5.5.1 Submit Message	66
5.5.2 Query Message.....	68
5.6 ARD Ramp Rate Hourly	70
5.6.1 Submit Message	70
5.6.2 Query Message.....	72
5.7 ARD Schedule.....	75
5.7.1 Submit Message	75
5.7.2 Delete Message.....	76
5.7.3 Query Message.....	77
5.8 ARD Schedule Bid	79
5.8.1 Submit Message	79
5.8.2 Query Message.....	81
5.9 ARD Schedule Bid Hourly	83

5.9.1 Submit Message	83
5.9.2 Query Message.....	85
5.10 ARD Schedule Detail	88
5.10.1 Submit Message	88
5.10.2 Query Message.....	89
5.11 ARD Schedule Selection.....	92
5.11.1 Submit Message	92
5.11.2 Query Message.....	93
6. Demand Web Services	95
6.1 Demand Bids.....	95
6.1.1 Submit Message	96
6.1.2 Query Message.....	101
6.2 Demand Portfolio	105
6.2.1 Submit Message	105
6.2.2 Delete Message.....	105
6.2.3 Query Message.....	105
6.3 Demand By Participant	106
6.3.1 Query Message.....	106
6.4 Detailed Demand By Participant	111
6.4.1 Query Message.....	111
6.5 SubAccount.....	118
6.5.1 Query Message.....	118
6.6 Load Response Offer	120
6.6.1 Submit Message	120
6.6.2 Delete Message.....	122
6.6.3 Query Message.....	124
6.7 Load Response Result.....	126
6.7.1 Query Message.....	126
7. Generation Web Services.....	128
7.1 Generation By Participant	132
7.1.1 Query Message.....	132
7.2 Generation Portfolio	135
7.2.1 Submit Message	135
7.2.2 Delete Message.....	135
7.2.3 Query Message.....	135
7.3 Schedule	136
7.3.1 Submit Message	136
7.3.2 Delete Message.....	137
7.3.3 Query Message.....	138
7.4 Schedule Bid.....	140
7.4.1 Submit Message	140
7.4.2 Query Message.....	144

7.5 Schedule Bid Hourly.....	146
7.5.1 Submit Message	146
7.5.2 Query Message.....	152
7.6 Schedule Detail	154
7.6.1 Submit Message	154
7.6.2 Query Message.....	159
7.7 Schedule Times Hourly	162
7.7.1 Submit Message	162
7.7.2 Query Message.....	165
7.8 Unit Schedule Selection	167
7.8.1 Submit Message	167
7.8.2 Query Message.....	168
7.9 Unit.....	170
7.9.1 Submit Message	170
7.9.2 Query Message.....	172
7.10 Unit Hourly	176
7.10.1 Submit Message	176
7.10.2 Query Message.....	180
7.11 Unit Ramp Rate.....	183
7.11.1 Submit Message	183
7.11.2 Query Message.....	185
7.12 Unit Ramp Rate Hourly	187
7.12.1 Submit Message	187
7.12.2 Query Message.....	189
7.13 Fuel Price Adjustment	191
7.13.1 Submit Message	191
7.13.2 Query Message.....	193
7.14 Fuel Price Adjustment Unit Hourly	198
7.14.1 Query Message.....	198
7.15 Fuel Price Adjustment Withdrawal	201
7.15.1 Submit Message	201
7.16 Fuel Price Adjustment Attachment	202
7.16.1 Submit Message	202
7.16.2 Delete Message	203
7.16.3 Query Message.....	204

8. Regulation Web Services 209

8.1 Regulation Portfolio.....	211
8.1.1 Submit Message	211
8.1.2 Delete Message	211
8.1.3 Query Message.....	211
8.2 Regulation Offer	212
8.2.1 Submit Message	212
8.2.2 Query Message.....	214
8.3 Regulation Hourly.....	218

8.3.1 Submit Message	218
8.3.2 Query Message.....	223
9. Public Web Services.....	227
9.1 Bid Types	227
9.1.1 Query Message.....	227
9.2 Binding Limits.....	229
9.2.1 Query Message.....	229
9.3 Interface Limits.....	231
9.3.1 Query Message.....	231
9.4 Net Tie Schedules.....	233
9.4.1 Query Message.....	233
9.5 Pricing	235
9.5.1 Query Message.....	235
9.6 System Summary.....	237
9.6.1 Query Message.....	237
Appendix A. Web Service Client – C#	239
Appendix B. Web Service Client – Java	243

About This Document

This document describes the SOAP messages and the Authentication and Authorization process used to exchange market data between a Market Participant and eMarket through the web service. This document explains how to access the eMarket web service, lays out the format and construction of SOAP messages used to exchange data, and briefly describes the Authentication and Authorization methods used to ensure security.

Purpose of this Document

This guide is designed to assist Market Participants develop personal interfaces that interact and exchange market data with the eMarket web service. This guide will help Market Participants comprehend and construct the market data messages essential for data exchange with eMarket.

This guide describes all the submit/query messages that can be constructed to exchange market data between a Market Participant and eMarket. This guide lays out required and/or optional data in messages from Market Participants, describes reports and other market data messages that can be requested from eMarket. This guide also describes the purpose and construction of SOAP messages, the adopted format for data exchange between Market Participants and eMarket. Additionally, this guide gives a brief description of the Authentication and Authorization process used to ensure security of market data exchange.

Scope and Prerequisite Knowledge

This document is intended to be used by all ISO New England Market Participants as an aid in developing new interfaces as well as assisting in the upgrade/re-design of existing interfaces.

Users of this Guide should be familiar with Extensible Markup Language (XML), Web Services, HTTP/HTTPS protocols, ISO New England's governing documents, business rules and operating procedures as well as have a working understanding of how the energy market works and functions in New England. Refer to the *References and Additional Information* section for helpful links.

Structure of this Document

- Chapter 1 gives an overview of the eMarket web service including eMarket web service design and access, participant roles, and authentication and authorization
- Chapter 1.4 describes SOAP messages; from their construction, to the format they are displayed in this document
- Chapter 3 describes restrictions and validations put on data submitted to eMarket

- Chapters 3.4 through 9 are devoted to explaining the functionality of the operations/messages that exchange market data between a Market Participant and eMarket

References and Additional Information

All of the data exchanges using the eMarket web services are described by operations that are defined in the eMarket WSDL file: [eMktMUI.wsdl](#) and by messages defined in the eMarket XSD file: [eMktMessages.xsd](#).

Additional information about Extensible Markup Language (XML), Web Services, and other helpful information can be found at the following links:

- XML –
 - <http://www.w3.org> > XML Technology
 - <http://www.w3schools.com> > Learn XML
- Web Services –
 - <http://www.w3.org> > Web Service Technology
 - <http://www.w3schools.com> > Learn Web Services
- SOAP –
 - <http://www.w3schools.com> > Learn SOAP

ISO New England governing documents include the Transmission, Markets & Services Tariff, ISO New England Manuals and Operating Procedures. They can be found at the following location:

- <http://www.iso-ne.com> > **Rules & Procedures**

Change Summary

Rev	Trans #	Date	Comments
0.1		08/02/10	Initial draft based on E1286.1 [cdo]
1.0	908E1323.1	08/11/10	Prepared for initial high level review. [krm]
2.0	908E1323.2	10/09/10	Incorporate changes by KC upon review of version 1.0. Additional edits and content change to reflect version 5 of the MUI DDN. [css]
3.0	908E1323.3	11/16/10	Added the LRMWType data type. [css]
4.0		01/26/11	<ul style="list-style-type: none"> Changed all Time data type specifications for hourly data to a DateTime data type specification to accommodate shortcomings of .NET applications. Changed the description of the data elements returned in response to the ARD Hourly query message. Introduced numerous changes to deal with the distinct differences between fixed demand bids and price-sensitive demand bids, including increasing the number of price-sensitive demand bids per hour from 9 to 10. Also made changes for Inc/Dec demand bids, in particular limiting the number of bids per hour to 50. [css]
5.0		02/17/11	<ul style="list-style-type: none"> Added the CostType data type to cater for Startup and NoLoad costs. Changed the resolution and range of the FactorType data type to accommodate actual Factor values. [css]
6.0	908E1323.6	11/17/11	<ul style="list-style-type: none"> Added section on offer parameter checking against Market Monitoring thresholds Incorporate changes to language proposed by ISO New England [css]
6.1		12/12/11	<ul style="list-style-type: none"> Replaced all occurrences of <soapenv:Envelope> xmlns:... with <soapenv:Envelope xmlns:... Specified the order in which data will be returned in queries for multi-segment or multi-value data: GetARDHourly, GetARDRampRate, GetARDScheduleBid, GetDemandBid, GetRegUnitHourly, GetScheduleBid, GetUnitHourly, GetUnitRampRate
6.2	908E1323.8	02/01/12	<ul style="list-style-type: none"> Changed the "Nil" attribute of the MaxConsumption and MinConsumption elements to No in §5.8.1.2 Added a sample in §6.1.1.3 that illustrates how to submit demand bids of various types at the same or different Pnodes within a single SubmitDemandBid message. Changed the "Nil" attribute of the RTHighOperLimit element to No in §7.7.1.2 Corrected validation rule in §7.7.1.4 and §7.10.1.4 for MaxDailyEnergy when unit self schedules for at least one hour. Clarified the EcoMax vs CSO validation rule in §7.7.1.4 to emphasize that a warning is issued when the rule is violated.

Rev	Trans #	Date	Comments
			<ul style="list-style-type: none"> Added the EcoMax vs CSO validation rule in §7.10.1.4 because it equally applies to hourly updates of EcoMax. Corrected the nested limit validation rule in §7.10.1.4 to make it identical to the rule expressed in §7.7.1.4. Clarified the rule for validating the setting of the MustRun in §7.10.1.4
6.3	908E1323.9	02/29/12	<ul style="list-style-type: none"> Change §7.6.1.2 to make MaxDailyStarts and RTHighOperLimit elements nillable.
6.4		03/15/12	<ul style="list-style-type: none"> Corrected the description of the purpose for Messages and how to query them in §4.1.1.1 Corrected the validation rule for the consumption limits of dispatchable ARDs in §5.3.1.4 and §5.10.1.4 Clarified in §8.2.1.2 that submittal of regulation offers for generator's that are not regulation resources are simply ignored. Clarified in §8.2.2.2 that requests for regulation offer data for generator's that are not regulation resources simply return nothing. Clarified in §8.3.1.1.2 that submittal of regulation hourly updates for generator's that are not regulation resources are simply ignored. Clarified in §8.3.2.2 that requests for regulation hourly updates data for generator's that are not regulation resources simply return nothing. Corrected the description of the validation performed on the value of MaxDailyEnergy when one is submitted in the Schedule Detail data - §7.6.1.4 and §7.10.1.4 Added description of the warning message issued if MinRun or MinDown time value submit is a Schedule Detail message is greater than 1 for Claim 10 and Claim 30 capable units - §7.6.1.4
6.5	908E1323.10	04/24/12	<ul style="list-style-type: none"> Added description of the warning message issued if a unit's submitted ramp rate value exceeds the demonstrated capability of the value in §7.9.1.4 Added validation rule for ramp rate values in §5.1.1.4, §5.5.1.4, §7.9.1.4 and §7.11.1.4 Added specification of maximum number of elements that can be submitted in a message or listed in a query message
6.6	908E1323.11	05/08/12	<ul style="list-style-type: none"> Added sections §7.6.1.5 and §7.10.1.5 that describe the checking of generating unit offer parameters against the Market Monitoring thresholds.
6.7	908E1323.12	08/31/12	<ul style="list-style-type: none"> Clarified validation rules for Demand Bids in §6.1.1.4 Relaxed the requirement that a Unit ID be specified when querying unit schedules (§7.3.3); when an ID is not specified, the query returns data for every schedule of every unit that belongs to the Participant with which the issuer of the request is associated.

Rev	Trans #	Date	Comments
7.0		04/25/13	<ul style="list-style-type: none"> Updated §3.2 to reflect the changes to the DA timeline. Added statement on when the cleared results of the Day Ahead Market are available to be downloaded in §9 Corrected the Data Type of all StartupCost (Hot, Intermediate, Cold), NoLoadCost, CondenseHourlyCost data elements to be CostType, Decimal 10.2
8.0	908E1323.13	12/23/13	<ul style="list-style-type: none"> Updated to reflect changes introduced by the enhancement of the Regulation Market and for alternative technology regulating resources Replaced the SubmitRegUnitHourly message with two new messages: SubmitRegFinancialHourly and SubmitRegPhysicalHourly Renamed GetRegUnitHourly to GetRegHourly since it also applies to resources other than units Added a new portfolio type: "Regulation" for regulation resources Renamed and changed GetRegUnitHourlyResponse to GetRegHourlyResponse because there are regulation resources that are not units and because more data is now returned Updated the SubmitUnit and GetUnit messages to remove the RegHighDefault and RegLowDefault elements
8.1	908E1323.14	02/19/14	<ul style="list-style-type: none"> Added a section for messages that are applicable to only Regulation Resources and moved all the regulation messages from the Generation section to this new section. Added description of rules that govern regulation offers in the Submission Deadlines section. Numerous editorial corrections and clarifications.
8.2	908E1323.15	04/04/14	<ul style="list-style-type: none"> Revised §2.3.2 and §8.3.1 to discuss explicit deletes of Regulation hourly updates. Various other editorial corrections
8.3	908E1323.16	04/15/14	<ul style="list-style-type: none"> Minor editorial corrections
8.4	908E1323.17	05/16/14	<ul style="list-style-type: none"> Changed the data type for regulation offers from <i>MktPriceType</i> to <i>PriceType</i> and thereby prevent the submittal of negative offers.

Rev	Trans #	Date	Comments
8.5	908E1323.18	07/08/14	<p>Several changes to support Hourly Markets (aka Energy Market Offer Flexibility).</p> <ul style="list-style-type: none"> • Removed all elements for the enhancement of the Regulation Market and for alternative technology regulating resources from Regulation operations • Revised discussion of deletions in §2.3. Now includes an explicit operation to delete hourly data. Each impacted message also includes discussion of explicit hourly data deletion. • Removed “1 to 5 repetitions” reference for all QueryFilters in SOAP examples. Only a single day can be queried now. • Added LRMWOutType, MktPriceType, and PercentType to Data Types table in §3.1.1 • Cleaned up numerous SOAP message examples that had erroneous and/or extraneous data • Introduced new messages and modified existing messages to view and modify fuel price and fuel blends • Added description for new messages added as part of Hourly Markets: ARD Ramp Rate Hourly, ARD Schedule Bid Hourly, Unit Ramp Rate Hourly, Schedule Bid Hourly, and Schedule Times Hourly • Modified descriptions for messages that had elements modified, added, or removed as part of Hourly Markets: ARD Hourly, ARD Schedule Bid, ARD Schedule Detail, Demand Bid, Schedule Bid, Schedule Detail, Unit, Unit Hourly
9.0	908E1323.19	07/15/14	<p>Major Revision # for Hourly Markets (aka Energy Market Offer Flexibility). Changes in this revision include:</p> <ul style="list-style-type: none"> • Updated §3.2 to reflect changes to eMarket deadlines. In particular added section on fuel price adjustments. • Updated §3.3 to reflect changes to definition and description of Reference schedules. • Updated note on delete web services in §5, §7 and §8 to describe new explicit delete functions. • Updated Figure 1, Figure 2, Figure 3 • Updated all of Appendix B • Updated validations rules throughout document • Various editorial and other corrections

Rev	Trans #	Date	Comments
9.1	908E1323.19	09/05/14	<p>No functional changes; only corrections and clarifications:</p> <ul style="list-style-type: none"> • Added section heading 3.2.1, 3.2.2, 3.2.3 and 3.2.4. • Regulation Market Timeline – Intraday Re-Offers can be submitted up to 30 minutes before the start of the hour, not 5 minutes before as previously stated. • Replaced rules for validation of offer and bid prices with language using terms defined from Market Rule 1 in §5.8.1.4, §5.9.1.3, §6.1.1.4, §7.4.1.4, §7.5.1.3 • §6.6.1.4 – Changed language specifying when the DRTP is published to be consistent with Market Rule 1. • §7.4.1.4 and §7.5.1.3 – Corrected validation rule for Startup and No Load cost parameters. • §7.5.1.1 – Schedule Bid Hourly submit message – added the additional information that must be submitted with the message and emphasized that this is the only message that can be used in the RT Intraday Re-Offer period to submit offer updates for the current operating day and updates to only the schedule being used in the Real Time market • §7.5.1.3 – Corrected validation rule for Startup and No Load cost parameters. • §7.5.1.4 – Fixed language governing validation of startup and no load costs during the RT Intraday Re-Offer period. • §7.6.1.4, §7.8.1.4, §7.10.1.4, §8.2.1.4 – Expanded on the limit nesting rule of limits. • §7.7.1.5 – Added validation rule for the sum of the time parameters. • §7.13 – Added reference to the eMarket Users Guide for a complete detailed description of Fuel Price Adjustment requests. • §7.13.1.4 – Replaced detailed definition of FPA request limit with a reference to Market Rule 1 where it is defined. • §7.13.2.4, §7.14.1.4 – documented additional FPA states
9.2		10/30/14	<p>Corrections of minor errors in the SOAP message and sample XML message of submittals in</p> <ul style="list-style-type: none"> • §7.4.1.3 for Schedule Bid operations • §7.5.1.2 for Schedule Bid Hourly operations • §7.7.1.3 for Schedule Times Hourly operations

Rev	Trans #	Date	Comments
10.0	908E1323.20	12/15/14	<p>Major Revision # to re-instate all elements for the enhancement of the Regulation Market and for alternative technology regulating resources. Changes in this revision include:</p> <ul style="list-style-type: none"> • In §3.1.1, added the <i>DerateFactorType</i> data type. • Updated §3.2 to reflect changes to the eMarket deadlines for the Regulation Market. In particular, changes to the close hour of the Regulation Market Intraday Re-Offer Period and to what can be submitted during this period, and finally changes to the number of minutes before the beginning of the hour that intraday re-offers can be submitted. • Updated §8.2 Regulation Offer to <ul style="list-style-type: none"> ◆ Replace the single regulation offer price with the three part regulation offer prices that can be submitted in the new Regulation Market. ◆ Add the validation rules for ATRR regulation offers. ◆ Modify validation rules for the financial regulation offers. ◆ Describe all the new parameters returned in response to the query of regulation offers. • Updated §8.3 Regulation Hourly to <ul style="list-style-type: none"> ◆ Replace the single regulation offer price with the three part regulation offer prices that can be submitted in the new Regulation Market. ◆ Add the validation rules for ATRR regulation offers. ◆ Modify validation rules for the financial regulation offers.
10.1	908E1323.20a	04/09/15	<p>Minor revision to change Regulation “Capability” element from MWFlowType to MWType, consistent with implementation, in the response type associated with GetRegOffer and GetRegHourly methods.</p>
10.2	908E1323.21	06/25/15	<p>Revision to document the Divisional Accounting changes for Participants who register subaccounts in the ISO-NE Customer and Asset Management System (CAMS) and other editorial changes/corrections.</p> <ul style="list-style-type: none"> • Added §1.4 to introduce the concept of demand bid subaccounts • Added §3.4 to emphasize the date order in which default bid and offer data must be submitted. • Described changes for subaccounts made to the eMarket Demand Bid XML operations discussed in chapter 6

Rev	Trans #	Date	Comments
10.3		07/22/15	<p>Added an attribute to the GetDemandBid, GetDemandByParticipant and GetDetailedDemandByParticipant query operations to give Participants explicit control over whether to include subaccount information in the response message. Revised Sections:</p> <ul style="list-style-type: none"> • 1.4 Demand Bid Subaccounts to emphasize what is returned by the query operations when the attribute is specified and when it is not specified. • 6.1 Demand Bids to add note that users of Participants with registered subaccounts can submit demand bids in distinct subaccounts. • 6.1.1 SubmitDemandBid, – to specify that the Participant can specify the Default subaccount in addition to his named subaccounts in CAMS and that they can use the SubmitDemandBid with the “delete” attribute to delete bids for a terminated subaccount. • 6.1.2 GetDemandBid, 6.3.1 GetDemandByParticipant, 6.4.1 GetDetailedDemandByParticipant – to specify that the Participant must now use the “subAccounts” attribute to explicitly control whether the query is to return subaccount information and to clarify the validation rules.

1. Web Service Overview

This chapter gives an overview of web service concepts associated with the eMarket web service, how to access the eMarket web service, and Market Participant roles.

1.1 Web Service Design

The eMarket web service, or programmatic interface, is described by operations that are defined in the eMarket WSDL. These operations also describe the messages that are used to transfer market data between a Market Participant and the eMarket application through the interaction of the Market Participant's client interface and eMarket's web service. The web service operation messages use SOAP format for data transmission, which is discussed in more detail in chapter 2.

All web service operations follow a request/reply pattern that is typical of HTTP(S) communication. A request may contain a message that modifies (or submits) data or it may contain a message that queries for data. A reply contains a message that is either: 1) a confirmation of data modification, including warnings when applicable, 2) an error, or 3) the response to a query.

Any web service operation that allows data to be modified will have a corresponding web service operation that allows for that data to be queried. In many cases, this web service operation pairing will have a submit message that contains the exact same data as a query response message. This relationship is referred to as having symmetrical messages. However, not all submit/query response messages are symmetrical. There are cases in which more data is returned in a query response message than can be contained in a submittal message.

There are some web service operations that simply have query messages, and they are used for the sole purpose of requesting specific data from the eMarket application. A more in depth description of the symmetry of web service operation messages used by the eMarket web service can be found in section 2.7 entitled Submittal and Query Response Symmetry.

1.2 Accessing the eMarket Web Service

The eMarket web services are based on SOAP (version 1.1) over HTTPS (version 1.1). A client application that accesses eMarket can be written in nearly any modern enterprise technology and language, such as Java, .NET, C++, Ruby, PHP, etc. The web service is accessible to authorized Participants through an ISO New England published URL, and all submittals and queries are serviced via the same URL. ISO New England provides limited support for two industry leading technologies viz. Microsoft.Net and Java by providing sample implementations.

Authentication to the interface is done using digital certificates issued by ISO New England. Potential users need to register with the Customer Support department within ISO New England in order to obtain valid certification for access to the programmatic interface. Once the company is registered with ISO New England, the Participant's company appoints a Security Administrator (a member of their own staff) who in turn grants permissions (or roles) to the potential users of eMarket or any other market application from that Participant company. Users contact their own Security Administrator for roles to be opened on their certificates (which reside on their own computers) Refer to the [Customer and Asset Management System Application Group Roles](#) document on the ISO web page for [User Guides](#).

Questions or inquires about certificates for eMarket access should be addressed to the Customer Support Department at ISO New England.

1.3 Participant Roles

Roles define the web service operations/messages that a Market Participant user can use to submit or query data. Distinct roles are typically associated with a class of market resources, e.g. Asset Related Demands (ARD), Units (Generation), Demands, Regulation (Generators and Alternative Technology Regulating Resources) or Public Reports. For example, an ARD Read/Write role assigned to a Participant user would allow the user to use all the ARD web service operations, but would not allow the use of the Demand web service operations.

Roles are also divided into two sub-types; "Read Only" and "Read/Write" (also known as "Update"). Read Only roles restrict the user to Query messages associated with their role's web service operations, while Read/Write roles allow the use of both Submit and Query messages. For example, a Participant user with a Read Only ARD role can only use Query messages associated with ARD operations, while a Participant user with a Read/Write ARD role can use all the ARD web service operation messages. Public web service operations can be used by any Participant with any eMarket role. Use of a web service by a user without a corresponding role will throw an error (see fault handling in section 2.1).

Refer to the [Customer and Asset Management System Application Group Roles](#) document on the ISO-NE Web for a complete list and description of the eMarket roles.

Some eMarket users can represent more than one Participant; these are referred to as "agents". In such cases, the user needs to specify the Participant that they are representing. This is done by setting the **party** attribute of the root element, as in the example below:

```
<SubmitARD party="participant1">  
...  
</SubmitARD>
```

In this example, the logged in user can submit the message on behalf of "participant1". If a user is defined to represent more than one Participant, omitting the party attribute would throw a fault (see fault handling in section 2.1).

eMarket users that can represent more than one Participant must be assigned roles by the Security Administrator of each Participant that they can represent.

1.4 Demand Bid Subaccounts

Participants with registered subaccounts in the ISO-NE Customer and Asset Management System (CAMS) for their demand bids may now use these subaccounts when their users submit demand bids into the Market using the eMarket application. A new query message, GetSubAccount, is provided for Participants to get the list of subaccount names recognized by eMarket.

Notes: The subaccount feature is optional in the eMarket application. For Participants who do not register subaccounts in CAMS or do not want to use their subaccounts registered in CAMS, the eMarket Web Services are essentially unchanged in that the same XML messages are used to submit and query demand bid data and the XML query response messages remain the same. Existing eMarket client applications can be used without change. **It is however recommended that the client applications be updated and re-built using the new eMarket WSDL.**

For Participants with registered subaccounts in CAMS, a default subaccount named "Default" is automatically created in the eMarket application. This subaccount is used when a Participant with registered subaccounts does not specify a subaccount when submitting demand bids.

Participants with registered subaccounts who want to retrieve demand bid data on a subaccount basis must explicitly specify this in their XML query message. They may then specify subaccount names when submitting query messages for demand bids and thereby filter the data returned. When the subaccount is not specified, eMarket returns demand bid data for all of the Participant's subaccounts. If a Participant does not explicitly specify that they want subaccount information in response to their XML query message for demand bids, **only demand bids for the "Default" subaccount will be returned.**

Note: Since Participant users can submit demand bids for several days into the future and it may be possible for the Participant to terminate a subaccount after bids for future days have been submitted, it is important to remember that the bids are not deleted when the subaccount is terminated; they remain in place and will be considered when clearing the

Day-Ahead Market on those future days. ***It is the responsibility of the Participant users to delete the bids that may be in place for future days when a subaccount is terminated.***

2. SOAP Messages

This chapter describes SOAP messages and how they are used in eMarket. This chapter describes constructing SOAP messages, restrictions on data submitted, and SOAP format/documentation in this guide.

SOAP is a specification for exchanging information involving Web Services such as eMarket. SOAP messages are constructed using Extensible Markup Language (XML) as a structure to store data. This XML structure is wrapped in a SOAP envelope that carries processing instructions and descriptions of the data for interpretation by an interface or Web Service. For reference, these operations are all described in eMktMUI.wsdl.

2.1 Submit and Query Responses

Each message sent to eMarket is an "all or nothing" event. eMarket will use database transactions, such that a commit will only occur on successful processing of an entire SOAP envelope. If an exception occurs while processing a message, a fault will be sent to the user with the appropriate error messages, and the transaction will be rolled back. When a fault is returned, none of the data submitted in the original message was saved.

```
<MUIFault>
  <Error>
    <Reason>Some error message</Reason>
  </Error>
  <Error>
    <Reason>Some other error message</Reason>
  </Error>
</MUIFault>
```

Query messages return a wide variety of information, and as such they do not have a standard response message unless the message returns a fault as shown above.

Conversely, each submit message that is sent to the eMarket application has a standard response message that confirms the message was received and processed. The response message contains a transaction ID that is used to track/indicate the confirmation of the message submitted. The standard response message to a submit message is shown below. It contains an optional Warning element that is used to indicate potential problems with the data that was submitted.

Standard Submit Response:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitConfirmation transactionId="?">
      <!--Zero or more repetitions:-->
      <mes:Warning>
        <mes:Reason?</mes:Reason>
      </mes:Warning>
    </mes:SubmitConfirmation>
  </soapenv:Body>
</soapenv:Envelope>
```

2.2 Format and Construction

SOAP messages are XML formatted structures wrapped in a SOAP envelope. XML formatted messages are organized with elements and attributes, and the structure looks very similar to HTML formatted messages. A simple XML message is shown below:

XML Message

```
<note>
  <to>Mike</to>
  <from>John</from>
  <heading>Reminder</heading>
  <body>Don't forget fishing this weekend!</body>
</note>
```

Messages used in the eMarket application look similar to the example above, however the XML formatted message is wrapped in a SOAP envelope. This SOAP envelope is shown below:

SOAP Envelope

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    ...
    (XML FORMAT HERE)
    ...
  </soapenv:Body>
</soapenv:Envelope>
```

An XML formatted message and a SOAP envelope¹ come together to form a SOAP message that is used to exchange data between a third party interface and a Web Service. With respect to eMarket, interfaces are constructed and operated by Market Participants, while the Web Service is operated by ISO New England. An example of a complete SOAP

¹ The samples within this document reference SOAP 1.1 as indicated by the namespace "http://schemas.xmlsoap.org/soap/envelope/".

message, an XML formatted message with a SOAP envelope, that is documented in this guide is shown below:

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARRampRate party="?">
      <!--1 or more repetitions:-->
      <mes:ARRampRate day="?" ID="?">
        <mes:RampRateCurve>
          <!--0 to 10 repetitions:-->
          <mes:RampRatePoint rate="?" MW="?"/>
        </mes:RampRateCurve>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:ARRampRate>
    </mes:SubmitARRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

In examples like the one shown above, question marks indicate elements or attributes that should be populated with values if not designated as optional. The "mes:" that appears in each element of the XML body is a namespace that eMarket uses to avoid naming conflicts.

The example message above demonstrates the format of messages that exists throughout this document. This format is used because it shows the complete SOAP message template used to exchange data with the eMarket web services. Each of the SOAP messages in the "Full SOAP Format" section of the web service operations can be used to submit/query data from eMarket, as long as the elements and attributes contain valid values. This document uses a color convention for distinguishing the various parts of the SOAP message. The color convention for items within these messages is shown below:

Element (brown)	Element Value (black)	Attribute (magenta)	Attribute Value (blue)	Comment (green)
--------------------	--------------------------	------------------------	---------------------------	--------------------

The eMarket data that is contained within the SOAP body is defined in the eMktMessages.xsd file. Within this document, the individual XML elements and attributes that comprise the eMarket data are described in Chapters 4 – 8 and using a table format with the following column headings:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
------	------	----------------------	-------------------	----------

Opt. – Indicates whether an element or attribute is optional

Nil. – Indicates whether an element or attribute is nillable

Element or Attribute – The name of element or attribute

Data Type; Format – Specifies the data type and format for the data

Comments – Specific information about the element or attribute

2.3 Delete Operations

There are two approaches for deleting data within eMarket – explicit and implicit.

Explicit deletions are accomplished using one of the following two methods:

1. Web Service Delete Operations – these are operations that are explicitly defined in the WSDL (e.g. DeletePortfolio).
2. Hourly Data Delete Operations – this type of deletion removes data for a specific hour and is accomplished by setting the “delete” attribute in a Submit message to true.

Implicit delete operations occur by virtue of removing or replacing all data within a sub-element in an XML message.

These concepts are further detailed below.

2.3.1 Explicit Web Service Delete Operations

The following explicit web service delete operations are provided:

1. DeleteARDSchedule
2. DeleteFuelPriceAdjustmentAttachment
3. DeleteLoadResponseOffer
4. DeletePortfolio
5. DeleteSchedule

These operations will remove existing objects in the database that match the deletion criteria (typically this is the ID field). They are documented in a "Delete Format" subsection of the type of resource. Each delete operation will return a DeleteConfirmation message that is very similar to the SubmitConfirmation message.

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes>DeleteConfirmation transactionId="?">
      <!--Zero or more repetitions:-->
      <mes:Warning>
        <mes:Reason?</mes:Reason>
      </mes:Warning>
    </mes>DeleteConfirmation>
  </soapenv:Body>
</soapenv:Envelope>

```

Note that "explicit partial intraday delete" operations are not supported. For example, a message is not defined to delete just the 4th hour of some hourly profiles (e.g. SubmitUnitHourly). To delete just that hour, a Submit message with an empty container for that hour must be submitted. That is the function of an implicit delete. For Regulation hourly updates an explicit form of the implicit delete must be used. These are discussed in the next section.

2.3.2 Explicit Hourly Data Delete Operations

Deleting data for a specific hour is done by setting the "delete" attribute as shown in the sample below.

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDHourly party="MyParty">
      <mes:ARDHourly day="2014-04-12" ID="123">
        <mes:HourlyProfile>
          <mes:HourlyValues time="2014-04-12T00:00:00-04:00"
delete="true"/>
        </mes:HourlyProfile>
      </mes:ARDHourly>
    </mes:SubmitARDHourly>
  </soapenv:Body>
</soapenv:Envelope>

```

This would be repeated for each hour that requires data to be deleted. The delete attribute is defined as optional with the default being false.

The following messages support explicit deletion of hourly data and are documented in the relevant subsection where the web service is described:

- SubmitARDHourly
- SubmitARDHourlyRampRate
- SubmitARDScheduleBidHourly
- SubmitDemandBid
- SubmitRegFinancialHourly

- SubmitRegPhysicalHourly
- SubmitScheduleBidHourly
- SubmitScheduleTimesHourly
- SubmitUnitHourly
- SubmitUnitRampRateHourly

2.3.3 Implicit Delete Operations

To delete all elements of a ramp rate curve, one needs to submit an empty container element. This is best shown by the example below.

Below is a full SOAP template for the SubmitARDRampRate message. *Note that the RampRateCurve element can contain 0 to 10 repetitions of the RampRatePoint element.*

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDRampRate party="?">
      <!--1 or more repetitions:-->
      <mes:ARDRampRate day="?" ID="?">
        <!--Optional:-->
        <mes:RampRateCap>?</mes:RampRateCap>
        <mes:RampRateCurve>
          <!--0 to 10 repetitions:-->
          <mes:RampRatePoint rate="?" MW="?">
        </mes:RampRateCurve>
        <!--Optional:-->
        <mes:ShortName>?</mes:ShortName>
      </mes:ARDRampRate>
    </mes:SubmitARDRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

This XML instance shows how one would delete all hours for the given resource and day.

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDRampRate party="MyParty">
      <mes:ARDRampRate day="2010-09-30" ID="123">
        <mes:RampRateCurve/>
      </mes:ARDRampRate>
    </mes:SubmitARDRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

The following messages allow implicit deletes and are documented in the relevant subsection where the web service is described:

- SubmitARDRampRate
- SubmitUnitRampRate

2.4 Optionality and Nillability

Some messages contain elements and/or attributes that are optional. Element and attribute optionality is indicated in the XML Schema by specifying a "minOccurs=0" for elements and "use=optional" for attributes. The XSD snippets below show examples of each of these.

```
<complexType name="BindingLimitsOut">
  <sequence>
    <element name="BindingLimit" type="msg:BindingLimit"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="day" type="date" use="required"/>
</complexType>
```

```
<complexType name="ScheduleType">
  <sequence>
    .
    <element name="SchedTypeID" type="long"
      minOccurs="1" maxOccurs="1"/>
    .
  </sequence>
  <attribute name="schedID" type="long" use="optional"/>
</complexType>
```

Some messages contain elements that may have a null value. As an example, the SubmitARD message has Claim10 and Claim30 values that may be null. This is shown in the snippet below.

```
<complexType name="ARDIn">
  <sequence>
    <element name="Claim10" type="msg:MWType"
      minOccurs="1" maxOccurs="1" nillable="true"/>
    <element name="Claim30" type="msg:MWType"
      minOccurs="1" maxOccurs="1" nillable="true"/>
    <element name="DefaultStatus"
      type="msg:ResourceStatusTypeEnum"
      minOccurs="1" maxOccurs="1"/>
    <element name="RampRateDefault" type="msg:MWType"
      minOccurs="1" maxOccurs="1"/>
  </sequence>
  <attributeGroup ref="msg:DayAndIdAttrGroup"/>
</complexType>
```

This example instance shows how an element that is null is expressed.

```

<SubmitARD>
  <ARD day="?" ID="?">
    <Claim10 xsi:nil="true"/>
    <Claim30 xsi:nil="true"/>
    <DefaultStatus>Economic</DefaultStatus>
    <RampRateDefault>100</RampRateDefault>
  </ARD>
</SubmitARD>

```

The next sections describe the conventions for handling optional elements and attributes and elements that can be null.

2.4.1 Submit Messages with Nillable Elements

Any element that is marked as `nil="true"` will be interpreted as meaning "set the value in the database to NULL". The NULL value will be effective according to standard effective dating. A large number of Submit type messages contain elements that can be nil.

2.4.2 Submit Messages with Optional (Response Only) Elements

These are optional elements associated with symmetrical messages (see section 2.7). They are intended to contain values only with Query Response type messages. Should a value be provided on submittal, it will be ignored and not saved to the database. Nearly half of the Submit type messages contain Response Only, optional elements.

2.4.3 Submit Messages with Optional Attributes

There are 2 classes of optional attributes defined for Submit type messages. *Note that one cannot specify an XML attribute to be nillable in the same manner that an XML element allows.*

1. Writeable – If a value is provided for a writeable, optional attribute, then that value will be saved to the database. If a value is not provided, it will be interpreted as meaning "set the value in the database to NULL". The NULL value will be effective according to standard effective dating. The SubmitDemandBid message has a writeable, optional attribute used only when the bid type is fixed.
2. Read Only – If a value is provided for an optional attribute, it is used as a key to find an existing object in the database. The existing object will then be updated with values provided with other elements of the message. The absence of a value for an optional attribute indicates that a new instance of this object should be created in the database. Only the SubmitARDSchedule and SubmitSchedule messages use Read Only, optional attributes.

2.4.4 Query Messages with Optional Elements

There is no special handling of optional elements necessary for Query messages. Values provided for optional elements are used to filter the data that is returned.

2.4.5 Query Response Messages with Optional Elements

Many Query Response messages also contain optional elements. Some are the "Response Only" types associated with symmetrical messages. This type of optional element will always have a value in the Query Response. Those elements that are not "Response Only" are typically top level elements within a response element. The ARD element shown below is an example:

```
<element name="GetARDResponse">
  <complexType>
    <sequence>
      <element name="ARD" type="msg:ARDOut"
        minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

This construct supports a query that doesn't return any values that match the query criteria. If there are no values for the given element, the element tag will not appear in the XML.

2.4.6 Query Response Messages with Optional Attributes

Any Query Response message that contains optional attributes will have values for that attribute in the XML if the database has corresponding values. If the database does not have a corresponding value, the attribute tag will not appear in the XML.

2.4.7 Query Response Messages with Nillable Elements

Any Query Response message that contains nillable elements will specify nil="true" in the XML element if the database has a NULL value for that element.

2.5 Query Filters

When querying data from eMarket, a user will submit a query request message that contains filters to limit the data that is queried. Typical query filters are Day and resource ID (for unit/asset/node), though other filter criteria is possible depending on the nature of the data being queried. An example of a query message is shown below.

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNodeDefinition party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetNodeDefinition>
  </soapenv:Body>
</soapenv:Envelope>
```

In this example, Day is a required element and ID is an optional element meaning that it does not have to be included in the submitted message. This message, and all query messages that have optional elements, return a scope of data based on their optional elements.

Optional elements in query messages are intended to narrow the amount of data returned, allowing one or more units, assets, etc., to be selected instead of all being selected. In the example above, nodes are being queried for their definitions. If the optional element ID is not included when this message is submitted, eMarket will return all of the nodes in the market with their definitions which is a large volume of data. If the optional element is included, eMarket will return node definitions only for the nodes for which IDs are supplied. Some query messages will have multiple optional elements, allowing for a customized unit, hour, node, etc. selection. This same principle applies to all query messages, optional elements narrowing the scope of data returned.

2.6 Sample SOAP Message Format

The "Sample of Submittal (or Query)" message that follows each "Full SOAP Format" message in this document uses the full SOAP message, however the elements/attributes have made up values to show how data could be submitted. These sample messages also have the SOAP envelope, namespaces, comments, and party attributes removed in order to reduce the overall length of this document. The samples below show the differences between a full SOAP message and its corresponding sample message. The full SOAP message below has the elements and attributes that are removed in its corresponding sample message highlighted in white.

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitPortfolio party="?">
      <!--1 or more repetitions:-->
      <mes:Portfolio name="?" type="?">
        <!--Zero or more repetitions:-->
        <mes:Resource name="?" resourceID="?">
        </mes:Portfolio>
      </mes:SubmitPortfolio>
    </soapenv:Body>
  </soapenv:Envelope>

```

This full SOAP message would have this sample message documented below.

```

<SubmitPortfolio>
  <Portfolio name="Portfolio1" type="Generation">
    <Resource name="Unit1" resourceID="5678"/>
  </Portfolio>
</SubmitPortfolio>

```

The SOAP envelope, namespaces, comments, and party attributes that were removed in this example are common throughout this document; they will always be removed in sample messages. The sample message above will not process without the highlighted portions of the full SOAP message, with the exception of the lines of comments (indicated in green). Lines of comments are inserted in the full SOAP message to indicate optionality and element repetition, and a message will process with or without them.

Both of the above examples are taken from the Submit Message in the Portfolio operation of the Common Web Services chapter to show an actual instance of this formatting.

Sample messages may or may not have the elements/attributes that are optional included for the common reason of saving document length. It is important to note that a message will process with or without optional elements and attributes included, even though optional elements/attributes may not be shown in a sample SOAP message.

2.7 Submittal and Query Response Symmetry

In general, data that can be submitted is viewable in a near-identical XML structure in a Query Response message. For example, the XML below shows sample data for a Submit message of a Portfolio operation.

```

<SubmitPortfolio>
  <Portfolio name="Portfoliol" type="Generation">
    <Resource name="Unit1" resourceID="5678"/>
    <Resource name="Unit2" resourceID="142"/>
    <Resource name="Unit3" resourceID="9521"/>
  </Portfolio>
</SubmitPortfolio>

```

Except for the outermost XML element name, the response to a Query Portfolio message is identical to the Submit Portfolio message.

```

<GetPortfolioResponse>
  <Portfolio name="Portfoliol" type="Generation">
    <Resource name="Unit1" resourceID="5678"/>
    <Resource name="Unit2" resourceID="142"/>
    <Resource name="Unit3" resourceID="9521"/>
  </Portfolio>
</GetPortfolioResponse>

```

This symmetrical relationship occurs for many messages, though there are some exceptions whereby the query response contains more data than was submitted. For example, in the example above, either the resource name or the resource ID need to be provided when creating or revising a portfolio as shown below. However both will always be returned in the response to the Query Portfolio message.

```

<SubmitPortfolio>
  <Portfolio name="Portfoliol" type="Generation">
    <Resource name="Unit1"/>
    <Resource resourceID="142"/>
    <Resource name="Unit3"/>
  </Portfolio>
</SubmitPortfolio>

```

In other cases the additional data cannot be submitted. It is stored in the Market database and comes from data requested when a Market Participant registers with ISO New England. This data is static or changeable through Customer Service processes, but is available for download to provide a complete picture of the asset.

Messages that have "significant" differences between submittal and query response have two message definitions defined (i.e. one for submittal and one for query response).

Several Submit/Query Response pairs differ only by a description and/or name field. These pairings use a common message definition for both submittal and query response, with the description or name field being defined as optional. The Submit ARD Ramp Rate message and its corresponding Query ARD Ramp Rate Response show this relationship.

```

<SubmitARDRampRate>
  <ARDRampRate day="2010-07-07" ID="1234">
    <RampRateCurve>
      <RampRatePoint rate="2.0" MW="10"/>
      <RampRatePoint rate="1.0" MW="20"/>
    </RampRateCurve>
  </ARDRampRate>
</SubmitARDRampRate>

```

The same data shown above will be returned in response to a Query ARD Ramp Rate message; the only difference being the outermost element and the returned ShortName element. For cases like this, in the comments section of the elements/attributes tables associated with each full SOAP message, a "Read Only" statement will be used to identify the "Response Only" elements. An example of this is shown below in the response message to an ARD Ramp Rate query.

```

<GetARDRampRateResponse>
  <ARDRampRate day="2010-07-07" ID="1234">
    <RampRateCurve>
      <RampRatePoint rate="2.0" MW="10"/>
      <RampRatePoint rate="1.0" MW="20"/>
    </RampRateCurve>
    <ShortName>ShrtNme</ShortName>
  </ARDRampRate>
</GetARDRampRateResponse>

```

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the ARD schedule. Read Only

2.8 Query Response Format

Operations in this document are formatted in two distinct ways. One format is specifically for operations that have both submit and query messages, while the other is for operations that simply have a query message. The main difference between these two formats, aside from the submit/query message format having a submit message, is how the response message is documented in the Data Returned section. The submit/query message format shows the data returned by a response message, and the response message elements of that message. The query message only format has the data returned by a response message, as well as the full SOAP response message with a sample response message. An example of each Data Returned section format is shown below.

Query Message Only Data Returned Section:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the node
No	No	Name	String; 30	The node name

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	BusNodeID	Long	The bus node ID
No	No	BusNodeName	String; 30	The bus node name
No	No	Factor	FactorType; Decimal 7.6	The node factor

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNodeDefinitionResponse>
      <!--Zero or more repetitions:-->
      <mes:NodeDefinitions day="">
        <!--Zero or more repetitions:-->
        <mes:NodeDefinition ID="" name="">
          <mes:BusNodeID?</mes:BusNodeID>
          <mes:BusNodeName?</mes:BusNodeName>
          <mes:Factor?</mes:Factor>
        </mes:NodeDefinition>
      </mes:NodeDefinitions>
    </mes:GetNodeDefinitionResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetNodeDefinitionResponse>
  <NodeDefinitions day="2010-07-07">
    <NodeDefinition ID="12345" name="Hub">
      <BusNodeID>09876</BusNodeID>
      <BusNodeName>MW HUB</BusNodeName>
      <Factor>0.89</Factor>
    </NodeDefinition>
  </NodeDefinitions>
</GetNodeDefinitionResponse>
```

Submit/Query Message Data Returned Section:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the ARD schedule. Read Only
No	No	RampRateCap	MWType; Decimal 6.1	The ramp rate cap for an asset. Read Only

Response Elements

```
<GetARDRampRateResponse>
  ...
  <ShortName>SchdNme</ShortName>
  <RampRateCap>111.1</RampRateCap>
  ...
</GetARDRampRateResponse>
```

Submit/Query operations that have "significant" differences between submittal and query response messages are documented in a special way in the Data Returned section of each message. An example of "significant differences" is the Submit ARD message/Query ARD Response pairing. The Submit ARD message accepts 7 values on submittal while Query ARD Response has 22 values that are returned. In response messages like this, only the delta elements are shown in the Response Elements sample in of the Data Returned section. If a submit message/query response message pair has the same elements and attributes, the Response Elements sample in each operation will have no elements or attributes, only the outermost message tags. An example of this is shown above in the "Submit/Query Message Data Returned Section". This example shows two elements in the Response Elements sample, meaning that the only difference between the submit message and the returned query response message were those two returned elements.

3. Data Restrictions and Validation

In order for data to submit without error, basic validations must first be met. This chapter describes the validations/restrictions for data and messages that need to be met in order to submit messages to the eMarket application.

Basic validations are restrictions on data values submitted, ensuring the data is submitted at the right time according to market rules, and is submitted in the correct format/range. The following sections, "Data Type Validation", "Submission Deadline Validation" and "Mitigation Threshold Checking", highlight these basic validations. This chapter is intended to outline the universal data restriction/validations necessary for submission of a message to the eMarket application.

Other restrictions/validations will need to be met as well, such as adhering to market rules defined by ISO New England. These additional restrictions/validations beyond basic validation will be documented in each of the Web Service operations that have submit messages, in sections entitled "Additional Value Restrictions". This is done because the market rule restrictions and validations can differ between messages and operations.

It is recommended that the Data Type Validation and Submission Deadline Validation sections be printed and used in parallel with constructing any web service operation messages. This will make referencing validations and value restrictions quick and simple.

3.1 Data Type Validation

A basic type of data restriction/validation involves Data Types. A Data Type has restrictions associated with it that are used for submittal messages. A Data Type defines/restricts the range and format of numbers and strings. Data Types are defined and used in the eMarket XSD file, which is linked to the eMarket WSDL file that provides the Web Service operation messages a Market Participant submits/receives.

An individual Data Type is associated with a specific element or attribute of a web service operation message. Each attribute and element that is in a given message is listed, in this document, in a table located in the "Mandatory and Optional Fields" section of the message. This table not only shows the elements and attributes of an operation, but the data type and format associated with the element or attribute listed as well. In the tables, Data Types are displayed in the following format: Data Type; Format. An example of an element/attribute table that can be found throughout this document is shown below:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ID	Long	The ID of the asset
No	No	Day	Date; yyyy-mm-dd	The market day
No	Yes	Claim10	MWType; Decimal 6.1	The claimed capability of the asset that can be achieved from an offline state in 10 minutes
No	Yes	Claim30	MWType; Decimal 6.1	The claimed capability of the asset that can be achieved from an offline state in 30 minutes
No	No	DayAheadRequired	Boolean; true (1), false (0)	Whether or not the asset is included in the Day-Ahead Market
No	No	DefaultStatus	Enumerated String; Economic, MustRun, Unavailable	The availability/operating status of the asset
No	No	RampRateDefault	MWType; Decimal 6.1	The default ramp rate to use for the asset if an hourly value does not exist

This table shows the element or attribute as well as the data type associated with it. The "Format" of the Data Type; Format column of the table is a short description of the type of data that is acceptable for submission.

3.1.1 Data Types

The following are the most common Data Types, and a description of the "Format" that follows them.

- In a Date data type, the format yyyy-mm-dd tells that the date must be submitted in a year-month-day format with dashes between numbers. For example, an element with "Data Type;Format", "Date; yyyy-mm-dd" could submit 2010-07-07(See section 3.1.2 below on Handling Times).
- In a DateTime data type, the format yyyy-mm-ddThh:mm:ss-hh:mm tells that the time must be submitted in an hour:minute:second format with an attached hour:minute adjustment for a time zone preceded with a date in the same format as a Date data type and an intervening "T" character. An example of a time submitted for four o'clock P.M. on July 7, 2010 with a four hour time zone offset is 2010-07-07T16:00:00-04:00 (See section 3.1.2 below on Handling Times).
- In an Enumerated String data type, the format tells the exact strings that can be submitted; however only one string listed can be submitted in an element. For example, an element having "Data Type;Format" "Enumerated String; Economic, MustRun, Unavailable" could have either "Economic", "MustRun", or "Unavailable" entered in the element as the chosen string.
- In a MWType, PriceType, or other data type that is followed by "Decimal x.x", the "Format" tells the number of digits that can be submitted, as well as to what decimal place. The x to the left of the decimal place tells the total number of digits that can be submitted, while the x to the right tells (of the total digits) how many digits can follow the decimal. For example, "Decimal 5.2" values can be five digits long, two of which must be beyond the decimal point. This would be represented as a value such as: 546.53. A "Decimal 6.5" could have six digits, five of which must be beyond the decimal point; 3.14159.
- In a Long data type, there is no given format because a long is an integer number that can be submitted and has no decimals.
- In a Boolean data type, either "true" or "false" is entered. This is indicated one of four ways in an element or attribute; true, false, 1, 0. True and the number one are equivalent, while false and the number zero are equivalent.
- In a String data type, "String" is followed by a number. This number indicates the number of characters that can be entered into the element or attribute. For example "String; 8" could be at most eight characters long. An example of this is "NameUnit" which is eight characters long.

The table below shows the most common data types and their descriptions. The "Description" column shows the exact strings, values, and/or ranges of data that can be submitted. There are additional enumerated strings data types that are not listed in this table; they will appear in the elements/attributes tables associated with each message if they are needed.

Data Type	Description
Boolean ¹	Boolean. values are; true, 1, false, 0
CostType	Decimal using 10.2 format and range is $0 \leq X \leq 9999999.99$. Unit in dollars
Date ¹	General format; yyyy-mm-dd
DateTime ¹	Hour beginning time. The general format is yyyy-mm-ddThh:mm:ss-hh:mm; time must use 24-hour format and not be negative.
DerateFactorType	Decimal using 3.2 format and range is $0.00 \leq x \leq 1.00$
ElapsedTimeType	Time in hours expressed as decimal using 5.2 format and range is $0 \leq X \leq 999.99$
FactorType	Decimal using 7.6 format and range is $-2.000000 \leq X \leq 2.000000$
Long ¹	Integer value used primarily for ID's
LRMWType	Decimal using 6.3 format and range is $0.1 \leq X \leq 999.999$, Units in Megawatts (MW)
LRMWOutType	Decimal using 6.3 format and range is $0.0 \leq X \leq 999.999$, Units in Megawatts (MW) used for cleared offers
MktPriceType ²	Decimal using 6.2 format and range is $-9999.99 \leq X \leq 9999.99$, Units in dollars
MinInterruptDurationType	Time in hours expressed as a single digit integer value
MWFlowType	Decimal using 6.1 format and range is $-99999.9 < X \leq 99999.9$, Units in Megawatts (MW)
MWRateType	Decimal using 5.1 format and range is $0 < X \leq 9999.9$, Units in Megawatts (MW)
MWType	Decimal using 6.1 format and range is $0 \leq X \leq 99999.9$, Units in Megawatts (MW)
PercentType	Decimal using 3.0 format and range is $0 \leq X \leq 100$
PriceType ²	Decimal using 6.2 format and range is $0 \leq X \leq 9999.99$, Units in dollars
ServiceFactorType	Decimal using 6.2 format and range is $0 \leq X \leq 9999.99$
String ¹	General purpose string, the number following the string in Data Type; Format is the maximum string length
Enumerated String	Enumerated String. Values vary, and are listed in each element/attribute table as "Enumerated String" data type appears. These types are itemized in the definition of each message.

Note 1: these types are native XML types. All other types are specifically defined for eMarket.

Note 2: PriceType values may be restricted to a greater extent than that specified by the XML. These more restrictive values are defined in the "Additional Value Restrictions" section of each message description where applicable.

3.1.2 Handling Times

Within the XML schema, the data types shown in the table below are used to represent date and time values. The lexical representation for these data types is specified by the ISO 8601 standard. The table shows the common notation for each, though the standard allows for great flexibility in formats.

Data Type	Lexical Representation	Example
Date	yyyy-mm-dd	2011-02-01
DateTime	yyyy-mm-ddThh:mm:ss-hh:mm	2011-02-01T11:00:00-05:00

Note that the DateTime format includes the time zone indicator and the example shows time in Eastern Standard Time. XML submissions to the ISO New England eMarket are required to specify a time zone indicator to avoid confusion during the daylight savings transition periods.

Adding "-04:00" and "-05:00" to the end of DateTime representations will specify Eastern Daylight and Eastern Standard respectively. All samples in documentation will show timestamps referencing Eastern Time, to mirror the market.

The samples below show the GetPricesResponse message for the two daylight saving time transition days.

Fall Transition

```
<GetPricesResponse>
  <Prices day="2010-11-07">
    <NodePrices ID="4000" name=".H.INTERNAL_HUB">
      <HourlyPrice time="2010-11-07T00:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T01:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T01:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T02:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T03:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T04:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T05:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T06:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T07:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T08:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T09:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T10:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T11:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T12:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T13:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T14:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T15:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T16:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T17:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T18:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T19:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T20:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T21:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T22:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-11-07T23:00:00-05:00" price="32.62"/>
    </NodePrices>
  </Prices>
</GetPricesResponse>
```

```
</NodePrices>
<Prices/>
</GetPricesResponse>
```

Spring Transition

```
<GetPricesResponse>
  <Prices day="2010-03-14">
    <NodePrices ID="4000" name=".H.INTERNAL_HUB">
      <HourlyPrice time="2010-03-14T00:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T01:00:00-05:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T03:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T04:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T05:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T06:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T07:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T08:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T09:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T10:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T11:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T12:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T13:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T14:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T15:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T16:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T17:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T18:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T19:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T20:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T21:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T22:00:00-04:00" price="32.62"/>
      <HourlyPrice time="2010-03-14T23:00:00-04:00" price="32.62"/>
    </NodePrices>
  <Prices/>
</GetPricesResponse>
```

Note: All time values are assumed to be **hour beginning** values, not hour ending.

3.2 Submission Deadlines

Another basic type of value restriction/validation involves submission deadlines. Submission deadlines are placed not on individual elements or attributes, but on the entire message submitted to the eMarket application. Submission deadlines are market restrictions/validations established by Market Rule that define deadlines by which messages must be submitted. If a message is submitted after a submission deadline that message and all of its contents will not be processed.

3.2.1 Energy Market Timeline

The following submission deadlines for energy offers/bids exist in eMarket:

Deadline	Description
Day-Ahead Market for two days from now and beyond – Open and Close hours	Normally, the market opens at 18:30 and closes at 23:59 Eastern time each day for the market day two days from now. Energy offer/bid submittals can be made between 18:30 and 23:59 every day, for the day after tomorrow and up to nine days in the future.
Day-Ahead Market for tomorrow and beyond – Open and Close hour	Normally, the market opens at 00:00 and closes at 10:00 Eastern time each day for the next day's market. Energy offer/bid submittals can be made between midnight (00:00) and 10:00 every day, up to ten days in the future. If a bid or offer is to be used in the Day-Ahead Market, it must be submitted by the 10:00 deadline on the day before the operating day. The markets database is unavailable for submittal of any energy offers/bids for tomorrow and beyond between Day-Ahead Market Close and Re-Offer Period Open, which is when the Day-Ahead commitment software is running. Note: the RT Intraday Re-Offer Period for today is however open, so the markets database is available for submittal of intraday re-offers of financial parameters.
Re-Offer Period Open hour	Normally, the Re-Offer Period opens daily as soon as the Day-Ahead market results are approved and posted, but no later than 13:30 Eastern time. While the Re-Offer Period is open, financial energy offers by generators and energy bids by Asset-Related Demands can be submitted for tomorrow's market day. For those assets that were not committed in the Day-Ahead Market for tomorrow, changes to their non-financial offer/bid parameters (limits, times, etc.) for tomorrow's market day can also be submitted during this period of time. Note: During the Re-Offer Period energy offers/bids for two days from now and up to nine days in the future can be made without restriction.

Deadline	Description
Re-Offer Period Close hour	<p>Normally, the re-offer period closes daily at 14:00 Eastern time. The markets database is unavailable for submittal of any energy offers between Re-Offer Period Close and RT Intraday Re-Offer Open, which is when the Real Time commitment software is running to establish and finalize the operating plan for tomorrow.</p> <p>Note: the RT Intraday Re-Offer Period for today is however open, so the markets database is available for submittal of intraday re-offers of financial parameters.</p>
Real Time (RT) Intraday Re-Offer Period – Open and Special rules for Close hour	<p>The RT Intraday Re-Offer period opens at 18:30 Eastern time the day before the operating day and implicitly closes at 22:30 of the operating day. During this period of time, hourly updates of energy offers for future hours of the operating day can be submitted. Offers for the next hour can be submitted up to 30 minutes before the beginning of the hour. Therefore</p> <ul style="list-style-type: none"> • After 23:30 the day before the operating day, hourly updates for the <u>first</u> hour of the operating day can no longer be submitted • After 22:30 of the operating day, hourly updates for the operating day can no longer be submitted. <p>During the RT Intraday Re-Offer period, only <u>hourly updates of financial energy offers</u> for the operating day can be submitted; that is to say changes to incremental energy offers/bids (MW/Price pairs) and under strict rules, changes to generator’s Startup and No Load Cost offers. All other components of an asset’s offer/bid for the operating day cannot be changed.</p>

There are special conditions where the above times may change such as extenuating circumstances when the closure of the Day-Ahead Market must be delayed. Market Participants are notified whenever these times are altered; the notifications are posted as special notices on the ISO website. Market Participants can sign up with the Customer Support Department at ISO New England to receive these notices by email.

3.2.2 Regulation Market Timeline

The following submission deadlines for regulation offers exist in eMarket:

Deadline	Description
Regulation Market for two days from now and beyond – Open and Close hour	<p>The market opens at 18:30 and closes at 23:59 Eastern time each day for the market day two days from now. Regulation offers, daily and hourly updates, can be submitted between 18:30 and 23:59 every day, for the day after tomorrow and up to nine days in the future.</p> <p>Same as the offers/bids for the Energy market.</p>

Deadline	Description
Regulation Market for tomorrow and beyond – Open and Close hour	<p>The market opens at 00:00 and closes at 14:00 Eastern time each day for the next day’s market. Regulation offers, daily and hourly updates, can be submitted between midnight (00:00) and 14:00 every day, up to ten days in the future.</p> <p>Note that 14:00 is the close of the Energy market Re-Offer Period.</p>
Regulation Market (RM) Intraday Re-Offer Period – Open and Special rules for Close hour	<p>The RM Intraday Re-Offer period opens at 14:00 Eastern time the day before the operating day and implicitly closes are 22:55 of the operating day. During this period, hourly updates of regulation offers for future hours of the operating day can be submitted. Offers for the next hour can be submitted up to 5 minutes before the beginning of the hour. Therefore</p> <ul style="list-style-type: none"> • After 23:55 the day before the operating day, hourly updates for the <u>first</u> hour of the operating day can no longer be submitted • After 22:55 of the operating day, hourly updates for the operating day can no longer be submitted. <p>During the Intraday Reoffer period,</p> <ul style="list-style-type: none"> • For regulation-capable generators, <u>hourly updates of only the regulation financial offer parameters</u> (Regulation Service Price, Regulation Inter-Temporal Opportunity Capacity Cost, and Regulation Other Capacity Cost) can be submitted. • For Alternative Technology Regulation Resources (ATRR), hourly updates of both the regulation financial and physical (Availability for Regulation, Automatic Ramp Rate, Regulation High and Low Limits) can be submitted. <p>What this means is that Participants can submit hourly updates of their regulation financial offers at any time for future hours of today and for any hour of tomorrow. Participants of ATRRs can also submit hourly updates of their regulation physical offers at any time for future hours of today and for any hour of tomorrow.</p>

3.2.3 Fuel Price Adjustments Submittal Timeline

The following submission and processing deadlines and rules for Fuel Price Adjustment (FPA) requests exist in eMarket:

- FPA requests can be submitted at any time with a start hour for any future hours of today and any hour of tomorrow. FPA requests for the next hour of today must be submitted at least 30 minutes before the start of the hour.
- FPA requests are processed within one minute after being received except for two periods of time where the processing is suspended:
 - From the closure of the Day-Ahead Market and the opening of the Re-Offer Period for tomorrow, the processing of FPA requests for

tomorrow is suspended while the commitment software is running to clear the tomorrow's Day-Ahead Market.

- From the closure of the Re-Offer Period and the opening of the RT Intraday Re-Offer Period for tomorrow, the processing of FPA requests for tomorrow is suspended while the commitment software is running to establish and finalize the tomorrow's operating plan.

FPA requests for tomorrow submitted while processing is suspended remain in "Pending" state until processing resumes.

3.2.4 Rules for Timeline Checks

Every message submitted to the eMarket application has an embedded timestamp that cannot be edited, which tells eMarket when the message was submitted. Almost all the submission messages have the ability to enter day and sometimes time information; either for the past or the future. When this day and time is entered, a deadline validation is set based on whether the entered dates and times have already passed. If they have passed, there is no validation needed. Validation restrictions occur only when the day and time entered are in the future. Demand bids and generation offers submitted for the Day-Ahead Market are examples of messages that contain dates/times that can be submitted for the future.

The time against which these validations are done will be the time that the eMarket application receives the message for processing, which is the earliest time that eMarket can recognize a valid document.

Consider the following examples (assumes a submission deadline of 10:00):

- A user sends a submittal message for the next day at 09:59.45 AM. eMarket receives the complete document at 09:59:48 AM. Assume the validation and database processing required to submit this transaction is significant, such that the transaction does not complete until 10:00:12 AM. The transaction succeeds, because eMarket received the complete request at 09:59:48 AM, before the cut-off window.
- A user sends a submittal message for the next day at 10:01.45 AM. eMarket receives the complete document at 10:01:48 AM. eMarket will reject this submission because it was received after the cut-off time.
- A user sends a submittal message for the next day at 09:59.45 AM. Due to unusual network latency, eMarket receives the complete document at 10:00:13 AM. eMarket will reject this submission because it was received after the cut-off time, even though it was sent before the cut-off time. eMarket has no way of knowing that this document was sent before the cut-off time, and therefore can only rely on the time the document was received.

3.3 Mitigation Threshold Checking

All unit offer parameters (and only those) submitted by the Participants are checked against threshold values calculated by the ISO Internal Market Monitor according to the Market Rules. For some parameters, like the physical parameters (startup and notification times, transition times, minimum run time, minimum down time) and cost parameters (cold, intermediate and hot startup costs and no load cost), if the offer value submitted exceeds its threshold, the whole offer is rejected. For other parameters, if the offer value submitted exceeds its threshold, the offer is accepted but a warning is returned in the response message indicating that the combined unit offer values are subject to mitigation if they violate the Market Monitor's impact and/or conduct tests when the unit is committed or dispatched.

All of the parameters found in a unit's cost-based Reference schedule are the Reference Level values calculated by the ISO Internal Market Monitor for these parameters. There is one Reference schedule for each fuel that a unit is configured to use.

- Schedule 98 corresponds to the unit's Fuel Index #1. There will always be one of these for each unit.
- Schedule 97 corresponds to the unit's Fuel Index #2, if it is a multi-fuel unit; if it is a single fuel unit, this schedule either does not exist or is ignored if one does exist.
- Schedule 96 corresponds to the unit's Fuel Index #3, if it is a three or four fuel unit; if it is a single or dual fuel unit, this schedule either does not exist or is ignored if one does exist.
- Schedule 95 corresponds to the unit's Fuel Index #4, if it is a four fuel unit; otherwise, this schedule either does not exist or is ignored if one does exist.

Threshold values used by eMarket to validate Participant offers are derived from the unit Reference Levels. For physical offer parameters, the Reference Levels are the same for all fuel indexes that a unit is configured to use, i.e., independent of the fuel. For this reason, the Reference Level values used to derive the threshold values of physical parameters are contained in Reference schedule 98. Because of the fuel blend parameters associated with financial offers (Startup and NoLoad costs, Incremental Energy Offers), the Reference Level values found in all of a unit's Reference schedules are used to derive the threshold values of the financial parameters. The derivation of the threshold values from the reference values is defined in the Market Rule.

3.4 Effective Dates of Data

Much of the data in the eMarket application is time sensitive. As data is submitted, a date for which the data applies is normally required. As a rule, it is not necessary to submit data for each Market day. With “Submit” requests for default bid and offer data², the submitted data is effective starting on the market day specified in the request and for all subsequent days up to the day the data is changed.

Note: Data must be submitted for successive market days. This means that you should not submit data for a future date, say July 10, and then submit data for an earlier date, say July 5, because the latter data submittal will overwrite the data previously submitted.

² SubmitARD, SubmitARDRampRate, SubmitARDScheduleBid, SubmitARDScheduleDetail, SubmitARDScheduleSelection, SubmitScheduleBid, SubmitScheduleDetail, SubmitScheduleSelection, SubmitUnit, SubmitUnitRampRate, SubmitRegOffer

4. Common Web Services

This section defines the submittals and queries that are common among or independent of the ARD, Demand, Generation, Regulation and Load Response functional areas.

4.1 Messages

Messages allow for an ISO-NE market operator or application to communicate pertinent market information to one or more market participants. The message format used enables a market participant to receive priority labeled public and private messages that have been issued by a market operator or application. A public message is a message that is available to all market participants, while a private message is a message directed to a specific set (of one or more market participants).

Note: this functionality is currently not used.

4.1.1 Query Message

4.1.1.1 Purpose of Message

The purpose of this message is to get all the messages associated with a specific privacy, private or public, organized by priority for a given date. Messages have an effective period defined by a start date and a termination date. So this query will return messages whose effective period spans the date specified in the request. Data returned in the response includes the message priority, the message text, and its ID as well as the start date of the message and its type: Public or Private.

4.1.1.2 Mandatory and Optional Fields

To get or query messages, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day desired
No	No	MessageType	Enumerated String; Both, Private, Public	The privacy type of the messages desired

4.1.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetMessages party=""?>
      <!--1 or more repetitions:-->
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <mes:MessageType?</mes:MessageType>
      </mes:QueryFilters>
    </mes:GetMessages>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetMessages>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <MessageType>Private</MessageType>
  </QueryFilters>
</GetMessages>
```

4.1.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	messageType	Enumerated String; Both, Private, Public	The privacy type of the message
No	No	ID	Long	The ID of the message
No	No	Priority	Long	The assigned relative importance of the message
No	No	Text	String; 500	The text contents of the message

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetMessagesResponse>
      <!--Zero or more repetitions:-->
      <mes:Messages day="?" messageType="?">
        <!--Zero or more repetitions:-->
        <mes:Message ID="?">
          <mes:Priority?</mes:Priority>
          <mes:Text?</mes:Text>
        </mes:Message>
      </mes:Messages>
    </mes:GetMessagesResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetMessagesResponse>
  <Messages day="2010-07-07" messageType="public">
    <Message ID="09876">
      <Priority>12345</Priority>
      <Text>message contents</Text>
    </Message>
  </Messages>
</GetMessagesResponse>
```

4.2 Node

Nodes are the locations where ISO New England calculates and publishes Day Ahead and Real Time Market prices. They are also known as Pnodes. Some of these correspond to electrical locations in the market system network model, like the electrical node associated with most generators. Many are virtual locations representing groups or aggregations of electrical nodes.

4.2.1 Query Message

4.2.1.1 Purpose of Message

The purpose of this message is to get an XML report that lists all the pricing nodes (Pnodes) in the market system. The response message will contain a list of all these nodes, including the node name, type, and ID associated with each one. The possible node types are:

- Aggregate – These are virtual locations.
- Bus – These correspond directly to an electrical node.
- Hub – These are virtual locations
- Interface – These typically correspond directly to an electrical node, although some are virtual locations
- Zone – These are virtual locations representing the various load zones in the New England Market network model.
- FiveHundredKV – This node type is **currently not used** in the ISO New England market system.

The report is public, meaning that anyone authenticated to access the eMarket application can make this query.

4.2.1.2 Mandatory and Optional Fields

There are no query filter elements or attributes associated with this request. The "party" attribute of the root element has no impact on the data returned in the response message and can be omitted.

4.2.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNode party="?"/>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetNode/>
```

4.2.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ID	Long	The ID of the node
No	No	name	String; 30	The node name
No	No	type	Enumerated String; FiveHundredKV, Aggregate, Bus, Hub, Interface, Zone	The node type

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNodeResponse>
      <!--Zero or more repetitions:-->
      <mes:Node ID="?" name="?" type="?"/>
    </mes:GetNodeResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetNodeResponse>
  <Node ID="12345" name="MW Node" type="Hub"/>
</GetNodeResponse>
```

4.3 Node Definition

See section 4.2 for a definition of nodes.

4.3.1 Query Message

4.3.1.1 Purpose of Message

The purpose of this message is to get an XML report listing the electrical bus nodes that are aggregated in the virtual Pnodes listed in the XML report obtained with the Node query message. The data returned will contain the day and node IDs used to query, along with the name of these Pnodes and for each of these, the name, ID, and weighting factor of each bus node that is aggregated in the virtual Pnode. The weighting factors define the contribution of the bus node to the market price at the Pnode. Since the weighting factor values of each bus node changes for each market day, the query must specify a market day.

4.3.1.2 Mandatory and Optional Fields

To get or query node definitions, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the node

- An unlimited list of node IDs can be specified. If the node ID element is not specified then the query returns data for all virtual pricing nodes.

4.3.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNodeDefinition party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetNodeDefinition>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetNodeDefinition>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetNodeDefinition>
```

4.3.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the node
No	No	name	String; 30	The node name
No	No	BusNodeID	Long	The bus node ID
No	No	BusNodeName	String; 30	The bus node name
No	No	Factor	FactorType; Decimal 7.6	The node factor

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNodeDefinitionResponse>
      <!--Zero or more repetitions:-->
      <mes:NodeDefinitions day="?">
        <!--Zero or more repetitions:-->
        <mes:NodeDefinition ID="?" name="?">
          <mes:BusNodeID?</mes:BusNodeID>
          <mes:BusNodeName?</mes:BusNodeName>
          <mes:Factor?</mes:Factor>
        </mes:NodeDefinition>
      </mes:NodeDefinitions>
    </mes:GetNodeDefinitionResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetNodeDefinitionResponse>
  <NodeDefinitions day="2010-07-07">
    <NodeDefinition ID="12345" name="Hub">
      <BusNodeID>09876</BusNodeID>
      <BusNodeName>MW HUB</BusNodeName>
      <Factor>0.89</Factor>
    </NodeDefinition>
  </NodeDefinitions>
</GetNodeDefinitionResponse>
```

4.4 Portfolio

Portfolios provide a means to define the list of resources (ARDs, Generators, Regulation Resources³ and Demand Nodes) for which data will be returned in the "by Participant" and the "by Portfolio" XML reports: data for each unique resource listed in **all** the user's portfolios will be returned. They are also used in the eMarket User Interface to create lists of resources that the user can select to submit bids and offers or to examine previously submitted bids and offers. By using portfolios, users can organize resources in a manner suited to their business. Users can create any number of portfolios with each portfolio uniquely named by the user to describe its usage. A given resource (ARD, Pnode, Unit, or ATRR) can be assigned to multiple portfolios. It must be emphasized that portfolios are associated with a user: the person or web application associated with an eMarket user certificate.

4.4.1 Submit Message

4.4.1.1 Purpose of Message

The purpose of this message is to create/edit a portfolio. This message contains data relevant to the setup/modification of a portfolio, such as its type and name, as well as the name and ID of all the resources to list in the portfolio.

4.4.1.2 Mandatory and Optional Fields

To create or modify a portfolio entry, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	name	String; 40	The name of the portfolio
No	No	type	Enumerated String; ARD, Pnode, Generation, Regulation	The type of the portfolio
Yes	No	name	String; 40	The name of the resource
Yes	No	resourceID	Long	The ID of the resource

A limited number of portfolios can be created or edited in one request; refer to the maxOccurs attribute of the SubmitPortfolio message in the eMktMessages.xsd.

The same portfolio name can be associated with any and all portfolio types. In other words, one can create a portfolio "ABC" of type Generation and list one or more generators in it; then one can create a portfolio "ABC" of type ARD and list one or more ARDs in it; then one can create a portfolio "ABC" of type Pnode and list one or more load Pnodes in it; finally

³ Regulation resources are regulation capable generators and alternative technology regulation resources [ATRR].

one can create a portfolio "ABC" of type Regulation and list one or more ATRRs and regulation-capable generators in it. It must however be noted that if one lists a resource in a portfolio of one type, it will automatically be listed in the portfolio of the same name but of a different type if the resource is of the correct type. For example, if one lists a regulation-capable generator in the portfolio "ABC" of type Generation, the generator will automatically be listed in the portfolio "ABC" of type Regulation and its Pnode will automatically be listed in the portfolio "ABC" of type Pnode. *Therefore, it is recommended to create portfolios with unique names for each type if you want to strictly control what is listed in them.*

4.4.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitPortfolio party="">
      <!--1 or more repetitions:-->
      <mes:Portfolio name="" type="">
        <!--Zero or more repetitions:-->
        <mes:Resource name="" resourceID=""/>
      </mes:Portfolio>
    </mes:SubmitPortfolio>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitPortfolio>
  <Portfolio name="Portfolio1" type="Generation">
    <Resource name="Unit1" resourceID="5678"/>
  </Portfolio>
</SubmitPortfolio>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

4.4.1.4 Additional Value Restrictions

- Note that there is no specific operation to delete a resource from a portfolio. This is done as an implicit delete as defined in section 2.3.3.
- Because portfolios are associated with users, the party attribute is not relevant for this message, unless the user is an agent, i.e. a user that can act on behalf of multiple market participants, in which case a party attribute must be provided.
- You can specify the name of a resource and/or the resource ID. If you specify both, the name of the resource is not used; precedence is given to the resource ID.

- With generation and ARD portfolios, only resources belonging to the market participant that the user represents can be listed. If the user is an agent that can act on behalf of multiple market participants, then only resources belonging to the market participant identified by the "party" attribute on the root element can be listed.
- With Demand (Pnode) portfolios, only Hub, Interface and Zone Pnodes should be listed, although any public pricing node (Pnode) can be listed.
- With Regulation portfolios, only regulation-capable units and ATRRs that belong to the market participant can be listed.

4.4.2 Delete Message

4.4.2.1 Purpose of Message

The purpose of this message is to delete/remove a stored portfolio. This message contains data relevant to the deletion of a portfolio, such as the portfolio name and type.

4.4.2.2 Mandatory and Optional Fields

To delete a portfolio, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	PortfolioName	String; 40	The name of the portfolio to be deleted
Yes	No	PortfolioType	Enumerated String; ARD, Pnode, Generation, Regulation	The type of the portfolio to be deleted. If not specified, all types with the provided portfolio name will be deleted.

4.4.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:DeletePortfolio party="?">
      <mes:PortfolioName?</mes:PortfolioName>
      <!--Optional:-->
      <mes:PortfolioType?</mes:PortfolioType>
    </mes:DeletePortfolio>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Delete Submittal

```
<DeletePortfolio>
  <PortfolioName>portfolio1</PortfolioName>
  <PortfolioType>ARD</PortfolioType>
</DeletePortfolio>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

4.4.2.4 Additional Value Restrictions

- Because portfolios are associated with users, the party attribute is not relevant for this message. If a value is provided, it will be ignored.
- Because resources of all types can be assigned to a portfolio of a given name, if the portfolio type is specified only resources of that type will be deleted from the portfolio.

4.4.3 Query Message

4.4.3.1 Purpose of Message

The purpose of this message is to get created portfolios and the resources that are included in each one. Data returned includes the type and name of the portfolio, as well as the resource name and ID for all portfolios of the specified type.

4.4.3.2 Mandatory and Optional Fields

To get or query a portfolio response, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	PortfolioType	Enumerated Strings; ARD, Pnode, Generation, Regulation	The type of portfolio

4.4.3.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetPortfolio party="?">
      <mes:PortfolioType?</mes:PortfolioType>
    </mes:GetPortfolio>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetPortfolio>
  <PortfolioType>ARD</PortfolioType>
</GetPortfolio>
```

4.4.3.4 Data Returned

Except for the outermost tags the data returned in the Query Response matches the data described for the Submittal.

Response Elements

```
<GetPortfolioResponse>  
  ...  
</GetPortfolioResponse>
```

4.5 Schedule Type

Schedules are used to define sets of offer/bid values for a resource: generator or dispatchable ARD. Schedules are of two types:

1. Price-based – These schedules are used by market participants to create sets of offer/bid values for a resource.
2. Cost-based – These are used by ISO New England to create reference and mitigation schedules. Market Participants can view these schedules, but they cannot modify nor delete them.

For each resource, up to 100 schedules can be created, each with a short name, a description and a number. A schedule type is associated with each schedule number; a range of the schedule numbers is reserved for cost-based schedules.

4.5.1 Query Message

4.5.1.1 Purpose of Message

The purpose of this message is to get a XML report listing the type of each of the 100 schedule numbers (0-99) by resource type.

4.5.1.2 Mandatory and Optional Fields

To get or query schedule types, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Category	Enumerated String; ARD, Generation	The category or functional area

4.5.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleType party=""?>
      <mes:Category?></mes:Category>
    </mes:GetScheduleType>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetScheduleType>
  <Category>ARD</Category>
</GetScheduleType>
```

4.5.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	category	Enumerated String; ARD, Generation	The category or functional area
Yes	No	SchedDescription	String; 80	The schedule type description
No	No	SchedTypeName	Enumerated String; Cost, Price	The schedule type name
No	No	SchedTypeNum	Long	The schedule type number

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleTypeResponse>
      <mes:ScheduleTypes category="">
        <!--Zero or more repetitions:-->
        <mes:ScheduleType>
          <!--Optional:-->
          <mes:SchedDescription?</mes:SchedDescription>
          <mes:SchedTypeName?</mes:SchedTypeName>
          <mes:SchedTypeNum?</mes:SchedTypeNum>
        </mes:ScheduleType>
      </mes:ScheduleTypes>
    </mes:GetScheduleTypeResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetScheduleTypeResponse>
  <ScheduleTypes category="Generation">
    <ScheduleType>
      <SchedDescription>Some Description</SchedDescription>
      <SchedTypeName>Cost</SchedTypeName>
      <SchedTypeNum>1</SchedTypeNum>
    </ScheduleType>
  </ScheduleTypes>
</GetScheduleTypeResponse>
```

5. Asset-Related Demand (ARD) Web Services

This section defines the submittals and queries used for Asset Related Demand (ARD) operations.

The table below shows a summary of the ARD messages and indicates if data can be submitted, queried or deleted. It also shows when and how long submitted bid data is effective. As a rule, it is not necessary to submit data for each market day. With most messages, the submitted data is effective starting on the market day specified in the message and for all subsequent days up to the day for which data is submitted in a separate message. Hourly bid data is the exception as it only applies to a specific market day and hour.

Message	Brief Description of Usage	Actions			Data Remains Effective	
		Submit	Query	Delete	Until data is submitted for a future day	Only for the specified day and hour
ARD	Set or view the default bid data for an ARD	X	X		X	
ARD By Participant	View the ARD bids that cleared in the DA Market		X		n/a	n/a
ARD Hourly	Set or view the Hourly Overrides which apply for a market day.	X	X	See Note 1		X
ARD Portfolio	Manage Portfolios of ARDs	X	X	X	n/a See Note 3	n/a
ARD Ramp Rate	Set or view the Ramp Rate Curve. Used when a multi-segment curve is desired rather than the simple slope defined with the ARD message; it overrides the simple slope.	X	X	See Note 1	X	
ARD Ramp Rate Hourly	Set or view the Hourly Ramp Rate Curve Overrides.	X	X	See Note 1		X
ARD Schedule	Manage the Schedule container definitions. The following messages contain the details of the schedules.	X	X	X	X	
ARD Schedule Bid	Set or view the MW/Price pairs for a given schedule	X	X	See Note 2	X	
ARD Schedule Bid Hourly	Set or view the Hourly Overrides for MW/Price pairs for a given schedule	X	X	See Note 2		X

Message	Brief Description of Usage	Actions			Data Remains Effective	
		Submit	Query	Delete	Until data is submitted for a future day	Only for the specified day and hour
ARD Schedule Detail	Set or view the ARD bid data for a given schedule.	X	X		X	
ARD Schedule Selection	Set or view the availability of a schedule for a given market and day.	X	X		X	

Note 1: There is no “Delete” web service operation defined to explicitly delete an ARD Ramp Rate, ARD Ramp Rate Hourly, or an ARD Hourly offer once a set of data is submitted. Nor is there an operation defined to delete only a segment of the multi-segment curve data previously submitted. A whole new set of curve data or hourly data must be submitted. There is however provision in the “Submit” web service to remove a set of curve data by submitting an empty container. Similarly, there is provision in the “Submit” web services for ARD Hourly and ARD Ramp Rate Hourly offers to remove a specific hourly update offer by submitting a message with the “delete” attribute set. See section 2.3 for more information on delete operations.

Note 2: There is no operation defined to delete a particular MW/Price pair from a set of previously submitted bid data. A whole set of MW/Price pairs must be submitted to replace the previously submitted set.

Note 3: Portfolios are not effective dated. The data in a portfolio exists from when the portfolio is created until the portfolio is deleted. The data (asset list) in a portfolio can be updated using the SubmitPortfolio request; note however that the complete list of asset IDs must be submitted to replace the previously submitted list, if any.

When submitting data, it is important to understand the hierarchy of the ARD messages and the implications of data overrides. Figure 1 below shows the hierarchy of the ARD messages used for submittals, where the same bid data appears in different messages, and which messages have precedence over (overrides) the others. [**Note:** only the data elements applicable to this example are shown. See the following sections for the details on the data in each message.]

The ARD message is at the root of the hierarchy. The ARD record itself is initially created by ISO New England with default bid data when the asset is instantiated in the Market system. Participants must update some data values, like the DefaultStatus, using the ARD message.

To participate in the energy market, a Participant must submit at least once an ARD Schedule message and all of the child messages that go with it. The ARD Schedule message defines a schedule container for which Bid and Detail data must be submitted. The ARD Schedule primarily defines the names and type of a schedule for an ARD.

Once the schedule is created, then the following three messages must be used to fill out that schedule for the market:

- ARD Schedule Bid: defines the MW/Price pairs, i.e. energy bid data
- ARD Schedule Detail: defines limits and other bid data
- ARD Schedule Selection: sets the schedule as available or unavailable for the DA and/or the RT market.

Four additional messages allow for overriding either schedule based or default data. Bid data in other messages higher in the hierarchy are overridden by the presence of data in these messages:

- The ARD Ramp Rate message defines a ramp rate curve that overrides the simple slope defined with the ARD message.
- The ARD Ramp Rate Hourly message overrides the ramp rate curve provided with the ARD Ramp Rate message or the simple slope defined with the ARD message (one can specify either an hourly curve or single ramp rate value).
- The ARD Hourly message overrides data for all schedules and defaults for a given day.
 - ARD Hourly MinConsumption and MaxConsumption override for the specified hour of the day the corresponding values defined in the Schedule Detail of all schedules.
 - ARD Hourly ResourceStatus overrides for the specified hour of the day the DefaultStatus defined with the ARD message.
- The ARD Schedule Bid Hourly message overrides MW/Price pairs provided with the ARD Schedule Bid message. This is the only message that can be submitted during the RT Intraday Re-Offer period.

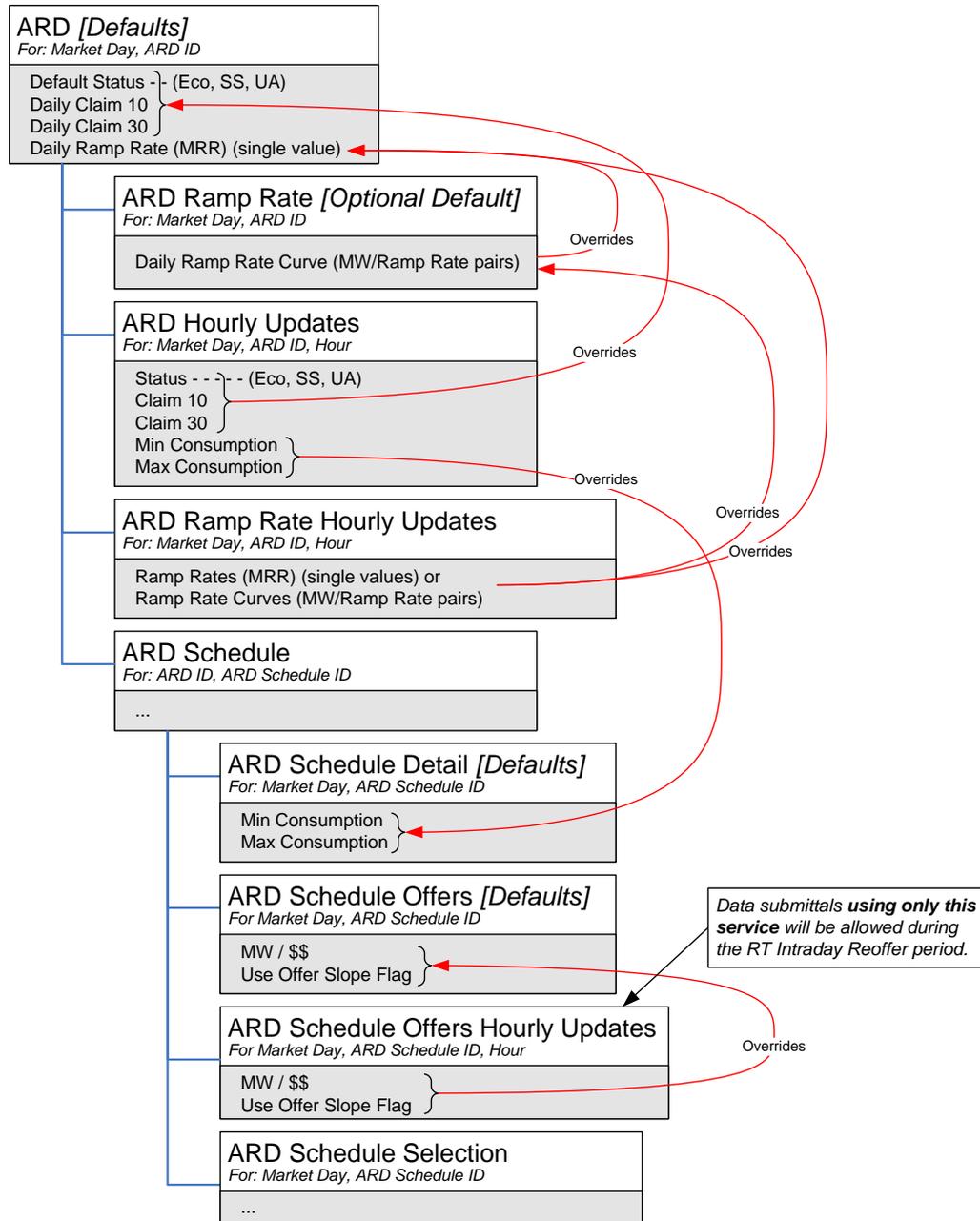


Figure 1 – Illustration of ARD Message Hierarchy and Data Precedence

5.1 ARD

An asset-related demand (ARD) is a physical load, either dispatchable or non-dispatchable, that has been discretely modeled within the ISO New England's Energy Management and Settlement systems and settles at a nodal location. Owners of non-dispatchable ARDs are not expected to submit bid data.

5.1.1 Submit Message

5.1.1.1 Purpose of Message

The purpose of this message is to submit data for a dispatchable asset-related demand that will be used to determine the availability/operating status and capability of the asset on a specific day. This message contains data relevant to the daily operations of an ARD asset, such as the market day, the asset ID, its Claim10 and Claim30 MW amounts, its availability/operating status and its ramp rate.

5.1.1.2 Mandatory and Optional Fields

To create or modify an asset-related demand entry, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ID	Long	The ID of the asset
No	No	day	Date; yyyy-mm-dd	The market day
No	Yes	Claim10	MWType; Decimal 6.1	The MW consumption reduction that can be achieved in 10 minutes
No	Yes	Claim30	MWType; Decimal 6.1	The MW consumption reductions that can be achieved in 30 minutes
No	No	DefaultStatus	Enumerated String; Economic, MustRun, Unavailable	The availability/operating status of the asset
No	No	RampRateDefault	MWType; Decimal 6.1	The default ramp rate to use for the asset if an hourly value does not exist

The nillable claim capability elements above should be given values if the asset is registered as Claim10 or Claim30 capable, i.e. the values submitted should not be nil="true"; otherwise a nil value (nil="true") should be submitted. Non-nil submitted values will be ignored in the market system if the asset does not have the corresponding registered capability.

A limited number of ARD IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARD message in the eMktMessages.xsd.

5.1.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARD party="?">
      <!--1 or more repetitions:-->
      <mes:ARD day=? ID=?>
        <mes:Claim10?</mes:Claim10>
        <mes:Claim30?</mes:Claim30>
        <mes:DefaultStatus?</mes:DefaultStatus>
        <mes:RampRateDefault?</mes:RampRateDefault>
      </mes:ARD>
    </mes:SubmitARD>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARD>
  <ARD day="2010-07-07" ID="123456">
    <Claim10 xsi:nil="true"/>
    <Claim30>60.0</Claim30>
    <DefaultStatus>Economic</DefaultStatus>
    <RampRateDefault>7.1</RampRateDefault>
  </ARD>
</SubmitARD>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.1.1.4 Additional Value Restrictions

- $0 \leq \text{RampRateDefault}$
- A warning will be issued if the RampRateDefault exceeds the RampRateCap provided that the demonstrated capability that appears in the GetARDResponse message is defined (not NULL)
- Nil=true for Claim10 sets the value in the database to NULL
- Nil=true for Claim30 sets the value in the database to NULL
- Claim10 value must be greater than or equal to zero when a non "Nil" is submitted
- Claim30 value must be greater than or equal to zero when a non "Nil" is submitted
- A warning will be returned if the ARD is Claim10 capable and any of the following conditions exist
 - Claim10 = 0 or NULL

- Claim10 > Claim10Cap when the demonstrated capability that appears in the GetARDResponse message is defined (not NULL)
- Claim10 > Claim30
- A warning will be returned if the ARD is Claim30 capable and any of the following conditions exist
 - Claim30 = 0 or NULL
 - Claim30 > Claim30Cap when the demonstrated capability that appears in the GetARDResponse message is defined (not NULL)

5.1.2 Query Message

5.1.2.1 Purpose of Message

The purpose of this message is to get all recorded information about an ARD asset, from the asset's availability/operating status on a specific day, to the identification, location, and ownership. The response will contain data about the ARD asset, some of which was entered with the corresponding Submit message. Data returned is noted in the table below in the section entitled Data Returned.

5.1.2.2 Mandatory and Optional Fields

To get or query the data on an ARD asset, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The specified market day
Yes	No	ID	Long	The ID of the asset

- Data for only one market date at a time can be queried.
- A limited number of asset IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the asset ID element is not specified then the query returns data for all the ARDs that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.1.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARD party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARD>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARD>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetARD>
```

5.1.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The optional elements and attributes below have values if an asset is Claim10 or Claim30 capable.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ARDLongName	String; 40	The full name or description of the asset
No	No	ARDShortName	String; 8	The asset's short name
No	No	AssetType	String; 40	The type of the asset (Pump or Non-pump)
No	No	AssetNumber	Long	The asset ID assigned to the ARD
Yes	Yes	Claim10	MWType; Decimal 6.1	The MW reduction that can be achieved in 10 minutes
Yes	Yes	Claim10Cap	MWType; Decimal 6.1	The Claim10 capability of the asset from the most recent audit.
No	No	Claim10Capable	Boolean; true (1), false (0)	Whether the asset is Claim10 capable
Yes	Yes	Claim30	MWType; Decimal 6.1	The MW reduction that can be achieved in 30 minutes

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	Yes	Claim30Cap	MWType; Decimal 6.1	The Claim30 capability of the asset from the most recent audit.
No	No	Claim30Capable	Boolean; true (1), false (0)	Whether the asset is Claim30 capable
No	No	DayAheadRequired	Boolean; true (1), false (0)	Whether the asset is required to bid in the Day-Ahead Market.
No	No	DefaultStatus	Enumerated String; Economic, MustRun, Unavailable	The availability/operating status of the asset
No	No	Dispatchable	Boolean; true (1), false (0)	Whether the asset is dispatchable
No	No	DispatchZone	String; 40	The dispatch zone in which is located the asset.
No	No	LeadParticipantID	Long	The ID of the Lead Participant
No	No	LeadParticipantName	String; 40	The name of the Lead Participant that supplies bids for the asset to the market
No	No	NodeID	Long	Numeric identifier of the node associated with the asset.
No	No	NodeName	String; 30	Name of the node associated with the asset.
No	Yes	RampRateCap	MWType; Decimal 6.1	The ramp rate capability of the asset from the most recent audit.
No	No	RampRateDefault	MWType; Decimal 6.1	The default ramp rate to use for the asset if an hourly value does not exist

Response Elements

```
<GetARDResponse>
...
<ARD day="2010-05-06" ID="123456">
  <ARDLongName>FICTITIOUS PUMP</ARDLongName>
  <ARDShortName>DARDX</ARDShortName>
  <AssetType>Pump</AssetType>
  <AssetNumber>123456</AssetNumber>
  <Claim10Settings Claim10Capable="true">
    <Claim10>10.0</Claim10>
    <Claim10Cap>15.0</Claim10Cap>
  </Claim10Settings>
  <Claim30Settings Claim30Capable="true">
    <Claim30>16</Claim30>
    <Claim30Cap>20</Claim30Cap>
  </Claim30Settings>
  <DayAheadRequired>false</DayAheadRequired>
  <DefaultStatus>Economic</DefaultStatus>
  <Dispatchable>true</Dispatchable>
  <DispatchZone>MA West</DispatchZone>
  <LeadParticipantID>123456</LeadParticipantID>
  <LeadParticipantName>Some Company Inc.</LeadParticipantName>
  <NodeName>AR. STATIONX13.8DARDS</NodeName>
  <NodeID>54321</NodeID>
  <RampRateCap>35</RampRateCap>
  <RampRateDefault>30</RampRateDefault>
</ARD>
...
</GetARDResponse>
```

5.2 ARD By Participant

See section 5.1 for a definition of an ARD asset.

The ARD By Participant operation allows Market Participants to get information about their ARD bids that cleared in the Day-Ahead Market.

5.2.1 Query Message

5.2.1.1 Purpose of Message

The purpose of this message is to get details about a Market Participants' ARD bids that cleared in the Day-Ahead Market for a specified market day and for an explicitly or implicitly defined list of ARDs. Data returned in this report includes the price per MW bid and the number of MW cleared in the Day-Ahead Market for a particular hour of the specified market day, as well as the asset's name, the name and ID of the schedule.

The data for this message is available after the Re-Offer period for the market day opens.

5.2.1.2 Mandatory and Optional Fields

To get the report of the Market Participants' cleared bids, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the asset

- Data for only one market date at a time can be queried.
- A limited number of asset IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must be listed in one of the user's ARD portfolios. No data will be returned for an ID that is not listed in one of the user's ARD portfolios.
- If the ID element is not specified then the query returns data for all the distinct ARDs that are listed in all of the user's ARD portfolios.

5.2.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDByParticipant party="??">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDByParticipant>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDByParticipant>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetARDByParticipant>
```

5.2.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the asset
No	No	ARDShortName	String; 8	The short name of the ARD
No	No	ARDScheduleID	Long	The ID of the schedule
No	No	ARDScheduleShortName	String; 8	The short name of schedule
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day
No	No	MW	MWType; Decimal 6.1	The MW amount that cleared
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDByParticipantResponse>
      <!--Zero or more repetitions:-->
      <mes:ARDByParticipant day="?" ID="?">
        <mes:ARDSshortName>?</mes:ARDSshortName>
        <mes:ARDScheduleID>?</mes:ARDScheduleID>
        <mes:ARDScheduleShortName>?</mes:ARDScheduleShortName>
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" MW="?" price="?"/>
        </mes:HourlyProfile>
      </mes:ARDByParticipant>
    </mes:GetARDByParticipantResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetARDByParticipantResponse>
  <ARDByParticipant day="2010-05-05" ID="14227">
    <ARDSshortName>DARDX</ARDSshortName>
    <ARDScheduleID>1422712</ARDScheduleID>
    <ARDScheduleShortName>Price S1</ARDScheduleShortName>
    <HourlyProfile>
      <HourlyValues time="2010-05-05T15:00:00-04:00"
        MW="4" price="35.1"/>
    </HourlyProfile>
  </ARDByParticipant>
</GetARDByParticipantResponse>
```

5.3 ARD Hourly

See section 5.1 for a definition of an ARD asset.

5.3.1 Submit Message

5.3.1.1 Purpose of Message

The purpose of this message is to add or update details about an ARD asset's consumption and reserve capabilities when applicable, for a specific hour of a market day. The message can also be used to delete an hourly update by simply setting the "delete" attribute to TRUE on the message element for the hour to delete.

Note that ARD Hourly messages override all schedule and/or default values for the day and hour specified.

5.3.1.2 Mandatory and Optional Fields

To create or modify an hourly override entry for an asset-related demand, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the asset
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
Yes	No	delete	Boolean; true (1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
Yes	No	Claim10	MWType; Decimal 6.1	The claimed MW consumption reduction that can be achieved in 10 minutes. If claim 10 capable, a value must be provided.
Yes	No	Claim30	MWType; Decimal 6.1	The claimed MW consumption reduction that can be achieved in 30 minutes. If claim 30 capable, a value must be provided.
No	No	MaxConsumption	MWType; Decimal 6.1	The maximum consumption of the asset
No	No	MinConsumption	MWType; Decimal 6.1	The minimum consumption of the asset
No	No	ResourceStatus	Enumerated String; Economic, MustRun, Unavailable	The operation status of the asset
Yes	No	ShortName	String; 8	The short name of the asset. Read Only

- A limited number of ARD IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARDHourly message in the eMktMessages.xsd.

5.3.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDHourly party="?">
      <!--1 or more repetitions:-->
      <mes:ARDHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--0 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <mes:Claim10?</mes:Claim10>
            <mes:Claim30?</mes:Claim30>
            <mes:MaxConsumption?</mes:MaxConsumption>
            <mes:MinConsumption?</mes:MinConsumption>
            <mes:ResourceStatus?</mes:ResourceStatus>
          </mes:HourlyValues>
        </mes:HourlyProfile>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:ARDHourly>
    </mes:SubmitARDHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDHourly>
  <ARDHourly day="2010-07-07" ID="123456">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T01:00:00-04:00">
        <Claim10>15</Claim10>
        <Claim30>35</Claim30>
        <MaxConsumption>200</MaxConsumption>
        <MinConsumption>200</MinConsumption>
        <ResourceStatus>Economic</ResourceStatus>
      </HourlyValues>
    </HourlyProfile>
  </ARDHourly>
</SubmitARDHourly>
```

Sample of Deleting Hourly Data

```
<SubmitARDHourly>
  <ARDHourly day="2010-07-07" ID="123456">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T01:00:00-04:00" delete="true"/>
    </HourlyProfile>
  </ARDHourly>
</SubmitARDHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.3.1.4 Additional Value Restrictions

- MaxConsumption \geq MinConsumption ≥ 0 for dispatchable ARDs
MaxConsumption = MinConsumption ≥ 0 for non-dispatchable ARDs.
Note that both must be provided otherwise an error will be reported.
- Claim10 value must be greater than or equal to zero. It must be provided if the asset is Claim10 capable.
- Claim30 value must be greater than or equal to zero. It must be provided if the asset is Claim30 capable.
- A warning will be returned if the ARD is Claim10 capable and any of the following conditions exist
 - Claim10 = 0
 - Claim10 > Claim10Cap when the demonstrated capability that appears in the GetARDResponse message is defined (not NULL)
 - Claim10 > Claim30
- A warning will be returned if the ARD is Claim30 capable and any of the following conditions exist
 - Claim30 = 0
 - Claim30 > Claim30Cap when the demonstrated capability that appears in the GetARDResponse message is defined (not NULL)

Deleting data for an hour is equivalent to removing the override for the market hour.

5.3.2 Query Message

5.3.2.1 Purpose of Message

The purpose of this message is to get the consumption limits and availability/operating status data for ARD assets that override the daily or schedule values for a given hour of a market day. The response will contain data that was entered with a corresponding Submit message or NULL values, meaning that there are no override values. Data returned includes the date and time, the short name and ID of the asset, the availability/operating status, and the min/max consumption limits.

5.3.2.2 Mandatory and Optional Fields

To get or query the hourly overrides on ARDs, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
Yes	No	ID	Long	The ID of the asset

- Data for only one market date at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A limited number of asset IDs can be specified; see the maxOccurs attribute of the QueryByDayIdAndHour type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the asset ID element is not specified then the query returns data for all the ARDs that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.3.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDHourly party="??">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDHourly>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <Hour>2010-07-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetARDHourly>
```

5.3.2.4 Data Returned

The data returned in the Query Response matches the data described for the Submittal with the exception of nullability. Data will be returned for every hour where there is an hourly update/override defines.. The data in the response message will be in ascending hour value.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	Claim10	MWType; Decimal 6.1	The claimed MW consumption reduction that can be achieved in 10 minutes.
No	Yes	Claim30	MWType; Decimal 6.1	The claimed MW consumption reduction that can be achieved in 30 minutes
No	Yes	MaxConsumption	MWType: Decimal 6.1	The maximum consumption of the asset
No	Yes	MinConsumption	MWType: Decimal 6.1	The minimum consumption of the asset
No	Yes	ResourceStatus	Enumerated String: Economic, MustRun, Unavailable	The operation status of the asset
No	No	ShortName	String; 8	The short name of the ARD

Response Elements when there are Overrides to report

```

<GetARDHourlyResponse>
  ...
  <HourlyValues time="2011-07-20T00:00:00-04:00">
    <Claim10>15</Claim10>
    <Claim30>35</Claim30>
    <MaxConsumption>100</MaxConsumption>
    <MinConsumption>10</MinConsumption>
    <ResourceStatus>Economic</ResourceStatus>
  </HourlyValues>
  <ShortName>DARDX</ShortName>
  ...
</GetARDHourlyResponse>

```

Response Elements when there are no Overrides to report⁴

```
<GetARDHourlyResponse>
  ...
  <HourlyValues time="2011-07-20T00:00:00-04:00">
    <Claim10 xsi:nil="true"/>
    <Claim30 xsi:nil="true"/>
    <MaxConsumption xsi:nil="true"/>
    <MinConsumption xsi:nil="true"/>
    <ResourceStatus xsi:nil="true"/>
  </HourlyValues>
  <ShortName>DARDX</ShortName>
  ...
</GetARDHourlyResponse>
```

⁴ The namespace declaration `xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance` has been removed to improve readability.

5.4 ARD Portfolio

See section 4.4 for a definition of a portfolio.

5.4.1 Submit Message

This functionality is provided by the Portfolio's Submit Message described in section 4.4.1

5.4.2 Delete Message

This functionality is provided by the Portfolio's Delete Message described in section 4.4.2

5.4.3 Query Message

This functionality is provided by the Portfolio's Query Message described in section 4.4.3

5.5 ARD Ramp Rate

This operation involves the details of an ARD's consumption ramp rate curve for a given day. A ramp rate defines how quickly an ARD can increase/decrease its consumption in MW per min. See section 5.1 for a definition of an ARD asset.

5.5.1 Submit Message

5.5.1.1 Purpose of Message

The purpose of this message is to submit data relevant to an asset-related demand's consumption ramp rate. This message will submit information that will be used to create the asset's ramp rate curve; specifically the effective market day, the asset's ID, a ramp rate value in MW/min, and the MW consumption range upper bound for the defined ramp rate.

5.5.1.2 Mandatory and Optional Fields

To create or modify an asset's ramp rate curve, you need to provide the following. The optional elements RampRateCap and ShortName are read-only elements. Refer to section 2.4 for information about read-only elements.

Opt	Nil	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the asset
Yes	No	RampRateCap	MWType; Decimal 6.1	The ramp rate capability of the asset from the most recent audit. Read Only
No	No	rate	MWType; Decimal 6.1	The consumption rate of change (MW/min)
No	No	MW	MWType; Decimal 6.1	The upper bound of the MW consumption range to which the "rate" applies. The lower bound is defined by another ramp rate curve element in the same message where the upper bound is less than this value; if one is not defined, the lower bound is zero
Yes	No	ShortName	String; 8	The short name of the asset. Read Only

A limited number of ARD IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARDRampRate message in the eMktMessages.xsd.

5.5.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDRampRate party="?">
      <!--1 or more repetitions:-->
      <mes:ARDRampRate day="? ID=?">
        <!--Optional:-->
        <mes:RampRateCap?</mes:RampRateCap>
        <mes:RampRateCurve>
          <!--0 to 10 repetitions:-->
          <mes:RampRatePoint rate=? MW=?"/>
        </mes:RampRateCurve>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:ARDRampRate>
    </mes:SubmitARDRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDRampRate>
  <ARDRampRate day="2010-07-07" ID="123456">
    <RampRateCurve>
      <RampRatePoint rate="11.0" MW="1.0"/>
      <RampRatePoint rate="8.5" MW="120.0"/>
      <RampRatePoint rate="11.0" MW="200.0"/>
    </RampRateCurve>
  </ARDRampRate>
</SubmitARDRampRate>
```

Sample of Deleting Ramp Rate Curve

```
<SubmitARDRampRate>
  <ARDRampRate day="2010-07-07" ID="123456">
    <RampRateCurve/>
  </ARDRampRate>
</SubmitARDRampRate>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.5.1.4 Additional Value Restrictions

- $0 \leq \text{rate}$; $0 < \text{MW}$
- A warning will be issued if the rate value exceeds the RampRateCap when the demonstrated capability that appears in the GetARDRampRateResponse message is defined (not NULL)

- The RampRatePoint data can be submitted in any order; they don't need to be listed in increasing order of the MW values
- Complete sets of ramp rate curve segments must be submitted; they cannot be submitted separately and independently. A maximum of ten segments can be submitted.

5.5.2 Query Message

5.5.2.1 Purpose of Message

The purpose of this message is to get an ARD's ramp rate curve value(s). Data returned includes the day, the asset name and ID, the asset's ramp rate cap, and the ramp rate and MW range upper bound of each curve segment.

5.5.2.2 Mandatory and Optional Fields

To get or query an asset-related demand ramp rate curve data, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the asset

- Data for only one market date at a time can be queried.
- A limited number of asset IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the asset ID element is not specified then the query returns data for all the ARDs that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.5.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDRampRate party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDRampRate>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetARDRampRate>
```

5.5.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in increasing MW value order. If there is no ramp rate curve defined for an asset, nothing is returned.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the ARD. Read Only
No	No	RampRateCap	MWType; Decimal 6.1	The ramp rate capability of the asset from the most recent audit. Read Only

Response Elements

```
<GetARDRampRateResponse>
  ...
  <ShortName>ABCDEFGH</ShortName>
  <RampRateCap>111.1</RampRateCap>
  ...
</GetARDRampRateResponse>
```

5.6 ARD Ramp Rate Hourly

This operation involves the details of an ARD's consumption ramp rate curve for a given day and set of hours. A ramp rate defines how quickly an ARD can increase/decrease its consumption in MW per min. See section 5.1 for a definition of an ARD asset.

5.6.1 Submit Message

5.6.1.1 Purpose of Message

The purpose of this message is to submit hourly data relevant to an asset-related demand's consumption ramp rate. This message will submit information that will be used to override the asset's daily ramp rate curve; specifically the effective market day and hour, the asset's ID, and either a ramp rate value in MW/min or a ramp rate curve. The message can also be used to delete an hourly update by simply setting the "delete" attribute to TRUE on the message element for the hour to delete.

5.6.1.2 Mandatory and Optional Fields

To create or modify an asset's ramp rate curve, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the asset
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	delete	Boolean; true (1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
Yes	No	FixedRampRate	MWRateType; 5.1	The ramp rate (MW/min) to use if a curve is not provided.
Yes	No	rate	MWRateType; 5.1	The generation rate of change (MW/min) when provided as part of a curve.
Yes	No	MW	MWType; Decimal 6.1	The upper bound of the MW generation range to which the "rate" applies. The lower bound is defined by another ramp rate curve element where the upper bound is less than this value or zero if none is defined.
Yes	No	ShortName	String; 8	The short name of the asset. Read Only

A limited number of asset IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARRampRateHourly message in the eMktMessages.xsd.

5.6.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARRampRateHourly party="?">
      <!--1 or more repetitions:-->
      <mes:ARRampRateHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <!--You have a CHOICE of the next 2 items:-->
            <mes:FixedRampRate?</mes:FixedRampRate>
            <!--1 to 10 repetitions:-->
            <mes:RampRateCurve rate="?" MW="?"/>
          </mes:HourlyValues>
        </mes:HourlyProfile>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </ARRampRateHourly >
    </mes:SubmitARRampRateHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARRampRateHourly party="P1">
  <ARRampRateHourly day="2013-08-07" ID="1234">
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00">
        <FixedRampRate>12</FixedRampRate>
      </HourlyValues>
      <HourlyValues time="2013-08-07T17:00:00-04:00">
        <RampRateCurve rate="11.0" MW="2.0"/>
      </HourlyValues>
    </HourlyProfile>
  </ARRampRateHourly>
</SubmitARRampRateHourly>
```

Sample of Deleting Hourly Data

```
<SubmitARDRampRateHourly party="P1">
  <ARDRampRateHourly day="2013-08-07" ID="1234">
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00"
delete="true"/>
    </HourlyProfile>
  </ARDRampRateHourly>
</SubmitARDRampRateHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.6.1.4 Additional Value Restrictions

- The XSD imposes a Choice for either a fixed ramp rate value or a ramp rate curve.
- $0 \leq \text{rate}$; $0 < \text{MW}$
- A warning will be issued if the rate value exceeds the RampRateCap when the demonstrated capability that appears in the GetARDRampRateResponse message is defined (not NULL)
- The RampRateCurve data can be submitted in any order; they don't need to be listed in increasing order of the MW values
- Complete sets of ramp rate curve segments must be submitted; they cannot be submitted separately and independently. A maximum of ten segments can be submitted.

5.6.2 Query Message

5.6.2.1 Purpose of Message

The purpose of this message is to retrieve a submitted ARD's hourly ramp rate data that override the daily values of a market day. Data returned includes the day, the asset name and ID, the asset's hourly ramp rate data if any exists. The ramp rate data can be either a single ramp rate value or a ramp rate curve.

5.6.2.2 Mandatory and Optional Fields

To get or query ARD hourly ramp rate data, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	ID	Long	The ID of the asset

- Data for only one market date at a time can be queried.

- If the Hour element is not specified then the query returns data for every hour of the market day.
- A list of asset IDs can be specified; see the maxOccurs attribute of the QueryByDayIDAndHour type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the asset ID element is not specified then the query returns data for all the ARDs that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDRampRateHourly party="??">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDRampRateHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDRampRateHourly>
  <QueryFilters>
    <Day>2013-08-07</Day>
    <Hour>2013-08-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetARDRampRateHourly>
```

5.6.2.3 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in increasing MW value order.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the asset. Read Only.

Response Elements

```
<GetARDRampRateHourlyResponse>
  <ARDRampRateHourly>
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00">
        <RampRateCurve rate="11.0" MW="1.0"/>
        <RampRateCurve rate="12.0" MW="2.0"/>
      </HourlyValues>
      <HourlyValues time="2013-08-07T17:00:00-04:00">
        <FixedRampRate>8.0</FixedRampRate>
      </HourlyValues>
    </HourlyProfile>
    <ShortName>ShortName</ShortName>
  </ARDRampRateHourly>
</GetUnitRampRateHourlyResponse>
```

5.7 ARD Schedule

See section 4.5 for a definition of a schedule.

5.7.1 Submit Message

5.7.1.1 Purpose of Message

The purpose of this message is to create a schedule for an asset-related demand that can then be used to submit bids into the Market system. To create a schedule the request must specify the asset ID, a schedule type ID and provide a long and short name for the schedule. The message can also be used to the change/modify the name attributes of a previously created schedule.

5.7.1.2 Mandatory and Optional Fields

To create or modify an asset-related demand schedule, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	No	schedID	Long	The schedule ID
No	No	ARDID	Long	The asset ID
No	No	SchedLongName	String; 40	The long name or description of the schedule
No	No	SchedShortName	String; 8	The short name of the schedule
No	No	SchedTypeID	Long	The schedule type ID

- The attribute schedID is required only when modifying any of the elements of a previously created schedule. It is not required when creating a new schedule because it will be created automatically.
- The schedule type ID must be one of a price-based schedule. The response to the Schedule Type query message (section 4.5) lists all the IDs of price-based schedules.
- Data for a limited number of ARD IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARDSchedule message in the eMktMessages.xsd.

5.7.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDSchedule party="??">
      <!--1 or more repetitions:-->
      <mes:ARDSchedule schedID="??">
        <mes:ARDID?</mes:ARDID>
        <mes:SchedLongName?</mes:SchedLongName>
        <mes:SchedShortName?</mes:SchedShortName>
        <mes:SchedTypeID?</mes:SchedTypeID>
      </mes:ARDSchedule>
    </mes:SubmitARDSchedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDSchedule>
  <ARDSchedule schedID="1234501">
    <ARDID>12345</ARDID>
    <SchedLongName>the long name</SchedLongName>
    <SchedShortName>shrtme</SchedShortName>
    <SchedTypeID>1</SchedTypeID>
  </ARDSchedule>
</SubmitARDSchedule>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.7.1.4 Additional Value Restrictions

- Valid ARD IDs and ARD Schedule identifiers must be submitted

5.7.2 Delete Message

5.7.2.1 Purpose of Message

The purpose of this message is to delete an ARD schedule.

5.7.2.2 Mandatory and Optional Fields

To delete an ARD schedule, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	SchedID	Long	The ID of the schedule to be deleted

5.7.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:DeleteARDSchedule party="?">
      <mes:SchedID?</mes:SchedID>
    </mes:DeleteARDSchedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Delete Submittal

```
<DeleteARDSchedule>
  <SchedID>12345678</SchedID>
</DeleteARDSchedule>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.7.2.4 Additional Value Restrictions

A given schedule cannot be deleted when any of the following conditions exist:

- The schedule is marked as "available" for any day in the past, present, or future within the data retention period of the market database (14 days in the past).
- The ARD is committed on the schedule for any day within the data retention period of the market database.

If any of these conditions exist, the attempt to delete the schedule will result in an error.

5.7.3 Query Message

5.7.3.1 Purpose of Message

The purpose of this message is to get information about all the schedules associated with one or more asset-related demand, specifically the long and short name of the schedules, the schedule type IDs, and the name and ID of the ARD.

5.7.3.2 Mandatory and Optional Fields

To get or query an ARD schedule, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ID	Long	The ARD ID

- A limited number of ARD IDs can be specified; see the maxOccurs attribute of the GetARDSchedule message in the eMktMessages.xsd.

5.7.3.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDSchedule party="">
      <!--1 or more repetitions-->
      <mes:ID?</mes:ID>
    </mes:GetARDSchedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDSchedule>
  <ID>123456</ID>
</GetARDSchedule>
```

5.7.3.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ARDShortName	String; 8	The short name of the ARD
No	No	SchedType	Enumerated String; Cost, Price	The schedule type

Response Elements

```
<GetARDScheduleResponse>
  ...
  <ARDShortName>ShrtNme</ARDShortName>
  <SchedType>Price</SchedType>
  ...
</GetARDScheduleResponse>
```

5.8 ARD Schedule Bid

See section 4.5 for a definition of a schedule.

5.8.1 Submit Message

5.8.1.1 Purpose of Message

The purpose of this message is to submit energy bid data under a schedule for an ARD. This message contains data relevant to the submission of bids for energy, specifically the market day for the bid, the price/MW bid blocks, and the schedule ID.

5.8.1.2 Mandatory and Optional Fields

To create or modify an ARD schedule bid, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 entitled for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)
No	No	MW	MWType; Decimal 6.1	The upper bound of the MW consumption range for the offered price. The lower bound is the upper bound MW value that is defined for a different price in the same message and that is less than this value; if one is not defined, the lower bound is zero.
No	No	UseOfferSlope	Boolean; true (1), false (0)	Whether to interpolate the price between the MW upper bound in an ARD bid curve (Price/MW bid blocks)
Yes	No	ShortName	String; 8	The short name of the schedule. Read Only

Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARDScheduleBid message in the eMktMessages.xsd.

5.8.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDScheduleBid party="?">
      <!--1 or more repetitions:-->
      <mes:ARDScheduleBid day="? ID=?">
        <mes:PriceCurve>
          <!--1 to 10 repetitions:-->
          <mes:PricePoint price=? MW=?"/>
        </mes:PriceCurve>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
        <mes:UseOfferSlope?</mes:UseOfferSlope>
      </mes:ARDScheduleBid>
    </mes:SubmitARDScheduleBid>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDScheduleBid>
  <ARDScheduleBid day="2010-07-07" ID="123456">
    <PriceCurve>
      <PricePoint price="50.00" MW="50.0"/>
      <PricePoint price="40.00" MW="100.0"/>
    </PriceCurve>
    <UseOfferSlope>true</UseOfferSlope>
  </ARDScheduleBid>
</SubmitARDScheduleBid>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.8.1.4 Additional Value Restrictions

- MW values must be greater than or equal to zero; i.e. $MW \geq 0$
- Price must be less than or equal to the Energy Offer Cap price and greater than or equal to the Energy Offer Floor price; i.e. $\text{Energy Offer Floor price} \leq \text{Price} \leq \text{Energy Offer Cap price}$ ⁵.
- Complete sets of Price/MW bid pairs must be submitted; they cannot be submitted separately and independently. A maximum of ten Price/MW bid pairs per schedule can be submitted. At least 1 Price/MW pair must be submitted.
- Bids must be monotonically decreasing (price and MW); i.e. the price for a given MW upper bound must be less than or equal to the price for

⁵ Energy Offer Floor price and Energy Offer Cap price are both defined in Market Rule 1, section I.2.2.

a lower MW upper bound. Note that Price/MW bid pairs don't need to be submitted in any particular order. However the MW values of each pair must be different from the others such that the MW value for a pair must be greater than the MW of the previous pair after the Price/MW pairs are ordered in increasing MW value. The monotonically decreasing rule is enforced after the complete set of pairs are received, ordered and processed.

- Cost-based schedules cannot be modified

5.8.2 Query Message

5.8.2.1 Purpose of Message

The purpose of this message is to get ARD schedule bid data that are applicable to a specified market day. The response will contain the schedule bid data for a specified market day and a list of ARD schedules. The data returned includes the market day, the price/MW bid blocks, and the ID and short name of the specified schedules.

5.8.2.2 Mandatory and Optional Fields

To get or query an ARD schedule bids, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the schedule

- Data for only one market date at a time can be queried.
- A limited number of schedule IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all be associated with assets that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If a schedule ID is not specified then the query returns data for every schedule of every asset that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.8.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDScheduleBid party=""?>
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDScheduleBid>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDScheduleBid>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetARDScheduleBid>
```

5.8.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in increasing MW value order.

Opt	Nil	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the schedule.

Response Elements

```
<GetARDScheduleBidResponse>
  ...
  <ShortName>SchdNme</ShortName>
  ...
</GetARDScheduleBidResponse>
```

5.9 ARD Schedule Bid Hourly

See section 4.5 for a definition of a schedule.

5.9.1 Submit Message

5.9.1.1 Purpose of Message

The purpose of this message is to submit hourly energy bid overrides under a schedule for an ARD. This message contains data relevant to the submission of bids for energy, specifically the market day and hour for the bid, the price/MW bid blocks, and the schedule ID. The message can also be used to delete an hourly update by simply setting the “delete” attribute to TRUE on the message element for the hour to delete.

5.9.1.2 Mandatory and Optional Fields

To create or modify an hourly ARD schedule bid, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 entitled for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	delete	Boolean; true (1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
No	No	ID	Long	The ID of the schedule
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)
No	No	MW	MWType; Decimal 6.1	The upper bound of the MW consumption range for the offered price. The lower bound is the upper bound MW value that is defined for a different price in the same message and that is less than this value; if one is not defined, the lower bound is zero.
No	No	UseOfferSlope	Boolean; true (1), false (0)	Whether to interpolate the price between the MW upper bound in an ARD bid curve (Price/MW bid blocks)
Yes	No	ShortName	String; 8	The short name of the ARD. Read Only

Full SOAP Message

```
<soapenv:Envelope>
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDScheduleBidHourly party="">
      <!--1 or more repetitions:-->
      <mes:ARDScheduleBidHourly day="" ID="">
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="" delete="">
            <mes:PriceCurve>
              <!--1 to 10 repetitions:-->
              <mes:PricePoint price="" MW=""/>
            </mes:PriceCurve>
            <mes:UseOfferSlope?></mes:UseOfferSlope>
          </mes:HourlyValues>
        </mes:HourlyProfile>
        <!--Optional:-->
        <mes:ShortName?></mes:ShortName>
      </mes:ARDScheduleBidHourly>
    </mes:SubmitARDScheduleBidHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDScheduleBidHourly party="P1">
  <ARDScheduleBidHourly day="2013-08-07" id="109">
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00">
        <PriceCurve>
          <PricePoint price="10" MW="101"/>
        </PriceCurve>
        <UseOfferSlope>true</UseOfferSlope>
      </HourlyValues>
      <HourlyValues time="2013-08-07T17:00:00-04:00">
        <PriceCurve>
          <PricePoint price="10" MW="101"/>
        </PriceCurve>
        <UseOfferSlope>true</UseOfferSlope>
      </HourlyValues>
    </HourlyProfile>
  </ARDScheduleBidHourly>
</SubmitARDScheduleBidHourly>
```

Sample of Deleting Hourly Data

```
<SubmitARDScheduleBidHourly party="P1">
  <ARDScheduleBidHourly day="2013-08-07" id="109">
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00" delete="true"/>
    </HourlyProfile>
  </ARDScheduleBidHourly>
</SubmitARDScheduleBidHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.9.1.3 Additional Value Restrictions

- MW values must be greater than or equal to zero; i.e. $MW \geq 0$
- Price must be less than or equal to the Energy Offer Cap price and greater than or equal to the Energy Offer Floor price; i.e. Energy Offer Floor price \leq Price \leq Energy Offer Cap price⁶.
- Complete sets of Price/MW bid pairs must be submitted; they cannot be submitted separately and independently. A maximum of ten Price/MW bid pairs per schedule can be submitted. At least 1 Price/MW pair must be submitted.
- Bids must be monotonically decreasing (price and MW); i.e. the price for a given MW upper bound must be less than or equal to the price for a lower MW upper bound. Note that Price/MW bid pairs don't need to be submitted in any particular order. However the MW values of each pair must be different from the others such that the MW value for a pair must be greater than the MW of the previous pair after the Price/MW pairs are ordered in increasing MW value. The monotonically decreasing rule is enforced after the complete set of pairs are received, ordered and processed.
- Cost-based schedules cannot be modified

5.9.2 Query Message

5.9.2.1 Purpose of Message

The purpose of this message is to get ARD schedule bid data that are applicable to a specified market day and hour. The response will contain the hourly schedule bid data for a specified market day and a list of ARD schedules. The data returned includes the market day, the price/MW bid blocks, and the ID and short name of the specified schedules.

⁶ Energy Offer Floor price and Energy Offer Cap price are both defined in Market Rule 1, section I.2.2.

5.9.2.2 Mandatory and Optional Fields

To get or query an asset-related demand hourly, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	ID	Long	The id of the schedule

- Data for only one market date at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A limited number of schedule IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all be associated with assets that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the schedule ID is not specified then the query returns data for every schedule of every asset that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.9.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDScheduleBidHourly party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDScheduleBidHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDScheduleBidHourly>
  <QueryFilters>
    <Day>2013-08-07</Day>
    <Hour>2013-08-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetARDScheduleBidHourly>
```

5.9.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	UseOfferSlope	Boolean; true (1), false (0)	Whether to interpolate the price between the MW upper bound in an ARD bid curve (Price/MW bid blocks)
No	No	ShortName	String; 8	The short name of the asset.

Response Elements

```
<GetARDScheduleBidHourlyResponse>
  <ARDScheduleBidHourly day="2010-07-07" ID="123456">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T01:00:00-04:00">
        <PriceCurve>
          <PricePoint price="10" MW="30">
            <PricePoint price="11" MW="38">
          </PriceCurve>
          <UseOfferSlope>true</UseOfferSlope>
        </HourlyValues>
      </HourlyProfile>
      <ShortName>ShortName</ShortName>
    </ARDScheduleBidHourly>
  </GetARDScheduleBidHourlyResponse>
```

5.10 ARD Schedule Detail

See section 4.5 for a definition of a schedule.

5.10.1 Submit Message

5.10.1.1 Purpose of Message

The purpose of this message is to submit all non-energy bid data under a schedule for an ARD. This message contains schedule data relevant to the energy dispatch of an ARD, such as the market day, the schedule ID, the max/min consumption limits, and the market to which the bid data applies.

5.10.1.2 Mandatory and Optional Fields

To create or modify an ARD schedule details, you need to provide the following. The optional elements are read-only elements. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	MarketType	Enumerated String; Both, DayAhead, RealTime	The energy markets in which this schedule's bid parameters are to be used.
No	No	MaxConsumption	MWType; Decimal 6.1	The economic maximum MW consumption dispatch limit
No	No	MinConsumption	MWType; Decimal 6.1	The economic minimum MW consumption dispatch limit
Yes	No	ScheduleAvailable	Boolean; true (1), false (0)	Whether the schedule is available for use ⁷ . Read Only.
Yes	No	ShortName	String; 8	The short name of the schedule. Read Only

Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARDScheduleDetail message in the eMktMessages.xsd.

⁷ As defined using the Submit ARD Schedule Selection message defined in §5.11.1

5.10.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDScheduleDetail party="?">
      <!--1 or more repetitions:-->
      <mes:ARDScheduleDetail day="? ID=?">
        <mes:MarketType?</mes:MarketType>
        <mes:MaxConsumption?</mes:MaxConsumption>
        <mes:MinConsumption?</mes:MinConsumption>
        <!--Optional:-->
        <mes:ScheduleAvailable?</mes:ScheduleAvailable>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:ARDScheduleDetail>
    </mes:SubmitARDScheduleDetail>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDScheduleDetail>
  <ARDScheduleDetail day="2010-07-07" ID="1234">
    <MarketType>DayAhead</MarketType>
    <MaxConsumption>13.0</MaxConsumption>
    <MinConsumption>7.0</MinConsumption>
  </ARDScheduleDetail>
</SubmitARDScheduleDetail>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.10.1.4 Additional Value Restrictions

- MaxConsumption ≥ MinConsumption ≥ 0 for dispatchable ARDs
MaxConsumption = MinConsumption ≥ 0 for non-dispatchable ARDs.
- Cost-based schedules cannot be modified

5.10.2 Query Message

5.10.2.1 Purpose of Message

The purpose of this message is to get all non-energy data for an ARD schedule for a specified day. The response will contain the data that was submitted with the corresponding Submit ARD Schedule Detail message. Data returned includes the market day, the economic consumption dispatch limits, the markets to which the bids apply, as well as the asset ID, and short name.

5.10.2.2 Mandatory and Optional Fields

To get or query an ARD schedule detail, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the schedule

- Data for only one market date at a time can be queried.
- A limited number of schedule IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd They must all be associated with ARDs that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If a schedule ID is not specified then the query returns data for every schedule of every ARD that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.10.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDScheduleDetail party=""?>
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetARDScheduleDetail>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDScheduleDetail>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetARDScheduleDetail>
```

5.10.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt	Nil	Element or Attribute	Data Type; Format	Comments
No	No	ScheduleAvailable	Boolean; true (1), false (0)	Whether the schedule is available for use in the specified markets. Read Only.
No	No	ShortName	String; 8	The short name of the schedule. Read Only

Response Elements

```
<GetARDScheduleDetailResponse>  
  ...  
  <ScheduleAvailable>true</ScheduleAvailable>  
  <ShortName>ShrtNme</ShortName>  
  ...  
</GetARDScheduleDetailResponse>
```

5.11 ARD Schedule Selection

See section 4.5 for a definition of a schedule.

5.11.1 Submit Message

5.11.1.1 Purpose of Message

The purpose of this message is to set the availability of a schedule for a specific market day. This message contains data relevant to setting the availability of a schedule, such as the market day, the schedule ID and its availability.

5.11.1.2 Mandatory and Optional Fields

To set or change the availability of a schedule, you need to provide the following. The optional element is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	Available	Boolean; true (1), false (0)	Whether the schedule is available
Yes	No	ShortName	String; 8	The short name of the schedule. Read Only

Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitARDScheduleSelection message in the eMktMessages.xsd.

5.11.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitARDScheduleSelection party="?">
      <!--1 or more repetitions:-->
      <mes:ARDScheduleSelection day="?">
        <!--Zero or more repetitions:-->
        <mes:ARDSchedule ID="?">
          <mes:Available?</mes:Available>
          <!--Optional:-->
          <mes:ShortName?</mes:ShortName>
        </mes:ARDSchedule>
      </mes:ARDScheduleSelection>
    </mes:SubmitARDScheduleSelection>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitARDScheduleSelection>
  <ARDScheduleSelection day="2010-07-07">
    <ARDSchedule ID="1234567890">
      <Available>true</Available>
    </ARDSchedule>
  </ARDScheduleSelection>
</SubmitARDScheduleSelection>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

5.11.1.4 Additional Value Restrictions

- Only one price-based schedule per market (Day-Ahead, Real Time) can be marked as available for a given market day. Therefore, when changing the availability of schedules, when one is set available for a particular market then the one that was previously available for that market must be set unavailable within the same message, and vice versa. Note that if you change the availability of a scheduled that is marked as applicable to both DA and RT markets, you may have to change the availability of two other schedules in the same message, one applicable to the DA market only and another applicable to the RT market only.
- The availability of cost-based schedules cannot be changed using this message. A request to do so will be rejected and an error returned.

5.11.2 Query Message

5.11.2.1 Purpose of Message

The purpose of this message is to get the availability of all the ARD schedules for a given day. The data returned includes the market day, the ID of the schedule, its short name and its availability.

5.11.2.2 Mandatory and Optional Fields

To get or query the ARD schedule availabilities, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

- Only one market date can be queried at a time.
- The query returns data for every schedule of every ARD that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

5.11.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetARDScheduleSelection party="?">
      <mes:Day?</mes:Day>
    </mes:GetARDScheduleSelection>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetARDScheduleSelection>
  <Day>2010-07-07</Day>
</GetARDScheduleSelection>
```

5.11.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt	Nil	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the schedule. Read Only

Response Elements

```
<GetARDScheduleSelectionResponse>
  ...
  <ShortName>ShrtNme</ShortName>
  ...
</GetARDScheduleSelectionResponse>
```

6. Demand Web Services

This section defines the submittals and queries used for Demand and Load Response operations.

The table below shows a summary of the Demand and Load Response messages and indicates if data can be submitted, queried or deleted. There is no hierarchy among the Demand and Load Response messages. Demand Bid data must be submitted for each market day; their effective period is only one day. Load Response Offers on the other hand are for user defined periods; i.e., the Market Participant defines the start market day and the termination market day and the period can span more than one month.

Message	Brief Description of Usage	Submit	Query	Delete
Demand Bids	Set (edit/create) or view a Demand Bid	X	X	See note 1
Demand Portfolio	Manage Pnode Portfolios	X	X	X
Demand By Participant	View the demand bids that cleared in the Day-Ahead market		X	
Detailed Demand By Participant	View the demand bids that cleared in the Day-Ahead market		X	
Load Response Offer	Set (edit/create) or view a Load Response Offer	X	X	X
Load Response Result	View the cleared Day-Ahead Market Load Response Results		X	

Note 1: See section 2.3 for information on the delete operation for this case.

6.1 Demand Bids

Demand bids are daily bids to purchase energy to satisfy load demand for a given date and time. There are four types of demand bids:

- Fixed demand bids – These specify a single MW amount irrespective of market clearing price.
- Price-Sensitive demand bids – These specify a MW amount for a price. These will clear if the clearing price is greater than or equal to the quoted price. Up to 10 price-sensitive demand bids at a Pnode per hour can be submitted.
- Increment bids (offers) – These specify a MW supply offer at a node for a price. Up to 50 increment bids at a Pnode per hour can be submitted.
- Decrement bids – These specify a MW consumption bid for a price at a node. Up to 50 decrement bids at a Pnode per hour can be submitted.

Participants with registered subaccounts in CAMS can have their users submit demand bids in distinct named subaccounts. See 1.4 Demand Bid Subaccounts.

6.1.1 Submit Message

6.1.1.1 Purpose of Message

The purpose of this message is to submit demand bids to the market. This message contains data relevant to placing a bid, such as the market day, the market hour, the Pnode ID, the bid type, the MW amount and when applicable the price per MWH. If the Participant has registered subaccounts in CAMS, the user can specify the subaccount for which the bids are being submitted; if the subaccount is not specified, the demand bids are submitted for the “Default” subaccount. The message can also be used to delete hourly data by simply setting the “delete” attribute to TRUE on the message element for the hour to delete.

6.1.1.2 Mandatory and Optional Fields

To create or modify a bid, you need to provide the following. The optional element NodeName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	No	SubAccount	String; 20	The name of the subaccount. The name of the default subaccount “Default” can be used.
No	No	bidType	Enumerated Sting: Fixed, PriceSensitive, Decrement, Increment	The type of bid
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the Pnode
Yes	No	nodeName	String; 40	The name of the Pnode. Read Only
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day
Yes	No	delete	Boolean; true(1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
Yes	No	FixedMW	MWType; Decimal 6.1 See Note 1	The MW amount for a Fixed demand bid
Yes	No	price	PriceType; Decimal 6.2 See Note 1	The price/MWH for PriceSensitive, Increment and Decrement bids
Yes	No	MW	MWType; Decimal 6.1 See Note 1	The MW amount for PriceSensitive, Increment and Decrement bids

Data for a limited number of Pnode IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitDemandBid message in the eMktMessages.xsd.

Note 1: If you select “Fixed” for “bidType”, you must submit a FixedMW amount; else if you select “PriceSensitive”, “Decrement” or “Increment”, you must submit one or more price/MW pairs of values.

Data for only one subaccount can be submitted per request. As explained in 1.4, Participants may register subaccounts for the demand bids submitted by their users. Participants with registered subaccounts should specify the subaccount when submitting demand bids. When the subaccount is not specified, eMarket enters the demand bids for the Participant’s default subaccount.

6.1.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitDemandBid party="?">
      <!--Optional:-->
      <mes:SubAccount?</mes:SubAccount>
      <!--1 or more repetitions:-->
      <mes:DemandBid bidType=?" day=?" ID=?">
        <!--Optional:-->
        <mes:nodeName?</mes:nodeName>
        <mes:HourlyProfile>
          <!--0 to 25 repetitions:-->
          <mes:HourlyBid time=?" delete=?">
            <!--If you select "bidType"=Fixed you must use-->
            <mes:FixedMW?</mes:FixedMW>
            <!--If you select "bidType"=PriceSensitive, or-->
            <!--Increment, or Decrement you must use-->
            <!--1 or more repetitions:-->
            <mes:PricePoint price=?" MW=?"/>
          </mes:HourlyBid>
        </mes:HourlyProfile>
      </mes:DemandBid>
    </mes:SubmitDemandBid>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitDemandBid>
  <SubAccount>Subaccount Name</SubAccount>
  <DemandBid bidType="PriceSensitive" day="2010-07-07"
    ID="12345">
    <HourlyProfile>
      <HourlyBid time="2010-07-07T16:00:00-04:00">
        <PricePoint price="35.00" MW="76.2"/>
      </HourlyBid>
    </HourlyProfile>
  </DemandBid>
</SubmitDemandBid>
```

Sample of Submittal Containing Multiple DemandBid Elements to submit bids of different types at the same or different Pnodes

```
<SubmitDemandBid>
  <SubAccount>Subaccount Name</SubAccount>
  <!--First some Fixed demand bids at a Pnode-->
  <DemandBid bidType="Fixed" day="2012-01-22" ID="4007">
    <!--Optional-->
    <NodeName>.Z.WCMASS</NodeName>
    <HourlyProfile>
      <!--0 to 25 repetitions-->
      <HourlyBid time="2012-01-22T00:00:00-05:00">
        <mes:FixedMW>50</mes:FixedMW>
      </HourlyBid>
      <HourlyBid time="2012-01-22T01:00:00-05:00">
        <mes:FixedMW>51</mes:FixedMW>
      </HourlyBid>
      <HourlyBid time="2012-01-22T02:00:00-05:00">
        <mes:FixedMW>52</mes:FixedMW>
      </HourlyBid>
    </HourlyProfile>
  </DemandBid>
  <!--Now some Price Sensitive demand bids at the same Pnode-->
  <DemandBid bidType="PriceSensitive" day="2012-01-22" ID="4007">
    <!--Optional-->
    <NodeName>.Z.WCMASS</NodeName>
    <HourlyProfile>
      <!--0 to 25 repetitions-->
      <HourlyBid time="2012-01-22T00:00:00-05:00">
        <!--1 or more repetitions; max 10-->
        <PricePoint price="11.00" MW="101.0"/>
        <PricePoint price="11.10" MW="101.1"/>
        <PricePoint price="11.20" MW="101.2"/>
      </HourlyBid>
      <HourlyBid time="2012-01-22T01:00:00-05:00">
        <!--1 or more repetitions; max 10-->
        <PricePoint price="21.00" MW="111.0"/>
        <PricePoint price="21.10" MW="111.1"/>
        <PricePoint price="21.20" MW="111.2"/>
      </HourlyBid>
      <HourlyBid time="2012-01-22T02:00:00-05:00">
        <!--1 or more repetitions; max 10-->
        <PricePoint price="31.00" MW="121.0"/>
      </HourlyBid>
    </HourlyProfile>
  </DemandBid>
```

```

        <PricePoint price="31.10" MW="121.1"/>
        <PricePoint price="31.20" MW="121.2"/>
    </HourlyBid>
</HourlyProfile>
</DemandBid>
<!--Next some Fixed demand bids at a different Pnode-->
<DemandBid bidType="Fixed" day="2012-01-22" ID="4004">
    <!--Optional-->
    <NodeName>.Z.CONNECTICUT</NodeName>
    <HourlyProfile>
        <!--0 to 25 repetitions-->
        <HourlyBid time="2012-01-22T03:00:00-05:00">
            <mes:FixedMW>53</mes:FixedMW>
        </HourlyBid>
        <HourlyBid time="2012-01-22T04:00:00-05:00">
            <mes:FixedMW>54</mes:FixedMW>
        </HourlyBid>
        <HourlyBid time="2012-01-22T05:00:00-05:00">
            <mes:FixedMW>55</mes:FixedMW>
        </HourlyBid>
    </HourlyProfile>
</DemandBid>
<!--Now some Price Sensitive demand bids at the same Pnode-->
<DemandBid bidType="PriceSensitive" day="2012-01-22" ID="4004">
    <!--Optional -->
    <NodeName>.Z.CONNECTICUT</NodeName>
    <HourlyProfile>
        <!--0 to 25 repetitions-->
        <HourlyBid time="2012-01-22T03:00:00-05:00">
            <!--1 or more repetitions; max 10-->
            <PricePoint price="11.30" MW="101.3"/>
            <PricePoint price="11.40" MW="101.4"/>
            <PricePoint price="11.50" MW="101.5"/>
        </HourlyBid>
        <HourlyBid time="2012-01-22T04:00:00-05:00">
            <!--1 or more repetitions; max 10-->
            <PricePoint price="21.30" MW="111.3"/>
            <PricePoint price="21.40" MW="111.4"/>
            <PricePoint price="21.50" MW="111.5"/>
        </HourlyBid>
        <HourlyBid time="2012-01-22T05:00:00-05:00">
            <!--1 or more repetitions; max 10-->
            <PricePoint price="31.30" MW="121.3"/>
            <PricePoint price="31.40" MW="121.4"/>
            <PricePoint price="31.50" MW="121.5"/>
        </HourlyBid>
    </HourlyProfile>
</DemandBid>
<!--Now some Inc demand bids at a Pnode-->
<DemandBid bidType="Increment" day="2012-01-22" ID="519">
    <!--Optional -->
    <NodeName>UN.NORWALKH17.1NRW1</NodeName>
    <HourlyProfile>
        <!--0 to 25 repetitions-->
        <HourlyBid time="2012-01-22T00:00:00-05:00">
            <!--1 or more repetitions; max 50-->
            <PricePoint price="36.28" MW="2"/>

```

```

        <PricePoint price="39.28" MW="3"/>
        <PricePoint price="40.28" MW="4"/>
    </HourlyBid>
    <HourlyBid time="2012-01-22T01:00:00-05:00">
        <!--1 or more repetitions; max 50-->
        <PricePoint price="36.28" MW="2"/>
        <PricePoint price="39.28" MW="3"/>
        <PricePoint price="40.28" MW="4"/>
    </HourlyBid>
    <HourlyBid time="2012-01-22T02:00:00-05:00">
        <!--1 or more repetitions; max 50-->
        <PricePoint price="36.28" MW="2"/>
        <PricePoint price="39.28" MW="3"/>
        <PricePoint price="40.28" MW="4"/>
    </HourlyBid>
</HourlyProfile>
</DemandBid>
<!--Now some Dec demand bids at a different Pnode-->
<DemandBid bidType="Decrement" day="2012-01-22" ID="4261">
    <!--Optional-->
    <NodeName> LD.TEWKSBR13.8</NodeName>
    <HourlyProfile>
        <!--0 to 25 repetitions-->
        <HourlyBid time="2012-01-22T00:00:00-05:00">
            <!--1 or more repetitions; max 50-->
            <PricePoint price="36.28" MW="2"/>
            <PricePoint price="39.28" MW="3"/>
            <PricePoint price="40.28" MW="4"/>
        </HourlyBid>
        <HourlyBid time="2012-01-22T01:00:00-05:00">
            <!--1 or more repetitions; max 50-->
            <PricePoint price="36.28" MW="2"/>
            <PricePoint price="39.28" MW="3"/>
            <PricePoint price="40.28" MW="4"/>
        </HourlyBid>
        <HourlyBid time="2012-01-22T02:00:00-05:00">
            <!--1 or more repetitions; max 50-->
            <PricePoint price="36.28" MW="2"/>
            <PricePoint price="39.28" MW="3"/>
            <PricePoint price="40.28" MW="4"/>
        </HourlyBid>
    </HourlyProfile>
</DemandBid>
</SubmitDemandBid>

```

Sample of Deleting Hourly Data

```

<SubmitDemandBid>
    <SubAccount>Subaccount Name</SubAccount>
    <DemandBid bidType="PriceSensitive" day="2012-01-22"
        ID="12345">
        <HourlyProfile>
            <HourlyBid time="2012-01-22T02:00:00-05:00" delete="true"/>
            <HourlyBid time="2012-01-22T03:00:00-05:00" delete="true"/>
        </HourlyProfile>
    </DemandBid>
</SubmitDemandBid>

```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

6.1.1.4 Additional Value Restrictions

- MW values must be greater than zero; $MW > 0$
- The Price of a bid must be less than or equal to the Energy Offer Cap price and greater than or equal to the Energy Floor price; i.e. $Energy\ Offer\ Floor\ price \leq Price \leq Energy\ Offer\ Cap\ price$ ⁸.
- Fixed and PriceSensitive demand bids can only be submitted at Load Zone pricing nodes.
- Increment, Decrement, and PriceSensitive demand bids can consist of multiple Price/MW pairs. Each pair is an independent offer or bid with the total for an hour of the offer or bid being the sum of the individual MW values. Since each is treated independently, entering multiple duplicate bids for a given hour is equivalent to entering a single bid where the MW value is the sum of the MW values of the duplicate bids; for example, three bids for hour-ending 02 of 2 MW @ \$10.00 is equivalent to one bid of 6 MW @ \$10.00. The sum of the MW(s) offered or bid at any one location are subject to maximum values stipulated in the Market Operations Manual, M-11.
- Demand bids submitted for an invalid subaccount name or a terminated subaccount will be rejected. *However, SubmitDemandBid requests to delete demand bids for a terminated subaccount will not be rejected.*

6.1.2 Query Message

6.1.2.1 Purpose of Message

The purpose of this message is to get submitted demand bids. The response will contain data about the demand bids, most of which was entered with a corresponding Submit message. Data returned includes the market day, the hour of the day, the bid type, the name and ID of the Pnode, the MW amount of Fixed demand bids, and the Price/MW pairs for the other type of demand bids. For Market Participants with registered subaccounts, they can specify whether or not the demand bids are to be returned with subaccount information, and if so, they can explicitly specify the subaccount for which the demand bids are to be returned.

⁸ Energy Offer Floor price and Energy Offer Cap price are both defined in Market Rule 1, section I.2.2.

6.1.2.2 Mandatory and Optional Fields

To get or query demand bids, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	BidType	Enumerated String ; All, Fixed, PriceSensitive, Decrement, Increment	The type of bid
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of a Pnode
Yes	No	subAccounts	Boolean; true (1), false (0)	A flag used to specify whether the demand bid data is to be returned grouped by subaccount. When this attribute is missing or set to False (0), the demand bids for only the Participant's "Default" subaccount is returned without any subaccount attributes in the response message.
Yes	No	SubAccount	String; 20	The subaccount name can only be specified if the "subAccounts" attribute is set to True (1).

- Only one market date with one BidType and a list of Pnode IDs can be specified per query
- A limited number of node IDs can be specified; see the maxOccurs attribute of the QueryByDayIdBidTypeAndSubAccount type in the eMktMessages.xsd.
- If the ID element is not specified then the query returns data for all the demand bids submitted by the Participant.
- A limited number of subaccount names can be specified when the "subAccounts" attribute is set to True (1); see the maxOccurs attribute of the QueryByDayIdBidTypeAndSubAccount type in the eMktMessages.xsd. *If a specified subaccount name does not match any of the subaccounts registered by the Participant, the request is rejected.*
- If the SubAccount element is not specified and the "subAccounts" is attribute set to True (1), then the query returns data for all subaccounts belonging to the Participant for which demand bids exist, including the "Default" subaccount. **Thus Participants without registered subaccounts in CAMS that submit queries with the "subAccounts" attribute set to True (1) will see a "subaccount" tag in the XML response message with "Default" as the subaccount name.**
- Query messages with a SubAccount element but no subAccounts attribute or the subAccounts attribute set to False (0) will be rejected.

6.1.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDemandBid party="?" subAccounts="?">
      <mes:QueryFilters>
        <mes:BidType?</mes:BidType>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
        <!--Zero or more repetitions:-->
        <mes:SubAccount?</mes:SubAccount>
      </mes:QueryFilters>
    </mes:GetDemandBid>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetDemandBid>
  <QueryFilters subAccounts="true">
    <BidType>All</BidType>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
    <SubAccount>Subaccount Name</SubAccount>
  </QueryFilters>
</GetDemandBid>
```

6.1.2.4 Data Returned

Depending on whether the “subAccounts” attribute is present or set in the query message, the response message format will vary. When the attribute is omitted or set to False (0), the response will be a number of DemandBid elements within the GetDemandBidResponse element. When the attribute is set to True (1), the response will be a number of DemandBidBySubAccount elements within the GetDemandBidResponse element.

Format of response containing DemandBid elements, i.e. response to a query with no “subAccounts” attribute or with attribute set to False (0):

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. For price-sensitive, increment and decrement demand bids, the data in the response message will be in increasing hour order with the MW/Price pairs within the hour in the order submitted.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	nodeName	String; 40	The node name. Read Only

Response Elements

```
<GetDemandBidResponse>
  ...
  <NodeName>NodeName</NodeName>
  ...
</GetDemandBidResponse>
```

Format of response containing DemandBidBySubAccount elements, i.e. response to a query with “subAccounts” attribute set to True (1):

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. For price-sensitive, increment and decrement demand bids, the data in the response message will be in increasing hour order with the MW/Price pairs within the hour in the order submitted.

For Participants with no registered subaccounts in CAMS who elect to use the GetDemandBid query with the “subAccounts” set to True (1), the subaccount name that will appear in the response will be “Default”.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	subAccount	String; 20	The subaccount name.
No	No	NodeName	String; 40	The node name. Read Only

Response Elements

```
<GetDemandBidResponse>
  <DemandBidBySubAccount subAccount="SubAccount Name">
    ...
    <NodeName>NodeName</NodeName>
    ...
  </DemandBidBySubAccount>
</GetDemandBidResponse>
```

6.2 Demand Portfolio

See section 4.4 for a definition of a portfolio.

6.2.1 Submit Message

This functionality is provided by the Portfolio's Submit Message described in section 4.4.1

6.2.2 Delete Message

This functionality is provided by the Portfolio's Delete Message described in section 4.4.2

6.2.3 Query Message

This functionality is provided by the Portfolio's Query Message described in section 4.4.3

6.3 Demand By Participant

The Demand By Participant operation allows Market Participants to get information about their demand bids that cleared in the Day-Ahead market. See section 6.1 for a description of demand bids.

6.3.1 Query Message

6.3.1.1 Purpose of Message

The purpose of this message is to get a report that lists the Participant's demand bids that cleared in the Day-Ahead market for a specified day and a list of Pnodes. The response data includes the market day, the hour of the day, the Pnode ID and name, the net MW amount cleared and the market clearing price. For Market Participants with registered subaccounts, they can specify whether or not the demand bids are to be returned with subaccount information, and if so, they can explicitly specify the subaccount for which the demand bids are to be returned.

The data for this message is available after the Re-Offer period for the market day opens.

6.3.1.2 Mandatory and Optional Fields

To get the report of the Market Participants' cleared bids, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of a Pnode
Yes	No	subAccounts	Boolean; true (1), false (0)	A flag used to specify whether the demand bid data is to be returned grouped by subaccount. When this attribute is missing or set to False (0), the demand bids for only the Participant's "Default" subaccount is returned without any subaccount attributes in the response message.
Yes	No	SubAccount	String; 20	The subaccount name can only be specified if the "subAccounts" attribute is set to True (1).

- Only one market date at a time can be queried.
- A limited number of Pnode IDs can be specified; see the maxOccurs attribute of the QueryByDayIdAndSubAccount type in the eMktMessages.xsd. They must however be listed in one of the user's Pnode portfolios. No data will be returned for an ID that is not listed in one of the user's Pnode portfolios.

- If the ID element is not specified then the query returns data for all the distinct Pnodes that are listed in all of the user's Pnode portfolios.
- A limited number of subaccount names can be specified when the "subAccounts" attribute is set to True (1); see the maxOccurs attribute of the QueryByDayIdAndSubAccount type in the eMktMessages.xsd. *If a specified subaccount name does not match any of the subaccounts registered by the Participant the request is rejected.*
- If the SubAccount element is not specified and the "subAccounts" is attribute set to True (1), then the query returns data for all subaccounts belonging to the Participant for which demand bids cleared, including the "Default" subaccount. ***Thus Participants without registered subaccounts in CAMS that submit queries with the "subAccounts" attribute set to True (1) will see a "subaccount" tag in the XML response message with "Default" as the subaccount name.***
- Query messages with a SubAccount element but no subAccounts attribute or the subAccounts attribute set to False (0) will be rejected.

6.3.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDemandByParticipant party="?" subAccounts="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
        <!--Zero or more repetitions:-->
        <mes:SubAccount?</mes:SubAccount>
      </mes:QueryFilters>
    </mes:GetDemandByParticipant>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetDemandByParticipant subAccounts="true">
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
    <SubAccount>Subaccount Name</SubAccount>
  </QueryFilters>
</GetDemandByParticipant>
```

6.3.1.4 Data Returned

Depending on whether the “subAccounts” attribute is present or set in the query message, the response message format will vary. When the attribute is omitted or set to False (0), the response will be a number of DemandByParticipant elements within the GetDemandByParticipantResponse element. When the attribute is set to True (1), the response will be a number of DemandByParticipantBySubAccount elements within the GetDemandByParticipantResponse element.

Format of response containing DemandByParticipant elements, i.e. response to a query with no “subAccounts” attribute or with attribute set to False (0):

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the Pnode
No	No	nodeName	String; 30	The name of the Pnode
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day
No	No	MW	MWType; Decimal 6.1	The MW amount that cleared
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDemandByParticipantResponse>
      <!--Zero or more repetitions:-->
      <mes:DemandByParticipant day="?" ID="?">
        <mes:nodeName?</mes:nodeName>
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" MW="?" price="?"/>
        </mes:HourlyProfile>
      </mes:DemandByParticipant>
    </mes:GetDemandByParticipantResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetDemandByParticipantResponse>
  <DemandByParticipant day="2010-05-05" ID="14227">
    <NodeName>SomeName</NodeName>
    <HourlyProfile>
      <HourlyValues time="2010-05-05T14:00:00-04:00"
        MW="12.3" price="34.05"/>
    </HourlyProfile>
  </DemandByParticipant>
</GetDemandByParticipantResponse>
```

Format of response containing DemandByParticipantBySubAccount elements, i.e. response to a query with “subAccounts” attribute set to True (1):

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	subAccount	String; 20	The subaccount name
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the Pnode
No	No	nodeName	String; 30	The name of the Pnode
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day
No	No	MW	MWType; Decimal 6.1	The MW amount that cleared
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)

Full SOAP Format

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDemandByParticipantResponse>
      <!--Zero or more repetitions:-->
      <mes:DemandByParticipantBySubAccount subAccount="?">
        <!--Zero or more repetitions:-->
        <mes:DemandByParticipant day="?" ID="?">
          <mes:NodeName?</mes:NodeName>
          <mes:HourlyProfile>
            <!--1 to 25 repetitions:-->
            <mes:HourlyValues time="?" MW="?" price="?"/>
          </mes:HourlyProfile>
        </mes:DemandByParticipant>
      </mes:DemandByParticipantBySubAccount>
    </mes:GetDemandByParticipantResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetDemandByParticipantResponse>
  <DemandByParticipantBySubAccount subAccount="Subaccount Name">
    <DemandByParticipant day="2010-05-05" ID="14227">
      <NodeName>SomeName</NodeName>
      <HourlyProfile>
        <HourlyValues time="2010-05-05T14:00:00-04:00"
          MW="12.3" price="34.05"/>
      </HourlyProfile>
    </DemandByParticipant>
  </DemandByParticipantBySubAccount>
</GetDemandByParticipantResponse>
```

6.4 Detailed Demand By Participant

The Detailed Demand By Participant operation allows Market Participants to get detailed information about their demand bids that cleared in the Day-Ahead market. It is a variation of the "Demand by Participant" query message described in section 6.3; instead of just the price and net MW amount that cleared in each hour of the market day, it returns the details of each demand bid that cleared.

6.4.1 Query Message

6.4.1.1 Purpose of Message

The purpose of this message is to get an XML report that lists the Market Participants' demand bids that cleared in the Day-Ahead market. The data returned in the report includes the market day and for each Pnode where demand bids cleared, the name and ID of the Pnode, and for each hour with cleared demand bids, a list of each and every demand bid that cleared including the bid type, the MW amount bid and the bid/offer price. For Market Participants with registered subaccounts, they can specify whether or not the demand bids are to be returned with subaccount information, and if so, they can explicitly specify the subaccount for which the demand bids are to be returned.

The data for this message is available after the Re-Offer period for the market day opens.

6.4.1.2 Mandatory and Optional Fields

To get the detailed demand by participant report, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of a Pnode
Yes	No	subAccounts	Boolean; true (1), false (0)	A flag used to specify whether the demand bid data is to be returned grouped by subaccount. When this attribute is missing or set to False (0), the demand bids for only the Participant's "Default" subaccount is returned without any subaccount attributes in the response message.
Yes	No	SubAccount	String; 20	The subaccount name can only be specified if the "subAccounts" attribute is set to True (1).

- Only one market date at a time can be queried.

- A limited number of Pnode IDs can be specified; see the maxOccurs attribute of the QueryByDayIdAndSubAccount type in the eMktMessages.xsd. They must however be listed in one of the user's Pnode portfolios. No data will be returned for an ID that is not listed in one of the user's Pnode portfolios.
- If the ID element is not specified then the query returns data for all the distinct Pnodes that are listed in all of the user's Pnode portfolios.
- A limited number of subaccount names can be specified when the "subAccounts" attribute is set to True (1); see the maxOccurs attribute of the QueryByDayIdAndSubAccount type in the eMktMessages.xsd. If a specified subaccount name does not match any of the subaccounts registered by the Participant the request is rejected.
- If the SubAccount element is not specified and the "subAccounts" is attribute set to True (1), then the query returns data for all subaccounts belonging to the Participant for which demand bids cleared, including the "Default" subaccount. **Thus Participants without registered subaccounts in CAMS that submit queries with the "subAccounts" attribute set to True (1) will see a "subaccount" tag in the XML response message with "Default" as the subaccount name.**
- Query messages with a SubAccount element but no subAccounts attribute or the subAccounts attribute set to False (0) will be rejected.

6.4.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDetailedDemandByParticipant party="?" subAccounts="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
        <!--Zero or more repetitions:-->
        <mes:SubAccount?</mes:SubAccount>
      </mes:QueryFilters>
    </mes:GetDetailedDemandByParticipant>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetDetailedDemandByParticipant>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetDetailedDemandByParticipant>
```

6.4.1.4 Data Returned

Depending on whether the “subAccounts” attribute is present or set in the query message, the response message format will vary. When the attribute is omitted or set to False (0), the response will be a number of DetailedDemandByParticipant elements within the GetDetailedDemandByParticipantResponse element. When the attribute is set to True (1), the response will be a number of DetailedDemandByParticipantBySubAccount elements within the GetDetailedDemandByParticipantResponse element.

Format of response containing DetailedDemandByParticipant elements, i.e. response to a query with no “subAccounts” attribute or with attribute set to False (0):

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the node
No	No	nodeName	String; 30	The name of the node
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
No	No	bidType	Enumerated String; Fixed, PriceSensitive, Decrement, Increment	The type of bid
Yes	No	fixedMW	MWType; Decimal 6.1	The MW amount of Fixed demand bids.
No	No	price ⁹	PriceType; Decimal 6.2	The bid price (\$/MWH) of PriceSensitive, Decrement and Increment bids
No	No	MW ⁹	MWType; Decimal 6.1	The MW amount of PriceSensitive, Decrement and Increment bids

⁹ This element will appear in the message only if it is a PriceSensitive, Decrement or Increment bid/offer.

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDetailedDemandByParticipantResponse>
      <!--Zero or more repetitions:-->
      <mes:DetailedDemandByParticipant bidType="?"
                                     day="?" ID="?">
        <mes:NodeName?</mes:NodeName>
        <mes:HourlyProfile>
          <!--0 to 25 repetitions:-->
          <mes:HourlyBid time="?" fixedMW="?">
            <!--0 to 9 repetitions:-->
            <mes:PricePoint price="?" MW="?"/>
          </mes:HourlyBid>
        </mes:HourlyProfile>
      </mes:DetailedDemandByParticipant>
    </mes:GetDetailedDemandByParticipantResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetDetailedDemandByParticipantResponse>
  <DetailedDemandByParticipant bidType="PriceSensitive"
    day="2010-05-05" ID="14227">
    <NodeName>SomeName</NodeName>
    <HourlyProfile>
      <HourlyBid time="2010-05-05T16:00:00-04:00">
        <PricePoint price="35" MW="76"/>
        <PricePoint price="25" MW="176"/>
      </HourlyBid>
      <HourlyBid time="2010-05-05T17:00:00-04:00">
        <PricePoint price="25" MW="180"/>
        <PricePoint price="75.75" MW="25"/>
      </HourlyBid>
    </HourlyProfile>
  </DetailedDemandByParticipant>
  <DetailedDemandByParticipant bidType="Decrement"
    day="2010-05-05" ID="14227">
    <NodeName>SomeName</NodeName>
    <HourlyProfile>
      <HourlyBid time="2010-05-05T16:00:00-04:00">
        <PricePoint price="52" MW="88"/>
      </HourlyBid>
    </HourlyProfile>
  </DetailedDemandByParticipant>
  <DetailedDemandByParticipant bidType="Increment"
    day="2010-05-05" ID="14227">
    <NodeName>SomeName</NodeName>
    <HourlyProfile>
      <HourlyBid time="2010-05-05T17:00:00-04:00">
        <PricePoint price="75" MW="77"/>
      </HourlyBid>
    </HourlyProfile>
  </DetailedDemandByParticipant>
  <DetailedDemandByParticipant bidType="Fixed"
    day="2010-05-05" ID="14227">
    <NodeName>SomeName</NodeName>
    <HourlyProfile>
      <HourlyBid time="2010-05-05T08:00:00-04:00" fixedMW="10"/>
      <HourlyBid time="2010-05-05T09:00:00-04:00" fixedMW="11"/>
      <HourlyBid time="2010-05-05T10:00:00-04:00" fixedMW="12"/>
      <HourlyBid time="2010-05-05T11:00:00-04:00" fixedMW="13"/>
      <HourlyBid time="2010-05-05T12:00:00-04:00" fixedMW="14"/>
      <HourlyBid time="2010-05-05T13:00:00-04:00" fixedMW="15"/>
      <HourlyBid time="2010-05-05T14:00:00-04:00" fixedMW="16"/>
      <HourlyBid time="2010-05-05T15:00:00-04:00" fixedMW="17"/>
    </HourlyProfile>
  </DetailedDemandByParticipant>
</GetDetailedDemandByParticipantResponse>
```

Format of response containing DetailedDemandByParticipantBySubAccount elements, i.e. response to a query with “subAccounts” attribute set to True (1):

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	subAccount	String; 20	The subaccount name
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the node
No	No	nodeName	String; 30	The name of the node
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
No	No	bidType	Enumerated String; Fixed, PriceSensitive, Decrement, Increment	The type of bid
Yes	No	fixedMW	MWType; Decimal 6.1	The MW amount of Fixed demand bids.
No	No	price ¹⁰	PriceType; Decimal 6.2	The bid price (\$/MWH) of PriceSensitive, Decrement and Increment bids
No	No	MW ¹⁰	MWType; Decimal 6.1	The MW amount of PriceSensitive, Decrement and Increment bids

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetDetailedDemandByParticipantResponse>
      <!--Zero or more repetitions:-->
      <mes:DetailedDemandByParticipantBySubAccount subAccount="?">
        <!--Zero or more repetitions:-->
        <mes:DetailedDemandByParticipant bidType="?">
          day="?" ID="?">
          <mes:nodeName>?</mes:nodeName>
          <mes:HourlyProfile>
            <!--0 to 25 repetitions:-->
            <mes:HourlyBid time="?" fixedMW="?">
              <!--0 to 9 repetitions:-->
              <mes:PricePoint price="?" MW="?" />
            </mes:HourlyBid>
          </mes:HourlyProfile>
        </mes:DetailedDemandByParticipant>
      </mes:DetailedDemandByParticipantBySubAccount>
    </mes:GetDetailedDemandByParticipantResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

¹⁰ This element will appear in the message only if it is a PriceSensitive, Decrement or Increment bid/offer.

Sample of Query Response

```
<GetDetailedDemandByParticipantResponse>
  <DetailedDemandByParticipantBySubAccount
    subAccount="Subaccount Name">
    <DetailedDemandByParticipant bidType="PriceSensitive"
      day="2010-05-05" ID="14227">
      <NodeName>SomeName</NodeName>
      <HourlyProfile>
        <HourlyBid time="2010-05-05T16:00:00-04:00">
          <PricePoint price="35" MW="76"/>
          <PricePoint price="25" MW="176"/>
        </HourlyBid>
        <HourlyBid time="2010-05-05T17:00:00-04:00">
          <PricePoint price="25" MW="180"/>
          <PricePoint price="75.75" MW="25"/>
        </HourlyBid>
      </HourlyProfile>
    </DetailedDemandByParticipant>
  </DetailedDemandByParticipantBySubAccount>
</GetDetailedDemandByParticipantResponse>
```

6.5 SubAccount

This operation is used to get the subaccounts in effect for a Market Participant on a given day.

6.5.1 Query Message

6.5.1.1 Purpose of Message

The purpose of this message is to get the subaccounts that belong to the Market Participant.

6.5.1.2 Mandatory and Optional Fields

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

6.5.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetSubAccount party="?">
      <mes:Day?</mes:Day>
    </mes:GetSubAccount>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetSubAccount>
  <Day>2015-02-14</Day>
</GetSubAccount/>
```

6.5.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Name	String; 20	The subaccount name
No	No	Default	Boolean; true (1), false (0)	Whether the subaccount is the default subaccount
No	No	Terminated	Boolean; true (1), false (0)	Whether the subaccount is terminated

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetSubAccountResponse>
      <!--0 or more repetitions!-->
      <mes:SubAccount>
        <mes:Name>?</mes:Name>
        <mes:Default>?</mes:Default>
        <mes:Terminated>?</mes:Terminated>
      </mes:SubAccount>
    </mes: GetSubAccountResponse >
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetSubAccountResponse>
  <SubAccount>
    <Name>Subaccount One</Name>
    <Default>true</Default>
    <Terminated>>false</Terminated>
  </SubAccount>
  <SubAccount>
    <Name>Subaccount Two</Name>
    <Default>>false</Default>
    <Terminated>>false</Terminated>
  </SubAccount>
  <SubAccount>
    <Name>Subaccount Three</Name>
    <Default>>false</Default>
    <Terminated>>true</Terminated>
  </SubAccount>
</GetSubAccountResponse>
```

6.6 Load Response Offer

A load response offer is the price offered for a reduction of load by a Price Response Demand resource¹¹. A load response enabled demand resource is able to decrease load in response to instructions it receives from ISO New England.

6.6.1 Submit Message

6.6.1.1 Purpose of Message

The purpose of this message is to submit load response offers. This message contains data relevant to submitting load response offers, such as the name of the load response resource, start/stop day, offer price/MW, minimum interruption duration, and curtailment initiation price.

If a load response offer is to be used in the Day-Ahead Market, it must be submitted before the market closes the day before the operating day.

6.6.1.2 Mandatory and Optional Fields

To create or modify a load response offer, you need to provide the all of the following data. The optional element AssetID is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	name	String; 40	The name of the load response asset
Yes	No	AssetID	Long	The asset ID. Read Only
No	No	CurtailmentInitiationPrice	PriceType; Decimal 6.2	The price per interruption at which the amount of interruption is offered. This price is paid each time a load reduction is scheduled.
No	No	MinimumInterruptionDuration	ElapsedTimeType; Integer	The minimum period of time in hours that the load reduction must be scheduled, if cleared.
No	No	OfferMW	LRMWType; Decimal 6.3	The MW amount of load reduction offered.
No	No	OfferPrice	PriceType; Decimal 6.2	The offer price per MW (\$ per MWH) for the quoted load reduction
No	No	StartDay	Date; yyyy-mm-dd	The first day of the period for which the offer is valid
No	No	StopDay	Date; yyyy-mm-dd	The last day of the period for which the offer is valid

¹¹ The Price Response Demand (PRD) program replaced the DA Load Response program in 2012

Data for a limited number of asset names can be submitted in one request; refer to the maxOccurs attribute of the SubmitLoadResponseOffer message in the eMktMessages.xsd.

6.6.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitLoadResponseOffer party="?">
      <!--1 or more repetitions:-->
      <mes:LoadResponseOffer name="?">
        <!--Optional:-->
        <mes:AssetID?></mes:AssetID>
        <mes:CurtailmentInitiationPrice?>
          </mes:CurtailmentInitiationPrice>
        <mes:MinimumInterruptionDuration?>
          </mes:MinimumInterruptionDuration>
        <mes:OfferMW?></mes:OfferMW>
        <mes:OfferPrice?></mes:OfferPrice>
        <mes:StartDay?></mes:StartDay>
        <mes:StopDay?></mes:StopDay>
      </mes:LoadResponseOffer>
    </mes:SubmitLoadResponseOffer>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitLoadResponseOffer>
  <LoadResponseOffer name="Load Resp 1">
    <CurtailmentInitiationPrice>50</CurtailmentInitiationPrice>
    <MinimumInterruptionDuration>2</MinimumInterruptionDuration>
    <OfferMW>1.0</OfferMW>
    <OfferPrice>11.12</OfferPrice>
    <StartDay>2010-07-07</StartDay>
    <StopDay>2010-07-14</StopDay>
  </LoadResponseOffer>
</SubmitLoadResponseOffer>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

6.6.1.4 Additional Value Restrictions

- OfferMW \geq 0.1 MW
- OfferMW \leq the minimum of the **Maximum Interruptible Capacity (MIC)** values of the Load Response Resource for the dates spanned by the Load Response Offer at the time the offer is submitted. Note that the monthly MIC of a resource can change at any time, even after

an offer has been submitted and accepted. The market clearing software will use the lesser of the MIC effective for the market day and the OfferMW of the offer.

- OfferPrice \geq maximum of the monthly **Demand Response Threshold Price (DRTP)** for the months spanned by the Load Response Offer. Under the **Price Response Demand (PRD) Program**, the DRTP is calculated on a monthly basis as defined in Market Rule 1, Appendix E and posted to the ISO New England website by the 15th day of the preceding month in advance of the Demand Reduction Threshold Price's effective date. The DRTP applies to only the next month and does not change the DRTP for the current month.
- OfferPrice < the maximum defined in Market Rule 1, Appendix E (e.g. \$1000)
- CurtailmentInitiationPrice < the maximum defined in Market Rule 1, Appendix E (e.g. \$1000)
- The Minimum Interruption Duration must be defined in whole hourly increments
- $1 \leq$ Minimum Interruption Duration ≤ 4
- StartDay must be a date in the future
- StopDay \geq StartDay
Under the **Price Response Demand (PRD) Program**, the StopDay date must fall in a month where the Demand Response Threshold Price (DRTP) is defined; in other words, StopDay date \leq date of the last day of a month where the DRTP is defined. Since the DRTP of a month will be defined and published by the 15th day of the preceding month in advance of the Demand Reduction Threshold Price's effective date, Load Response Offers for the next month can be submitted only after the DRTP is published.
- If a Load Response offer is submitted for a range of days and a new offer is submitted for a day or a range of days within that offer range, the newly entered offer will overwrite the previous offer data for only those day(s) specified in the new offer.

6.6.2 Delete Message

6.6.2.1 Purpose of Message

The purpose of this message is to delete a load response offer.

6.6.2.2 Mandatory and Optional Fields

To delete a load response offer, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Name	String; 40	The name of the load response offer to be deleted
No	No	StartDate	Date; yyyy-mm-dd	The specified starting day

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	StopDay	Date; yyyy-mm-dd	The specified ending day, inclusive

6.6.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:DeleteLoadResponseOffer party="??">
      <mes:Name?</mes:Name>
      <mes:StartDay?</mes:StartDay>
      <mes:StopDay?</mes:StopDay>
    </mes:DeleteLoadResponseOffer>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Delete Submittal

```
<DeleteLoadResponseOffer>
  <Name>Name</Name>
  <StartDay>2010-07-07</StartDay>
  <StopDay>2010-07-08</StopDay>
</DeleteLoadResponseOffer>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

6.6.2.4 Additional Value Restrictions

The delete load response offer request will delete all Load Response Offers or portions thereof that fall within the delete date range or span either the StartDay date or the StopDay date of the range; for example,

- If Offer Start Day \geq Delete StartDay and Offer StopDay \leq Delete StopDay, then the entire offer is deleted
- If Offer Start Day $<$ Delete StartDay and Offer StopDay $>$ Delete StopDay, then the Load Response Offer for the range Delete StartDay thru Delete StopDay is deleted, resulting in two Load Response Offers remaining: one for the portion of the offer spanning the original offer StartDay thru the day before the Delete StartDay, and another for the portion spanning the day after the Delete StopDay thru the original offer StopDay
- If Offer Start Day $<$ Delete StartDay and Offer StopDay \leq Delete StopDay, then only the portion of the offer spanning original StartDay thru the day before the Delete StartDay will remain

- If Offer Start Day \geq Delete StartDay and Offer StopDay $>$ Delete StopDay, then only the portion of the offer spanning the day after the Delete StopDay thru the original offer StopDay will remain

If the delete load response offer request date range does not span any Load Response Offer, the request will be rejected.

6.6.3 Query Message

6.6.3.1 Purpose of Message

The purpose of this message is to get submitted load response offers. The data returned includes the name and ID of the load response asset, the first and last day of the offer, the offer price, the MW load reduction offered, the minimum interruption duration, and the curtailment initiation price.

6.6.3.2 Mandatory and Optional Fields

To get or query a load response offer, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Name	String; 40	The name of the load response asset

- A limited number of load response names can be specified per query; see the maxOccurs attribute of the GetLoadResponseOffer message in the eMktMessages.xsd

6.6.3.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetLoadResponseOffer party="?">
      <!--1 or more repetitions:-->
      <mes:Name?</mes:Name>
    </mes:GetLoadResponseOffer>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetLoadResponseOffer>
  <Name>AssetName</Name>
</GetLoadResponseOffer>
```

6.6.3.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	AssetID	Long	The asset ID. Read Only

Response Elements

```
<GetLoadResponseOfferResponse>  
  ...  
  <AssetID>12345</AssetID>  
  ...  
</GetLoadResponseOfferResponse>
```

6.7 Load Response Result

See section 6.5 for a definition of a load response.

6.7.1 Query Message

6.7.1.1 Purpose of Message

This purpose of this message is to get the load response offers that cleared in the Day-Ahead Market for a given Market Day. This response will contain information about the load response offers such as the time when the load is desired, the amount of MW desired, the LMP, etc.

The data for this message is available after the Re-Offer period for the market day opens.

6.7.1.2 Mandatory and Optional Fields

To get load response offers, you need to provide the following.

Opt.	Element or Attribute	Data Type; Format	Comments
No	Day	Date; yyyy-mm-dd	The market day

Only one market date at a time can be queried.

6.7.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetLoadResponseResult party="??">
      <mes:Day?></mes:Day?>
    </mes:GetLoadResponseResult>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetLoadResponseResult>
  <Day>2010-07-07</Day>
</GetLoadResponseResult>
```

6.7.1.4 Data Returned

Opt.	Element or Attribute	Data Type; Format	Comments
No	day	Date; yyyy-mm-dd	The market day
No	ID	Long	The ID of the load response
No	Name	String; 40	The name of the load response

Opt.	Element or Attribute	Data Type; Format	Comments
No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
No	LMP	PriceType; Decimal 6.2	The locational marginal price (\$ per MWH)
No	MW	MWType; Decimal 6.1	The MW amount approved

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetLoadResponseResultResponse>
      <!--Zero or more repetitions:-->
      <mes:LoadResponseResult day="?" ID="?">
        <mes:Name?</mes:Name>
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" LMP="?" MW="?"/>
        </mes:HourlyProfile>
      </mes:LoadResponseResult>
    </mes:GetLoadResponseResultResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetLoadResponseResultResponse>
  <LoadResponseResult day="2010-07-07" ID="12345">
    <Name>LoadResponseName</Name>
    <HourlyProfile>
      <HourlyValues time="2010-07-07T14:00:00-04:00"
        LMP="34.5" MW="4.5"/>
    </HourlyProfile>
  </LoadResponseResult>
</GetLoadResponseResultResponse>
```

7. Generation Web Services

This section defines the submittals and queries used for Generation operations.

The table below shows a summary of the Generation messages and indicates if data can be submitted, queried or deleted. It also shows when and how long submitted offer data is effective. As a rule, it is not necessary to submit data for each market day. With most messages, the submitted data is effective starting on the market day specified in the message and for all subsequent days up to the day for which data is submitted in a separate message. Hourly offer data is the exception as it only applies to a specific market day and hour.

Message	Brief Description of Usage	Actions			Data Remains Effective	
		Submit	Query	Delete	Until data is submitted for a future day	Only for the specified day and hour
Generation By Participant	View the Generation offers that cleared in the DA Market		X		n/a	n/a
Generation Portfolio	Manage Portfolios of Units (Generation assets)	X	X	X	n/a See Note 3	n/a
Schedule	Manage the Schedule container definitions. The following messages contain the details of the schedules.	X	X	X	X	
Schedule Bid	Set or view the MW/Price pairs for a given schedule	X	X	See Note 2	X	
Schedule Bid Hourly	Set or view the Hourly Overrides for MW/Price pairs for a given schedule	X	X	See Note 2		X
Schedule Detail	Set or view the Unit offer data for a given schedule.	X	X		X	
Schedule Times Hourly	Set or view the Hourly Overrides for startup and notification times for a given schedule	X	X	See Note 2		X
Unit Schedule Selection	Set or view the availability of a schedule for a given market and day.	X	X		X	
Unit	Set or view the default offer data for a Unit	X	X		X	
Unit Hourly	Set or view the Hourly Overrides which apply to all schedules for a day.	X	X	See Note 1		X

Message	Brief Description of Usage	Actions			Data Remains Effective	
		Submit	Query	Delete	Until data is submitted for a future day	Only for the specified day and hour
Unit Ramp Rate	Set or view the Ramp Rate Curve. Used when a multi-segment curve is desired rather than the simple slope defined with the Unit message; it overrides the simple slope.	X	X	See Note 1	X	
Unit Ramp Rate Hourly	Set or view the Hourly Ramp Rate Curve Overrides.	X	X	See Note 1		X

Note 1: There is no “Delete” web service operation defined to explicitly delete a unit Ramp Rate, Ramp Rate Hourly, or a unit Hourly Offer once a set of data is submitted. Nor is there an operation defined to delete only a segment of the multi-segment curve data previously submitted. A whole new set of curve data or hourly data must be submitted. There is however provision in the “Submit” web service to remove a set of curve data by submitting an empty container. Similarly, there is provision in the “Submit” web services for Unit Hourly, Ramp Rate Hourly, Schedule Offers Hourly, and Schedule Times Hourly offers to remove a specific hourly update offer by submitting a message with the “delete” attribute set. See section 2.3 on deletion operations for more information.

Note 2: There is no operation defined to delete a particular MW/Price pair from a set of previously submitted bid data. A whole set of MW/Price pairs must be submitted to replace the previously submitted set.

Note 3: Portfolios are not effective dated. The data in a portfolio exists from when the portfolio is created until the portfolio is deleted. The data (asset list) in a portfolio can be updated using the SubmitPortfolio request; note however that the complete list of asset IDs must be submitted to replace the previously submitted list, if any.

When submitting data, it is important to understand the hierarchy of the Generation messages and the implications of data overrides. Figure 2 below shows the hierarchy of the Generation messages used for submittals, where the same bid data appears in different messages, and which messages have precedence over (overrides) the others. **[Note:** only the data elements applicable to this example are shown. See the following sections for the details on the data in each message.]

The Unit message is at the root of the hierarchy. The Unit record itself is initially defined by ISO New England with default offer data when the asset is instantiated in the Market system. Participants must update some data values, like the DefaultStatus, using the Unit message.

To participate in the energy market, a Participant must submit at least once a Schedule message for the unit and all of the child messages that go with it. The Schedule message defines a schedule container for which Bid and Detail data must be submitted. The Schedule primarily defines the names and type of a schedule for a unit.

Once the schedule is created, then the following three messages must be used to fill out that schedule for the market:

- Schedule Bid: defines the MW/Price Pairs, Startup and No Load Costs, and Fuel Blends; i.e. energy offer data
- Schedule Detail: defines limits and other offer data
- Schedule Selection: sets the schedule as available or unavailable for the DA and/or the RT market.

For units that are eligible to participate in the Regulation Market, a separate Regulation Offer message must be submitted (see §8 for a description of the Regulation messages).

Five additional messages allow for overriding schedule based or default data. Offer data in other messages higher in the hierarchy are overridden by the presence of data in these messages:

- The Unit Ramp Rate message defines a ramp rate curve that overrides the simple slope defined with the Unit message.
- The Unit Ramp Rate Hourly message overrides the ramp rate curve provided with the Unit Ramp Rate message or the simple slope defined with the Unit message (one can specify either an hourly curve or single ramp rate value).
- The Unit Hourly message overrides data for all schedules and defaults for a given day.
 - Unit Hourly limits (EmerMin, EcoMax, EcoMin, and RTHighOperLimit) override for the specified hour of the day the corresponding limit values defined in the Schedule Detail of all schedules.
 - Unit Hourly ResourceStatus overrides for the specified hour of the day the DefaultStatus defined with the Unit message.
- The Schedule Bid Hourly message overrides MW/Price pairs, Startup/No Load Costs, and Fuel Blends provided with the Schedule Bid message. This is the only message that can be submitted during the RT Intraday Re-Offer period.
- The Schedule Times Hourly message overrides Startup and Notification times provided with the Schedule Detail message.

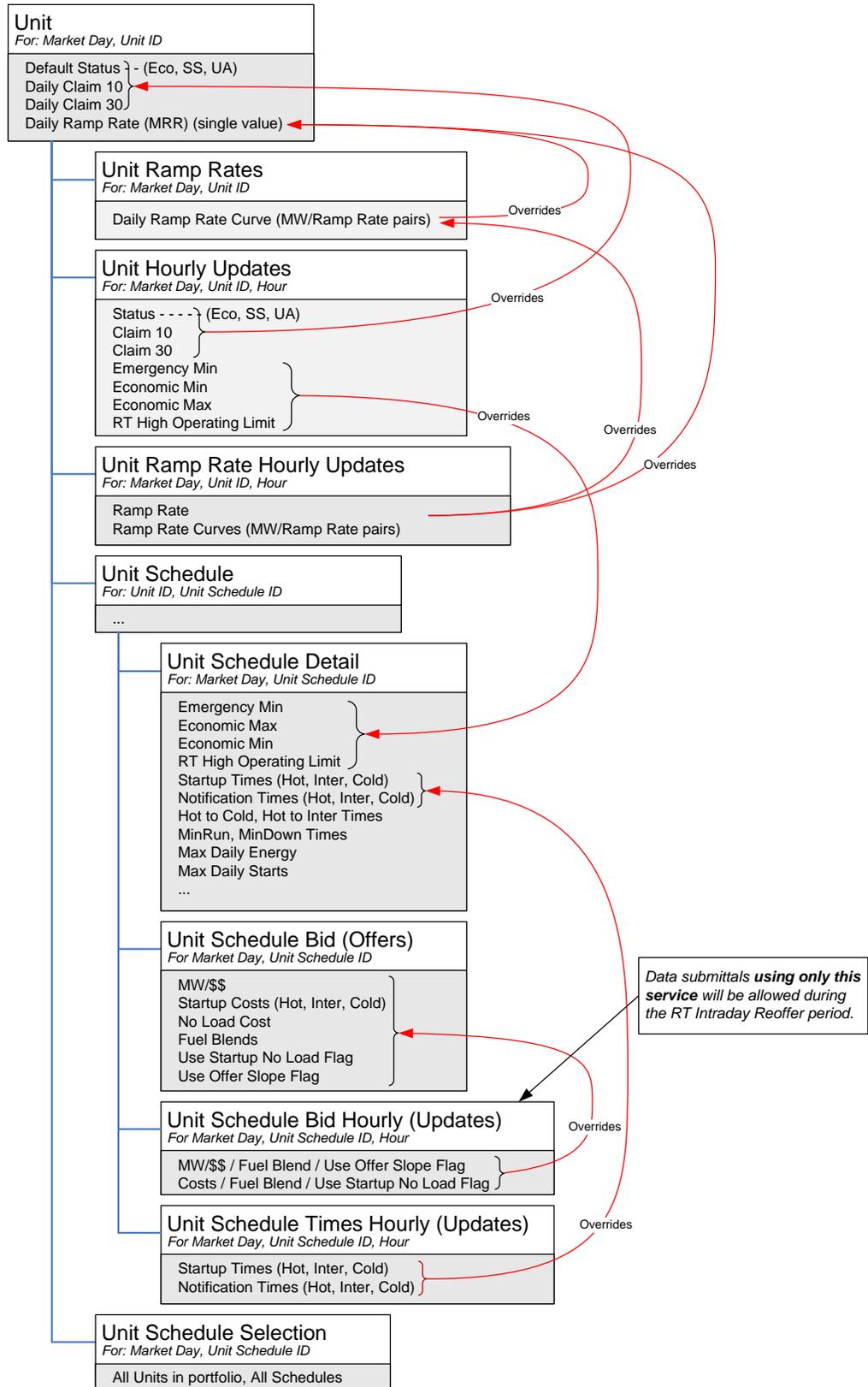


Figure 2 - Illustration of Generation Message Hierarchy and Data Precedence

7.1 Generation By Participant

The Generation By Participant operation allows Market Participants to get information about their generation offers that cleared in the Day-Ahead Market.

7.1.1 Query Message

7.1.1.1 Purpose of Message

The purpose of this message is to get details about a Market Participants' unit generation offers that cleared in the Day-Ahead Market for a specified market day and for an explicitly or implicitly defined list of units. Data returned in this report includes the price per MW bid and the number of MW cleared in the Day-Ahead Market for a particular hour of the specified market day, as well as the asset's name, the name and ID of the schedule.

The data for this message is available after the Re-Offer period for the market day opens.

7.1.1.2 Mandatory and Optional Fields

To get the report of the Market Participants' cleared offers, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of a unit

- Data for only one market date at a time can be queried.
- A limited number of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must be listed in one of the user's Generation portfolios. No data will be returned for an ID that is not listed in one of the user's Generation portfolios.
- If the ID element is not specified then the query returns data for all the distinct units that are listed in all of the user's Generation portfolios.

7.1.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetGenByParticipant party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetGenByParticipant>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetGenByParticipant>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetGenByParticipant>
```

7.1.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the unit
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day
No	No	MW	MWType; Decimal 6.1	The MW amount that cleared
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)
No	No	priceCapped	Boolean; true (1), false (0)	Whether the price is capped
No	No	UnitShortName	String; 8	The short name of the unit
No	No	UnitScheduleID	Long	The ID of the schedule
No	No	UnitScheduleShortName	String; 8	The short name of schedule

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetGenByParticipantResponse>
      <!--Zero or more repetitions:-->
      <mes:GenByParticipant day="?" ID="?">
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" MW="?" price="?"
            priceCapped="?"/>
        </mes:HourlyProfile>
        <mes:UnitShortName?></mes:UnitShortName>
        <mes:UnitScheduleID?></mes:UnitScheduleID>
        <mes:UnitScheduleShortName?></mes:UnitScheduleShortName>
      </mes:GenByParticipant>
    </mes:GetGenByParticipantResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetGenByParticipantResponse>
  <GenByParticipant day="2010-05-05" ID="473">
    <HourlyProfile>
      <HourlyValues time="2010-05-05T14:00:00-04:00"
        MW="10.5" price="54.29" priceCapped="false"/>
    </HourlyProfile>
    <UnitShortName>UN11</UnitShortName>
    <UnitScheduleID>12345</UnitScheduleID>
    <UnitScheduleShortName>DefPrice</UnitScheduleShortName>
  </GenByParticipant>
</GetGenByParticipantResponse>
```

7.2 Generation Portfolio

See section 4.4 for a definition of a portfolio.

7.2.1 Submit Message

This functionality is provided by the Portfolio's Submit Message described in section 4.4.1

7.2.2 Delete Message

This functionality is provided by the Portfolio's Delete Message described in section 4.4.2

7.2.3 Query Message

This functionality is provided by the Portfolio's Query Message described in section 4.4.3

7.3 Schedule

See section 4.5 for a definition of a schedule.

7.3.1 Submit Message

7.3.1.1 Purpose of Message

The purpose of this message is to create a schedule for a unit that can then be used to submit bids into the Market system. To create a schedule the request must specify the asset ID, a schedule type ID and provide a long and short name for the schedule. The message can also be used to change/modify the name attributes of a previously created schedule.

7.3.1.2 Mandatory and Optional Fields

To create or modify a schedule, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	No	schedID	Long	The schedule ID
Yes	No	SchedLongName	String; 40	The long name or description of the schedule
Yes	No	SchedShortName	String; 8	The short name of the schedule
No	No	SchedTypeID	Long	The schedule type ID
No	No	UnitID	Long	The unit ID

- The attribute schedID is required only when modifying any of the elements of a previously created schedule. It is not required when creating a new schedule because it will be created automatically.
- The schedule type ID must be one of a price-based schedule. The response to the Schedule Type query message (section 4.5) lists all the IDs of price-based schedules.
- Data for a limited number of unit IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitSchedule message in the eMktMessages.xsd.

7.3.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitSchedule party="?">
      <!--1 or more repetitions:-->
      <mes:Schedule schedID="?">
        <mes:SchedLongName?</mes:SchedLongName>
        <mes:SchedShortName?</mes:SchedShortName>
        <mes:SchedTypeID?</mes:SchedTypeID>
        <mes:UnitID?</mes:UnitID>
      </mes:Schedule>
    </mes:SubmitSchedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitSchedule>
  <Schedule schedID="123412">
    <SchedLongName>LongName</SchedLongName>
    <SchedShortName>SchdNme</SchedShortName>
    <SchedTypeName>Price</SchedTypeName>
    <SchedTypeNum>12</SchedTypeNum>
    <UnitID>1234</UnitID>
  </Schedule>
</SubmitSchedule>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.3.1.4 Additional Value Restrictions

- Valid unit IDs and unit Schedule identifiers must be submitted

7.3.2 Delete Message

7.3.2.1 Purpose of Message

The purpose of this message is to delete a unit schedule.

7.3.2.2 Mandatory and Optional Fields

To delete a unit schedule, you need to provide the following:

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	SchedID	Long	The ID of the schedule to be deleted

7.3.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:DeleteSchedule party="?">
      <mes:SchedID?</mes:SchedID>
    </mes:DeleteSchedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Delete Submittal

```
<DeleteSchedule>
  <SchedID>12345</SchedID>
</DeleteSchedule>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.3.2.4 Additional Value Restrictions

A given schedule cannot be deleted when any of the following conditions exist:

- The schedule is marked as "available" for any day in the past, present, or future within the data retention period of the market database (14 days in the past).
- The unit is committed on the schedule for any day within the data retention period of the market database.

If any of these conditions exist, the attempt to delete the schedule will result in an error.

7.3.3 Query Message

7.3.3.1 Purpose of Message

The purpose of this message is to get information about all the schedules associated with one or more units, specifically the long and short name of the schedules, the schedule type IDs, and the name and ID of the unit.

7.3.3.2 Mandatory and Optional Fields

To get or query a unit schedule, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	UnitID	Long	The unit ID

- A limited number of unit IDs can be specified; see the maxOccurs attribute of the GetSchedule message in the eMktMessages.xsd.
- If a unit ID is not specified then the query returns data for every schedule of every unit that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

7.3.3.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetSchedule party=""?>
      <!--1 or more repetitions:-->
      <mes:UnitID?</mes:UnitID>
    </mes:GetSchedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetSchedule>
  <UnitID>123456</UnitID>
</GetSchedule>
```

7.3.3.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	SchedType	Enumerated String; Cost, Price	The schedule type
No	No	UnitShortName	String; 8	The unit short name

Response Elements

```
<ScheduleResponse>
  ...
  <SchedType>Price</SchedType>
  <UnitShortName>ShrtNme</UnitShortName>
  ...
</ScheduleResponse>
```

7.4 Schedule Bid

See section 4.5 for a definition of a schedule.

7.4.1 Submit Message

7.4.1.1 Purpose of Message

The purpose of this message is to submit energy offer data under a schedule for a generating unit. This message contains data relevant to submitting offers for energy, specifically the market day for the offer, the price/MW offer blocks, Startup and No Load Costs, Fuel Blends, and the schedule ID.

7.4.1.2 Mandatory and Optional Fields

To create or modify a unit schedule bid, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)
No	No	MW	MWType; Decimal 6.1	The upper bound of the MW generation range for the offered price. The lower bound is the upper bound MW value that is defined for a different price in the same message and that is less than this value; if one is not defined, the lower bound is zero.
No	No	Cost	CostType; Decimal 10.2	The cost associated with Cold, Intermediate, and Hot Startup and No Load
Yes ¹²	No	Name	String;20	Fuel Index name
Yes ¹²	No	Percentage	PercentType; Decimal 3.0	The percentage associated with a given fuel index
Yes	No	ShortName	String; 8	The short name of the schedule. Read Only
No	No	UseOfferSlope	Boolean; true (1), false (0)	Whether to interpolate the price between the MW upper bound in a generator offer curve (Price/MW offer blocks)
No	No	UseStartupNoLoad	Boolean; true (1), false (0)	Whether the market must use the unit's startup/noload cost when evaluating the unit for commitment. A Participant can waive these costs by setting the flag to false

¹² Optional only for single-fuel units. The percentage is defaulted to 100 for single-fuel units.

- The fuel index must be one that the unit is configured to use. The list of fuel indices that a unit is configured to use can be obtained using the GetUnit query message described in §7.9.2
- Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitScheduleBid message in the eMktMessages.xsd.

7.4.1.3 SOAP Format

Full SOAP Message

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitScheduleBid party="?">
      <!--1 or more repetitions:-->
      <mes:ScheduleBid day="?" ID="?">
        <mes:PriceCurve>
          <!--1 to 10 repetitions:-->
          <mes:PricePoint price="?" MW="?">
            <!--0 to 4 repetitions:-->
            <mes:FuelIndex>
              <mes:Name?</mes:Name>
              <mes:Percentage?</mes:Percentage>
            </mes:FuelIndex>
          </mes:PricePoint>
        </mes:PriceCurve>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
        <mes:UseOfferSlope?</mes:UseOfferSlope>
        <mes:UseStartupNoLoad?</mes:UseStartupNoLoad>
        <mes:ColdStartupCostAndFuel>
          <mes:Cost?</mes:Cost>
          <!--0 to 4 repetitions:-->
          <mes:FuelIndex>
            <mes:Name?</mes:Name>
            <mes:Percentage?</mes:Percentage>
          </mes:FuelIndex>
        </mes:ColdStartupCostAndFuel>
        <mes:HotStartupCostAndFuel>
          <mes:Cost?</mes:Cost>
          <!--0 to 4 repetitions:-->
          <mes:FuelIndex>
            <mes:Name?</mes:Name>
            <mes:Percentage?</mes:Percentage>
          </mes:FuelIndex>
        </mes:HotStartupCostAndFuel>
        <mes:InterStartupCostAndFuel>
          <mes:Cost?</mes:Cost>
          <!--0 to 4 repetitions:-->
          <mes:FuelIndex>
            <mes:Name?</mes:Name>
            <mes:Percentage?</mes:Percentage>
          </mes:FuelIndex>
        </mes:InterStartupCostAndFuel>
      </mes:ScheduleBid>
    </mes:SubmitScheduleBid>
  </soapenv:Body>
</soapenv:Envelope>

```

```

        </mes:FuelIndex>
    </mes:InterStartupCostAndFuel>
    <mes:NoLoadStartupCostAndFuel>
        <mes:Cost?></mes:Cost>
        <!--0 to 4 repetitions:-->
        <mes:FuelIndex>
            <mes:Name?></mes:Name>
            <mes:Percentage?></mes:Percentage>
        </mes:FuelIndex>
    </mes:NoLoadStartupCostAndFuel>
</mes:ScheduleBid>
</mes:SubmitScheduleBid>
</soapenv:Body>
</soapenv:Envelope>

```

Sample of Submittal

```

<SubmitScheduleBid>
  <ScheduleBid day="2014-04-17" ID="1351596">
    <PriceCurve>
      <PricePoint price="328.28" MW="88">
        <FuelIndex>
          <Name>MA_natgas1</Name>
          <Percentage>80</Percentage>
        </FuelIndex>
        <FuelIndex>
          <Name>ULSDiesel</Name>
          <Percentage>20</Percentage>
        </FuelIndex>
      </PricePoint>
    </PriceCurve>
    <UseStartupNoLoad>true</UseStartupNoLoad>
    <UseOfferSlope>false</UseOfferSlope>
    <HotStartupCostAndFuel>
      <Cost>5954.31</Cost>
      <FuelIndex>
        <Name>MA_natgas1</Name>
        <Percentage>80</Percentage>
      </FuelIndex>
      <FuelIndex>
        <Name>ULSDiesel</Name>
        <Percentage>20</Percentage>
      </FuelIndex>
    </HotStartupCostAndFuel>
    <InterStartupCostAndFuel>
      <Cost>5954.31</Cost>
      <FuelIndex>
        <Name>MA_natgas1</Name>
        <Percentage>80</Percentage>
      </FuelIndex>
      <FuelIndex>
        <Name>ULSDiesel</Name>
        <Percentage>20</Percentage>
      </FuelIndex>
    </InterStartupCostAndFuel>
    <ColdStartupCostAndFuel>
      <Cost>5954.31</Cost>
    </ColdStartupCostAndFuel>
  </ScheduleBid>
</SubmitScheduleBid>

```

```

<FuelIndex>
  <Name>MA_natgas1</Name>
  <Percentage>80</Percentage>
</FuelIndex>
<FuelIndex>
  <Name>ULSDiesel</Name>
  <Percentage>20</Percentage>
</FuelIndex>
</ColdStartupCostAndFuel>
<NoLoadCostAndFuel>
  <Cost>531.16</Cost>
  <FuelIndex>
    <Name>MA_natgas1</Name>
    <Percentage>80</Percentage>
  </FuelIndex>
  <FuelIndex>
    <Name>ULSDiesel</Name>
    <Percentage>20</Percentage>
  </FuelIndex>
</NoLoadCostAndFuel>
<ShortName>ALP_RS</ShortName>
</ScheduleBid>
</SubmitScheduleBid>

```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.4.1.4 Additional Value Restrictions

- $MW \geq 0$
- Price of Incremental Energy Offers must be less than or equal to the Energy Offer Cap price and greater than or equal to the Energy Offer Floor price; i.e.
 $\text{Energy Offer Floor price} \leq \text{Price} \leq \text{Energy Offer Cap price}^{13}$.
- Complete sets of Price/MW offer pairs with fuel blend data must be submitted; they cannot be submitted separately and independently. A maximum of ten Price/MW offer pairs per schedule can be submitted.
- Offers must be monotonically increasing (price and MW); i.e. the price for a given MW upper bound must be greater than or equal to the price for a lower MW upper bound. Note that Price/MW bid pairs don't need to be submitted in any particular order. However the MW values of each pair must be different from the others such that the MW value for a pair must be greater than the MW of the previous pair after the Price/MW pairs are ordered in increasing MW value. The monotonically increasing rule is enforced after the complete set of pairs are received, ordered and processed.
- $0 \leq \text{HotStartupCost} \leq \text{InterStartupCost} \leq \text{ColdStartupCost}$
- $0 \leq \text{NoLoadCost}$

¹³ Energy Offer Floor price and Energy Offer Cap price are both defined in Market Rule 1, section I.2.2.

- Fuel Blend percentages must be submitted for every fuel that a unit is configured to use. The list of fuel indices that a unit is configured to use can be obtained using the GetUnit query message described in §7.9.2
- Fuel Blend percentages must add up to 100%. Fuel Blend percentages (fuel name and percentage) must be submitted for all financial offers: Price/MW pairs, Startup Costs and NoLoad Cost.
- For single-fuel units, the fuel blend percentage are optional (as indicated by a minOccurs=0 for the FuelIndex element in eMktMessages.xsd) because they default to 100; if a name is provided however, a percentage value of 100 is also required.
- Cost-based schedules cannot be modified

7.4.1.5 Checking of Offer Parameter Values against Market Monitoring Thresholds

An error message will be returned and the submit data request rejected if any of the following offer parameters exceed their threshold value established by the ISO-NE Market Monitor¹⁴:

- Cold, Intermediate and Hot Startup Costs
- No Load Cost

For multi-fuel units, the thresholds are dynamically calculated using the unit's reference levels and the fuel blend information submitted with the cost offers.

7.4.2 Query Message

7.4.2.1 Purpose of Message

The purpose of this message is to get the unit schedule default offer data for a specified market day. The response will contain the default offer data for the specified market day and for a list of unit schedules. The data returned includes the market day, the price/MW offer blocks, and the ID and short name of the schedules.

7.4.2.2 Mandatory and Optional Fields

To get or query unit schedule bids, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the schedule

- Data for only one market date at a time can be queried.
- A limited number of schedule IDs can be listed; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all be associated with units that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

¹⁴ Refer to Market Rule 1, Appendix A, Section III.A.5.5.

- If a schedule ID is not specified then the query returns data for every schedule of every unit that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

7.4.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleBid party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetScheduleBid>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetScheduleBid>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetScheduleBid>
```

7.4.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in increasing MW value order.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the schedule

Response Elements

```
<GetScheduleBidResponse>
  ...
  <ShortName>SchdNme</ShortName>
  ...
</GetScheduleBidResponse>
```

7.5 Schedule Bid Hourly

See section 4.5 for a definition of a schedule.

7.5.1 Submit Message

7.5.1.1 Purpose of Message

The purpose of this message is to submit hourly energy offer overrides under a schedule for a generating unit. This message contains data relevant to submitting offers for energy, specifically the market day and hour for the offer, the price/MW offer blocks, Startup and No Load Costs, Fuel Blends, Use Startup No Load and Use Offer Slope flags, and the schedule ID. The message can also be used to delete an hourly update by simply setting the “delete” attribute to TRUE on the message element for the hour to delete. During the RT Intraday Re-Offer period, this is the only message that can be used to submit hourly updates of the unit’s offers for the current operating day and hourly updates to **only** the unit’s schedule being used in the RT market (that is the schedule designated to be used for the DA market if the unit was committed in the DA market, else it is the schedule designated to be used for the RT market; they can be the same schedule and they can be different schedules).

7.5.1.2 Mandatory and Optional Fields

To create or modify a unit schedule bid, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	delete	Boolean; true(1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
No	No	ID	Long	The id of the schedule
No	No	price	PriceType; Decimal 6.2	The price per MW (\$ per MWH)

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	MW	MWType; Decimal 6.1	The upper bound of the MW consumption range for the offered price. The lower bound is the upper bound MW value that is defined for a different price in the same message and that is less than this value; if one is not defined, the lower bound is zero.
Yes ¹⁵	No	Name	String; 40	Name of the fuel index
Yes ¹⁵	No	Percentage	PercentType; Decimal 3.0	Fuel mix percentage
No	No	Cost	PriceType; Decimal 10.2	The NoLoad, HotStartup, InterStartup and ColdStartup cost values
No	No	UseStartupNoLoad	Boolean; true (1), false (0)	Whether the market must use the unit's Startup/NoLoad cost when evaluating the unit for commitment. A Participant can waive these costs by setting the flag to false
No	No	UseOfferSlope	Boolean; true (1), false (0)	Whether to interpolate the price between the MW upper bound in an bid curve (Price/MW bid blocks)
Yes	No	ShortName	String; 8	The short name of the unit. Read Only.

¹⁵ Optional only for single-fuel units. The percentage is defaulted to 100 for single-fuel units.

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitScheduleBidHourly party="">
      <!--1 to 100 repetitions:-->
      <mes:ScheduleBidHourly day="" ID="">
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="" delete="">
            <mes:PriceCurve>
              <!--1 to 10 repetitions:-->
              <mes:PricePoint price="" MW="">
                <!--0 to 4 repetitions:-->
                <mes:FuelIndex>
                  <mes:Name?</mes:Name>
                  <mes:Percentage?</mes:Percentage>
                </mes:FuelIndex>
              </mes:PricePoint>
            </mes:PriceCurve>
            <mes:UseStartupNoLoad?</mes:UseStartupNoLoad>
            <mes:UseOfferSlope?</mes:UseOfferSlope>
            <mes:HotStartupCostAndFuel>
              <mes:Cost?</mes:Cost>
              <!--0 to 4 repetitions:-->
              <mes:FuelIndex>
                <mes:Name?</mes:Name>
                <mes:Percentage?</mes:Percentage>
              </mes:FuelIndex>
            </mes:HotStartupCostAndFuel>
            <mes:InterStartupCostAndFuel>
              <mes:Cost?</mes:Cost>
              <!--0 to 4 repetitions:-->
              <mes:FuelIndex>
                <mes:Name?</mes:Name>
                <mes:Percentage?</mes:Percentage>
              </mes:FuelIndex>
            </mes:InterStartupCostAndFuel>
            <mes:ColdStartupCostAndFuel>
              <mes:Cost?</mes:Cost>
              <!--0 to 4 repetitions:-->
              <mes:FuelIndex>
                <mes:Name?</mes:Name>
                <mes:Percentage?</mes:Percentage>
              </mes:FuelIndex>
            </mes:ColdStartupCostAndFuel>
            <mes:NoLoadCostAndFuel>
              <mes:Cost?</mes:Cost>
              <!--0 to 4 repetitions:-->
              <mes:FuelIndex>
                <mes:Name?</mes:Name>
                <mes:Percentage?</mes:Percentage>
              </mes:FuelIndex>
            </mes:NoLoadCostAndFuel>
          </mes:HourlyValues>
        </mes:HourlyProfile>
      </mes:ScheduleBidHourly>
    </mes:SubmitScheduleBidHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        </mes:HourlyValues>
    </mes:HourlyProfile>
    <!--Optional:-->
    <mes:ShortName?></mes:ShortName>
</mes:ScheduleBidHourly>
</mes:SubmitScheduleBidHourly>
</soapenv:Body>
</soapenv:Envelope>

```

Sample of Submittal

```

<SubmitScheduleBidHourly party="ISO New England Inc">
  <ScheduleBidHourly day="2013-10-26" ID="32105">
    <HourlyProfile>
      <HourlyValues time="2013-10-26T02:00:00-04:00">
        <PriceCurve>
          <PricePoint price="10" MW="33">
            <FuelIndex>
              <Name>Algonquin</Name>
              <Percentage>33</Percentage>
            </FuelIndex>
            <FuelIndex>
              <Name>ULSDiesel</Name>
              <Percentage>67</Percentage>
            </FuelIndex>
          </PricePoint>
        </PriceCurve>
        <UseStartupNoLoad>true</UseStartupNoLoad>
        <UseOfferSlope>true</UseOfferSlope>
        <HotStartupCostAndFuel>
          <Cost>23</Cost>
          <FuelIndex>
            <Name>Algonquin</Name>
            <Percentage>75</Percentage>
          </FuelIndex>
          <FuelIndex>
            <Name>ULSDiesel</Name>
            <Percentage>25</Percentage>
          </FuelIndex>
        </HotStartupCostAndFuel>
        <InterStartupCostAndFuel>
          <Cost>34</Cost>
          <FuelIndex>
            <Name>Algonquin</Name>
            <Percentage>30</Percentage>
          </FuelIndex>
          <FuelIndex>
            <Name>ULSDiesel</Name>
            <Percentage>70</Percentage>
          </FuelIndex>
        </InterStartupCostAndFuel>
        <ColdStartupCostAndFuel>
          <Cost>56</Cost>
          <FuelIndex>
            <Name>Algonquin</Name>
            <Percentage>15</Percentage>
          </FuelIndex>
        </ColdStartupCostAndFuel>
      </HourlyValues>
    </HourlyProfile>
  </ScheduleBidHourly>
</SubmitScheduleBidHourly>

```

```

    <FuelIndex>
      <Name>ULSDiesel</Name>
      <Percentage>85</Percentage>
    </FuelIndex>
  </ColdStartupCostAndFuel>
  <NoLoadCostAndFuel>
    <Cost>90</Cost>
    <FuelIndex>
      <Name>Algonquin</Name>
      <Percentage>11</Percentage>
    </FuelIndex>
    <FuelIndex>
      <Name>ULSDiesel</Name>
      <Percentage>89</Percentage>
    </FuelIndex>
  </NoLoadCostAndFuel>
</HourlyValues>
</HourlyProfile>
<ShortName>ShortName</ShortName>
</ScheduleBidHourly>
</SubmitScheduleBidHourly>

```

Sample of Deleting Hourly Data

```

<SubmitScheduleBidHourly party="ISO New England Inc">
  <ScheduleBidHourly day="2013-10-26" ID="32105">
    <HourlyProfile>
      <HourlyValues time="2013-10-26T02:00:00-04:00"
delete="true"/>
    </HourlyProfile>
  </ScheduleBidHourly>
</SubmitScheduleBidHourly>

```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.5.1.3 Additional Value Restrictions

- $MW \geq 0$
- Price of Incremental Energy Offers must be less than or equal to the Energy Offer Cap price and greater than or equal to the Energy Offer Floor price; i.e.
 $\text{Energy Offer Floor price} \leq \text{Price} \leq \text{Energy Offer Cap price}$ ¹⁶.
- Complete sets of Price/MW offer pairs with fuel blend data must be submitted; they cannot be submitted separately and independently. A maximum of ten Price/MW offer pairs per schedule can be submitted.
- Offers must be monotonically increasing (price and MW); i.e. the price for a given MW upper bound must be greater than or equal to the price for a lower MW upper bound. Note that Price/MW bid pairs don't need to be submitted in any particular order. However the MW values of

¹⁶ Energy Offer Floor price and Energy Offer Cap price are both defined in Market Rule 1, section I.2.2.

each pair must be different from the others such that the MW value for a pair must be greater than the MW of the previous pair after the Price/MW pairs are ordered in increasing MW value. The monotonically increasing rule is enforced after the complete set of pairs are received, ordered and processed.

- $0 \leq \text{HotStartupCost} \leq \text{InterStartupCost} \leq \text{ColdStartupCost}$
- $0 \leq \text{NoLoadCost}$
- Fuel Blend percentages must be submitted for every fuel that a unit is configured to use. The list of fuel indices that a unit is configured to use can be obtained using the GetUnit query message described in §7.9.2
- Fuel Blend percentages must add up to 100%. Fuel Blend percentages (fuel name and percentage) must be submitted for all financial offers: Price/MW pairs, Startup Costs and NoLoad Cost.
- For single-fuel units, the fuel blend percentages are optional (as indicated by a minOccurs=0 for the FuelIndex element in eMktMessages.xsd) because they default to 100; if a name is provided however, a percentage value of 100 is also required.
- Cost-based schedules cannot be modified
- For hourly updates submitted during the RT Intraday Re-Offer period, the schedule ID must be the ID of the schedule to be used in the RT market for the market day and offers for only future hours will be accepted. Hourly updates for the next hour must be received by eMarket at least 30 minutes before the start of the hour, otherwise the request will be rejected.

7.5.1.4 Checking of Offer Parameter Values against Market Monitoring Thresholds

An error message will be returned and the submit data request rejected if any of the following offer parameters exceed their threshold value established by the ISO-NE Market Monitor¹⁷:

- Cold, Intermediate and Hot Startup Costs
- No Load Cost

For multi-fuel units, the thresholds are dynamically calculated using the unit's reference levels and the fuel blend information submitted with the cost offers. See §3.3 Mitigation Threshold Checking for other information on validation of offer parameters.

¹⁷ Refer to Market Rule 1, Appendix A, Section III.A.5.5. Threshold values are different for supply offer values submitted while the DA market and the Re-Offer periods are open versus those in effect during the RT Intraday Re-Offer period. As outlined in the Market Rule, you cannot increase your offer of Startup and No Load costs during the RT Intraday Re-Offer period without either a corresponding change to your fuel blend or submitting a Fuel Price Adjustment request. The allowable change in the Startup and No Load supply offers during the RT Intraday Re-Offer period is referenced in Market Rule 1, Appendix A, Section III.A.6.2.

7.5.2 Query Message

7.5.2.1 Purpose of Message

The purpose of this message is to get the schedule hourly override values for a given market day, for a list of market hours and for a list of units. The data returned includes the market day, the price/MW offer blocks, and the ID and short name of the schedules.

7.5.2.2 Mandatory and Optional Fields

To get or query a schedule offer hourly, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	ID	Long	The id of the schedule

- Data for only one market date at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A limited number of schedule IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all be associated with units that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If a schedule ID is not specified then the query returns data for every schedule of every unit that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleBidHourly party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetScheduleBidHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetScheduleBidHourly>
  <QueryFilters>
    <Day>2013-08-07</Day>
    <Hour>2013-08-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetScheduleBidHourly>
```

7.5.2.3 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in increasing MW value order.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the unit

Response Elements

```
<GetScheduleBidHourlyResponse>
  ...
  <ShortName>SchdNme</ShortName>
  ...
</GetScheduleBidHourlyResponse>
```

7.6 Schedule Detail

See section 4.5 for a definition of a schedule

7.6.1 Submit Message

7.6.1.1 Purpose of Message

The purpose of this message is to submit all non-energy offer data under a schedule for a unit. This message contains schedule data relevant to the energy dispatch of a unit, such as the market day, the schedule ID, the various dispatch limits, the various startup and notification times, unit daily parameters, etc.

7.6.1.2 Mandatory and Optional Fields

To create or modify a unit's schedule details, you need to provide the following. The optional elements are read-only elements. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	ColdNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time needed for unit to engage in cold startup
No	No	ColdStartupTime	ElapsedTimeType; Decimal 5.2	The time for a unit to startup from a cold state
No	Yes	CondenseAvailable	Boolean; true (1), false (0)	Whether the unit is available to run as a synchronous condenser
No	Yes	CondenseHourlyCost	CostType; Decimal 10.2	The hourly cost of running the unit as synchronous condenser
No	Yes	CondenseNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time needed to start a unit as synchronous condenser
No	Yes	CondensePower	MWType; Decimal 6.1	The unit's MW consumption when running as a condenser
No	Yes	CondenseStartupCost	CostType; Decimal 10.2	The cost of starting up unit as synchronous condenser
No	No	EcoMax	MWType; Decimal 6.1	The economic maximum MW dispatch limit
No	No	EcoMin	MWType; Decimal 6.1	The economic minimum MW dispatch limit
No	No	EmerMin	MWType; Decimal 6.1	The emergency minimum MW dispatch limit
No	No	HotNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time needed for a unit to engage in hot startup

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	HotStartupTime	ElapsedTimeType; Decimal 5.2	The time for a unit to startup from a hot state
No	No	HotToColdTime	ElapsedTimeType; Decimal 5.2	The time for the unit to transition from hot to cold state
No	No	HotToInterTime	ElapsedTimeType; Decimal 5.2	The time for the unit to transition from hot to intermediate state
No	No	InterNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time needed for a unit to engage in intermediate startup
No	No	InterStartupTime	ElapsedTimeType; Decimal 5.2	The time for a unit to startup from an intermediate state
No	No	MarketType	Enumerated String; Both, DayAhead, RealTime	The energy markets in which this schedule's offer parameters are to be used.
No	Yes	MaxDailyEnergy	MWType; Decimal 6.1	The maximum amount of energy (MWhr) that the unit can produce in one market day.
No	Yes	MaxDailyStarts	Long	The maximum number of times that the unit can be started in one market day.
No	No	MinDownTime	ElapsedTimeType; Decimal 5.2	The minimum amount of time in hours that unit must remain off-line when it is shutdown.
No	No	MinRunTime	ElapsedTimeType; Decimal 5.2	The minimum amount of time in hours that the unit must run when it is committed and brought on-line
No	Yes	RTHighOperLimit	MWType; Decimal 6.1	The maximum MW output that the unit can produce when required for reliability purposes.
Yes	No	ScheduleAvailable	Boolean; true (1), false (0)	Whether the schedule is available for use ¹⁸ . Read Only
Yes	No	ShortName	String; 8	The short name of the schedule. Read Only

Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitScheduleDetail message in the eMktMessages.xsd.

¹⁸ As defined using the Submit Schedule Selection message defined in §7.8.1

7.6.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitScheduleDetail party="??">
      <!--1 or more repetitions:-->
      <mes:ScheduleDetail day="??" ID="??">
        <mes:ColdNotifyTime?></mes:ColdNotifyTime>
        <mes:ColdStartupTime?></mes:ColdStartupTime>
        <mes:CondenseAvailable?></mes:CondenseAvailable>
        <mes:CondenseHourlyCost?></mes:CondenseHourlyCost>
        <mes:CondenseNotifyTime?></mes:CondenseNotifyTime>
        <mes:CondensePower?></mes:CondensePower>
        <mes:CondenseStartupCost?></mes:CondenseStartupCost>
        <mes:EcoMax?></mes:EcoMax>
        <mes:EcoMin?></mes:EcoMin>
        <mes:EmerMin?></mes:EmerMin>
        <mes:HotNotifyTime?></mes:HotNotifyTime>
        <mes:HotStartupTime?></mes:HotStartupTime>
        <mes:HotToColdTime?></mes:HotToColdTime>
        <mes:HotToInterTime?></mes:HotToInterTime>
        <mes:InterNotifyTime?></mes:InterNotifyTime>
        <mes:InterStartupTime?></mes:InterStartupTime>
        <mes:MarketType?></mes:MarketType>
        <mes:MaxDailyEnergy?></mes:MaxDailyEnergy>
        <mes:MaxDailyStarts?></mes:MaxDailyStarts>
        <mes:MinDownTime?></mes:MinDownTime>
        <mes:MinRunTime?></mes:MinRunTime>
        <mes:RTHighOperLimit?></mes:RTHighOperLimit>
        <!--Optional:-->
        <mes:ScheduleAvailable?></mes:ScheduleAvailable>
        <!--Optional:-->
        <mes:ShortName?></mes:ShortName>
      </mes:ScheduleDetail>
    </mes:SubmitScheduleDetail>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitScheduleDetail>
  <ScheduleDetail day="2010-06-05" ID="55205">
    <ColdNotifyTime>12</ColdNotifyTime>
    <ColdStartupTime>3</ColdStartupTime>
    <CondenseAvailable>false</CondenseAvailable>
    <CondenseHourlyCost xsi:nil="true"/>
    <CondenseNotifyTime xsi:nil="true"/>
    <CondensePower xsi:nil="true"/>
    <CondenseStartupCost xsi:nil="true"/>
    <EcoMax>18.8</EcoMax>
    <EcoMin>25</EcoMin>
    <EmerMin>10</EmerMin>
    <HotNotifyTime>8</HotNotifyTime>
    <HotStartupTime>2</HotStartupTime>
    <HotToColdTime>72</HotToColdTime>
    <HotToInterTime>36</HotToInterTime>
    <InterNotifyTime>10</InterNotifyTime>
    <InterStartupTime>2</InterStartupTime>
    <mes:MarketType>Both</mes:MarketType>
    <MaxDailyEnergy xsi:nil="true"/>
    <MaxDailyStarts>7</MaxDailyStarts>
    <MinDownTime>12.3</MinDownTime>
    <MinRunTime>16</MinRunTime>
    <RTHighOperLimit>100.00</RTHighOperLimit>
  </ScheduleDetail>
</SubmitScheduleDetail>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.6.1.4 Additional Value Restrictions

- EmerMin > 0 if EcoMax > 0
EcoMin > 0 if EcoMax > 0;
i.e. the Emergency and Economic Minimum limits must be greater than zero if the Economic Maximum limit is greater than zero
- EcoMax ≥ CSO if CSO > 0; i.e. for a resource with a Capacity Supply Obligation, its Economic Maximum Limit **should** be at least as high as its Capacity Supply Obligation. A warning will be returned if the rule is violated; it's only a warning because the resource may not be fully available.
- $0 \leq \text{EmerMin} \leq \text{EcoMin} \leq \text{EcoMax} \leq \text{RTHighOperLimit}$
This is the **limit nesting rule**. The data submitted is rejected in its entirety if this rule is violated.
If the schedule is flagged as available at the time the values are submitted and the unit is a regulating resource, the limit nesting rule is extended to the regulation limits (i.e. $\text{EcoMin} \leq \text{Regulation Low} < \text{Regulation High} \leq \text{EcoMax}$) where the regulation limits are the more constraining of the limits among the daily default (those submitted

using the Regulation Offer message) and the hourly updates (those submitted using the Regulation Hourly message) for the selected market day. A warning message is returned when the limit nesting rule with regulation limits is violated.

- $\text{HotToColdTime} \geq \text{HotToInterTime} \geq 0$
- $\text{ColdStartupTime} \geq \text{InterStartupTime} \geq \text{HotStartupTime} \geq 0$
- $\text{ColdNotifyTime} \geq \text{InterNotifyTime} \geq \text{HotNotifyTime} \geq 0$
- $\text{MinDownTime} \geq 0$
- $24 \geq \text{MinRunTime} \geq 0$
- MaxDailyEnergy (if defined) **should** be greater than or equal to $\text{EcoMin} \times \text{MinRunTime}$; a warning will be returned if it is not.
- When a unit self schedules for a number of hours (not necessarily consecutive and taking into consideration the default availability/operating status of the unit), the MaxDailyEnergy (if defined) **should** be greater than or equal to $\text{EcoMin} \times$ the total number of self scheduled hours; a warning will be returned if it is not.
- For Claim 10 capable Units, a warning will be returned when the sum of cold notification and cold startup time exceeds the Claim 10 criteria, i.e. $\geq 10\text{min}$.
- For Claim 30 capable Units, a warning will be returned when the sum of cold notification and cold startup time exceeds the Claim 30 criteria, i.e. $\geq 30\text{min}$.
- For Claim 10 capable Units and Claim 30 capable Units, a warning will be returned when $\text{MinRunTime} > 1$ or $\text{MinDownTime} > 1$ and values have been submitted at the unit level for Claim 10 or Claim 30 because time values greater than 1 hour makes the unit ineligible for off-line reserve capability.
- $\text{Nil}=\text{true}$ for $\text{CondenseHourlyCost}$ sets the value in the database to NULL meaning that it is not defined, i.e. not applicable.
- $\text{Nil}=\text{true}$ for $\text{CondenseNotifyTime}$ sets the value in the database to NULL meaning that it is not defined, i.e. not applicable.
- $\text{Nil}=\text{true}$ for CondensePower sets the value in the database to NULL meaning that it is not defined, i.e. not applicable.
- $\text{Nil}=\text{true}$ for $\text{CondenseStartupCost}$ sets the value in the database to NULL meaning that it is not defined, i.e. not applicable.
- $\text{Nil}=\text{true}$ for MaxDailyEnergy sets the value in the database to NULL meaning that it is not defined, i.e. not applicable.

7.6.1.5 Checking of Offer Parameter Values against Market Monitoring Thresholds

- An error message will be returned and the submit data request rejected if any of the following offer parameters exceed their threshold value established by the ISO-NE Market Monitor¹⁹:
 - ColdNotifyTime and ColdStartupTime
 - HotNotifyTime and HotStartupTime

¹⁹ Refer to Market Rule 1, Appendix A, Section III.A.6.1.

- InterNotifyTime and InterStartupTime
- HotToColdTime and HotToInterTime
- MinDownTime and MinRunTime.
- An error message will be returned and the submit data request rejected if any of the following conditions are true:
 - The sum of HotNotifyTime, HotStartupTime, MinDownTime, and MinRunTime values is greater than the sum of their reference values by more than a threshold value²⁰ established by the ISO-NE Market Monitor.
 - The sum of ColdNotifyTime, ColdStartupTime, MinDownTime, and MinRunTime values is greater than the sum of their reference values by more than a threshold value²⁰ established by the ISO-NE Market Monitor.
 - The sum of InterNotifyTime, InterStartupTime, MinDownTime, and MinRunTime values is greater than the sum of their reference values by more than a threshold value²⁰ established by the ISO-NE Market Monitor.
- A warning message will be returned if any of the following offer parameter values violate their threshold value established by the ISO-NE Market Monitor¹⁹: EcoMax, EcoMin, EmerMin, RTHighOperLimit, MaxDailyEnergy, and MaxDailyStarts.

7.6.2 Query Message

7.6.2.1 Purpose of Message

The purpose of this message is to get all non-energy data for a unit schedule for a specified day. The response will contain the data that was submitted with the corresponding Submit Schedule Detail message. Data returned includes the market day, the dispatch limits, the markets to which the bids apply, the various startup and notification times, etc.

7.6.2.2 Mandatory and Optional Fields

To get or query a unit schedule detail, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the schedule

- Data for a single market day at a time can be queried.
- A limited number of schedule IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all be associated with units that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

²⁰ Refer to Market Rule 1, Appendix A, Section III.A.6.1. Currently defined as 6 hours (August 2014)

- If a schedule ID is not specified then the query returns data for every schedule of every unit that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

7.6.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleDetail party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetScheduleDetail>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetScheduleDetail>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetScheduleDetail>
```

7.6.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	No	ScheduleAvailable	Boolean; true (1), false (0)	Whether the schedule is available for use.
No	No	ShortName	String; 8	The short name of the schedule ²¹ .

²¹ As defined using the Submit Schedule message defined in §7.3.1

Response Elements

```
<GetScheduleDetailResponse>  
  ...  
  <ScheduleAvailable>true</ScheduleAvailable>  
  <ShortName>Shrtname</ShortName>  
  ...  
</GetScheduleDetailResponse>
```

7.7 Schedule Times Hourly

See section 4.5 for a definition of a schedule

7.7.1 Submit Message

7.7.1.1 Purpose of Message

This message allows hourly override for the startup times and notification times defined at the daily level for a schedule. The message can also be used to delete an hourly update by simply setting the “delete” attribute to TRUE on the message element for the hour to delete.

7.7.1.2 Mandatory and Optional Fields

To create or modify a unit’s hourly schedule startup and notification times, you need to provide the following. The optional elements are read-only elements. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	time	DateTime; yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	delete	Boolean; false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
No	No	HotStartupTime	ElapsedTimeType; Decimal 5.2	The time in hours for a unit to startup from a hot state
No	No	InterStartupTime	ElapsedTimeType; Decimal 5.2	The time in hours for a unit to startup from an intermediate state
No	No	ColdStartupTime	ElapsedTimeType; Decimal 5.2	The time in hours for a unit to startup from a cold state
No	No	HotNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time in hours needed for a unit to engage in hot startup
No	No	InterNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time in hours needed for a unit to engage in intermediate
No	No	ColdNotifyTime	ElapsedTimeType; Decimal 5.2	The notification time in hours needed for unit to engage in cold startup
Yes	No	ShortName	String; 8	The short name of the unit. Read Only

Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitScheduleTimesHourly message in the eMktMessages.xsd.

7.7.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitScheduleTimesHourly party="?">
      <!--1 to 100 repetitions:-->
      <mes:ScheduleTimesHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <mes:HotStartupTime?</mes:HotStartupTime>
            <mes:InterStartupTime?</mes:InterStartupTime>
            <mes:ColdStartupTime?</mes:ColdStartupTime>
            <mes:HotNotifyTime?</mes:HotNotifyTime>
            <mes:InterNotifyTime?</mes:InterNotifyTime>
            <mes:ColdNotifyTime?</mes:ColdNotifyTime>
          </mes:HourlyValues>
        </mes:HourlyProfile>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:ScheduleTimesHourly>
    </mes:SubmitScheduleTimesHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitScheduleTimesHourly party="P1">
  <ScheduleTimesHourly day="2013-08-07">
    <Resource name="Unit1" resourceID="5678"/>
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00">
        <HotStartupTime>2</HotStartupTime >
        <InterStartupTime>2</InterStartupTime>
        <ColdStartupTime>5</ColderStartupTime>
        <HotNotifyTime Time>12</HotNotifyTimeTime>
        <InterNotifyTimeTime>14</InterNotifyTimeTime>
        <ColdNotifyTimeTime>16</ColdNotifyTimeTime>
      </HourlyValues>
    </HourlyProfile>
  </ScheduleTimesHourly>
</SubmitScheduleTimesHourly>
```

Sample of Deleting Hourly Data

```
<SubmitScheduleTimesHourly party="P1">
  <ScheduleTimesHourly day="2013-08-07">
    <Resource name="Unit1" resourceID="5678"/>
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00"
        delete="true"/>
    </HourlyProfile>
  </ScheduleTimesHourly>
</SubmitScheduleTimesHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.7.1.4 Additional Value Restrictions

- ColdStartupTime \geq InterStartupTime \geq HotStartupTime \geq 0
- ColdNotifyTime \geq InterNotifyTime \geq HotNotifyTime \geq 0

7.7.1.5 Checking of Offer Parameter Values against Market Monitoring Thresholds

An error message will be returned and the submit data request rejected if any of the following offer parameters for an hour exceed their threshold value established by the ISO-NE Market Monitor²²:

- ColdNotifyTime and ColdStartupTime
- HotNotifyTime and HotStartupTime
- InterNotifyTime and InterStartupTime

An error message will be returned and the submit data request rejected if any of the following conditions are true:

- The sum of HotNotifyTime, HotStartupTime, MinDownTime, and MinRunTime values is greater than the sum of their reference values by more than a threshold value²³ established by the ISO-NE Market Monitor.
- The sum of ColdNotifyTime, ColdStartupTime, MinDownTime, and MinRunTime values is greater than the sum of their reference values by more than a threshold value²³ established by the ISO-NE Market Monitor.
- The sum of InterNotifyTime, InterStartupTime, MinDownTime, and MinRunTime values is greater than the sum of their reference values by more than a threshold value²³ established by the ISO-NE Market Monitor.

²² Refer to Market Rule 1, Appendix A, Section III.A.6.1.

²³ Refer to Market Rule 1, Appendix A, Section III.A.6.1. Currently defined as 6 hours (August 2014)

7.7.2 Query Message

7.7.2.1 Purpose of Message

The purpose of this message is to get hourly startup and notification time data for a unit schedule for a specified day.

7.7.2.2 Mandatory and Optional Fields

To get or query a schedule times hourly, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	ID	Long	The id of the Unit

- Data for a single market day at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A limited number of schedule IDs can be specified; see the maxOccurs attribute of the QueryByDayIDAndHour type in the eMktMessages.xsd. They must all be associated with units that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the schedule ID is not specified then the query returns data for every schedule of every unit that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleTimesHourly party="??">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetScheduleTimesHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetScheduleTimesHourly>
  <QueryFilters>
    <Day>2013-08-07</Day>
    <Hour>2013-08-07T16:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetScheduleTimesHourly>
```

7.7.2.3 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the unit

Response Elements

```
<GetScheduleTimesHourlyResponse>
  <ScheduleTimesHourly day="2013-08-07" ID="123456">
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00">
        <HotStartupTime>2</HotStartupTime >
        <InterStartupTime>2</InterStartupTime>
        <ColdStartupTime>5</ColderStartupTime>
        <HotNotifyTime Time>12</HotNotifyTimeTime>
        <InterNotifyTimeTime>23</InterNotifyTimeTime>
        <ColdNotifyTimeTime>11</ColdNotifyTimeTime>
      </HourlyValues>
      ...
    </HourlyProfile>
    <ShortName>ShortName</ShortName>
  </ScheduleTimesHourly>
</GetScheduleTimesHourlyResponse>
```

7.8 Unit Schedule Selection

See section 4.5 for a definition of a schedule.

7.8.1 Submit Message

7.8.1.1 Purpose of Message

The purpose of this message is to set the availability of a schedule for a specific market day. This message contains data relevant to setting the availability of a schedule, such as the market day, the schedule ID and its availability.

7.8.1.2 Mandatory and Optional Fields

To set or change the availability of a schedule, you need to provide the following. The optional element is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the schedule
No	No	Available	Boolean; true (1), false (0)	Whether the schedule is available
Yes	No	ShortName	String; 8	The short name of the schedule. Read Only

Data for a limited number of schedule IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitScheduleSelection message in the eMktMessages.xsd.

7.8.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitScheduleSelection party="?">
      <!--1 or more repetitions:-->
      <mes:ScheduleSelection day="?">
        <!--Zero or more repetitions:-->
        <mes:Schedule ID="?">
          <mes:Available?</mes:Available>
          <!--Optional:-->
          <mes:ShortName?</mes:ShortName>
        </mes:Schedule>
      </mes:ScheduleSelection>
    </mes:SubmitScheduleSelection>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitScheduleSelection>  
  <ScheduleSelection day="2010-07-07">  
    <Schedule ID="1234567890">  
      <Available>true</Available>  
    </Schedule>  
  </ScheduleSelection>  
</SubmitScheduleSelection>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.8.1.4 Additional Value Restrictions

- Only one price-based schedule per market (Day-Ahead [DA], Real Time [RT]) can be marked as available for a given market day. Therefore, when changing the availability of schedules, when one is set available for a particular market then the one that was previously available for that market must be set unavailable within the same message, and vice versa. Note that if you change the availability of a schedule that is marked as applicable to both DA and RT markets, you may have to change the availability of two other schedules in the same message, one applicable to the DA market only and another applicable to the RT market only.
- When a schedule for a unit that is a regulating resource is made available, irrespective of whether it is a schedule designated for use in the DA market or the RT market or both, the regulation limit nesting rule is checked;
i.e. $EcoMin \leq Regulation\ Low < Regulation\ High \leq EcoMax$
where the economic and regulation limits are the more constraining of the limits among the daily default (those submitted using the Schedule Detail message and the Regulation Offers message) and the hourly updates (those submitted using the Unit Hourly message and the Regulation Hourly Updates message) for the selected market day. A warning message is returned when this rule is violated.
- The availability of cost-based schedules cannot be changed using this message. A request to do so will be rejected and an error returned.

7.8.2 Query Message

7.8.2.1 Purpose of Message

The purpose of this message is to get the availability of all the unit schedules for a given market day. The data returned includes the market day, the ID of the schedule, its short name and its availability.

7.8.2.2 Mandatory and Optional Fields

To get or query the unit schedule availabilities, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

- Only one market date at a time can be queried.
- The query returns data for every schedule of every unit that belongs to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

7.8.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetScheduleSelection party="?">
      <mes:Day?</mes:Day>
    </mes:GetScheduleSelection>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetScheduleSelection>
  <Day>2010-07-07</Day>
</GetScheduleSelection>
```

7.8.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the schedule. Read Only

Response Elements

```
<GetScheduleSelectionResponse>
  ...
  <ShortName>Shrtname</ShortName>
  ...
</GetScheduleSelectionResponse>
```

7.9 Unit

A generating unit is a facility that produces electric energy and contributes to system capacity.

7.9.1 Submit Message

7.9.1.1 Purpose of Message

The purpose of this message is to submit information about units. This message contains data relevant to units, such as the day, unit ID, the unit's off-line 10-minute and 30-minute reserve capability (Claim10 and Claim30 respectively) when eligible, the unit's default ramp rate, and its default status (Economic – Available for economic commitment and dispatch, MustRun – Self-scheduled, Unavailable).

7.9.1.2 Mandatory and Optional Fields

To modify a unit's default daily parameters, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the unit
No	Yes	Claim10	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 10 minutes
No	Yes	Claim30	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 30 minutes
No	No	DefaultStatus	Enumerated String; Economic, MustRun, Unavailable	The default availability/operating status of a unit
No	No	RampRateDefault	MWType; Decimal 6.1	The default ramp rate to use for the asset if an hourly value does not exist

The nillable claim capability elements above should be given values if the asset is registered as Claim10 or Claim30 capable, i.e. the values submitted should not be nil="true"; otherwise a nil value (nil="true") should be submitted. Non-nil submitted values will be ignored in the market system if the asset does not have the corresponding registered capability.

A limited number of unit IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitUnit message in the eMktMessages.xsd.

7.9.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitUnit party="">
      <!--1 or more repetitions:-->
      <mes:Unit day="" ID="">
        <mes:Claim10?</mes:Claim10>
        <mes:Claim30?</mes:Claim30>
        <mes:DefaultStatus?</mes:DefaultStatus>
        <mes:RampRateDefault?</mes:RampRateDefault>
      </mes:Unit>
    </mes:SubmitUnit>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitUnit>
  <Unit day="2010-07-07" ID="123456">
    <Claim10>0</Claim10>
    <Claim30>0</Claim30>
    <DefaultStatus>Economic</DefaultStatus>
    <RampRateDefault>10</RampRateDefault>
  </Unit>
</SubmitUnit>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.9.1.4 Additional Value Restrictions

- $0 \leq \text{RampRateDefault}$
- A warning will be issued if the RampRateDefault exceeds the RampRateCap when the demonstrated capability that appears in the GetUnitResponse message is defined (not NULL)
- Nil=true for Claim10 sets the value in the database to NULL
- Nil=true for Claim30 sets the value in the database to NULL
- Claim10 value must be greater than or equal to zero when a non “Nil” is submitted
- Claim30 value must be greater than or equal to zero when a non “Nil” is submitted
- A warning will be returned if the unit is Claim10 capable and any of the following conditions exist
 - Claim10 = 0 or NULL

- Claim10 > Claim10Cap when the demonstrated capability that appears in the GetUnitResponse message is defined (not NULL)
- Claim10 > Claim30
- The sum of the unit’s Cold Notification and Cold Startup times exceeds the Claim10 criteria, i.e. $\geq 10\text{min}$. The notification and startup times shall be taken from the schedule designated as available for DA and/or RT.
- Either of the unit’s MinRun or MinDown time > 1
- A warning will be returned if the unit is Claim30 capable and any of the following conditions exist
 - Claim30 = 0 or NULL
 - Claim30 > Claim30Cap when the demonstrated capability that appears in the GetUnitResponse message is defined (not NULL)
 - The sum of the unit’s Cold Notification and Cold Startup times exceeds Claim30 criteria, i.e. $\geq 30\text{min}$. The notification and startup times shall be taken from the schedule designated as available for DA and/or RT.
- For Claim 10 capable Units and Claim 30 capable Units, a warning will be returned when submitting either a Claim10 or a Claim30 value while MinRun time > 1 or MinDown time > 1 because time values greater than 1 hour makes the unit ineligible for off-line reserve capability. The MinRun and MinDown times shall be taken from the schedule designated as available for DA and/or RT.

7.9.2 Query Message

7.9.2.1 Purpose of Message

The purpose of this message is to get all recorded details for a specified unit. This response will contain data about the unit, some of which was entered with the corresponding Submit message. Data returned includes the type of unit, unit name/number, day, unit ID, cold startup cost, default status, etc. In particular, it includes the list of Fuel Index names that the unit is configured to use and that must be used when submitting fuel blend percentages with financial offers (Startup and NoLoad costs, Incremental Energy Offers [Price/MW pairs]) and Fuel Price Adjustment requests for fossil fuels.

7.9.2.2 Mandatory and Optional Fields

To get or query a unit, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The specified market day
Yes	No	ID	Long	The ID of the unit requested

- Data for only one market date at a time can be queried.

- A limited number of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the unit ID element is not specified then the query returns data for all the units that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

7.9.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetUnit party="">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetUnit>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetUnit>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetUnit>
```

7.9.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The optional elements and attributes below have values if a unit is Claim10 or Claim30 capable.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	CapacityResource	Boolean; true (1), false (0)	Whether the unit is a capacity resource
Yes	Yes	Claim10	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 10 minutes
Yes	Yes	Claim10Cap	MWType; Decimal 6.1	The Claim10 capability of the unit from the most recent audit
No	No	Claim10Capable	Boolean; true (1), false (0)	Whether the unit is Claim10 capable

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	Yes	Claim30	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 30 minutes
Yes	Yes	Claim30Cap	MWType; Decimal 6.1	The Claim30 capability of the unit from the most recent audit
No	No	Claim30Capable	Boolean; true (1), false (0)	Whether the unit is Claim30 capable
No	No	DefaultStatus	Enumerated String; Economic, MustRun, Unavailable	The default availability/operating status of a unit
No	No	FuelIndex	String; 40	Name of a fuel index. A unit can be configured to use up to four different fuels. The query returns the list of fuel indices that the unit is configured to use.
No	No	LeadParticipantID	Long	The lead Participant ID
No	No	LeadParticipantName	String; 40	The name of the Lead Participant that is responsible for submitting offers for the unit.
No	No	NodeID	Long	The numeric identifier of the Pnode associated with the unit
No	No	NodeName	String; 30	Name of the Pnode associated with the unit
No	Yes	RampRateCap	MWType; Decimal 6.1	The ramp rate capability of the unit from the most recent audit.
No	No	RampRateDefault	MWType; Decimal 6.1	The default ramp rate to use for the unit if a ramp rate curve or an hourly value do not exist
No	No	RegResource	Boolean; true (1), false (0)	Whether the unit is a regulation resource
No	No	TypeOfUnit	String; 40	The type of unit
No	No	TypeOfUnitLongName	String; 40	The long name of the type of unit
No	No	UnitLongName	String; 40	The unit long name
No	No	UnitNumber	Long	The unit number
No	No	UnitShortName	String; 8	The unit short name

Response Elements

```
<GetUnitResponse>
  <Unit day="2014-07-07" ID="12345678">
    <CapacityResource>true</CapacityResource>
    <Claim10Settings Claim10Capable="true">
      <Claim10>18</Claim10>
      <Claim10Cap>18</Claim10Cap>
    </Claim10Settings>
    <Claim30Settings Claim30Capable="true">
      <Claim30>27</Claim30>
      <Claim30Cap>27</Claim30Cap>
    </Claim30Settings>
    <DefaultStatus>Economic</DefaultStatus>
    <LeadParticipantID>12345</LeadParticipantID>
    <LeadParticipantName>MW PUD</LeadParticipantName>
    <NodeID>14250</NodeID>
    <NodeName>UN.STATION 12.8STM3</NodeName>
    <RampRateCap>25</RampRateCap>
    <RampRateDefault>25</RampRateDefault>
    <RegResource>true</RegResource>
    <FuelIndexList>
      <FuelIndex>No2</FuelIndex>
      <FuelIndex>ULSDiesel</FuelIndex>
    </FuelIndexList>
    <TypeOfUnit>steam</TypeOfUnit>
    <TypeOfUnitLongName>steam</TypeOfUnitLongName>
    <UnitLongName>MW Steam 3B</UnitLongName>
    <UnitNumber>12345</UnitNumber>
    <UnitShortName>STM3</UnitShortName>
  </Unit>
  ...
</GetUnitResponse>
```

7.10 Unit Hourly

See section 7.9 for a definition of a Unit.

7.10.1 Submit Message

7.10.1.1 Purpose of Message

The purpose of this message is to add or update details of a unit's potential generation for a specific hour of a market day. This message is used to submit data relevant to a unit's generation capabilities during a specified hour of a day, specifically the unit's

- Dispatch availability (status),
- Economic min/max generation limits,
- Emergency minimum generation limit,
- Real time high operating limit, which defines its availability and capability for emergency reliability dispatch purposes,
- Off-line 10-minute (Claim10) and 30-minute (Claim30) reserve capabilities values when the unit is defined to be capable of either.

The message can also be used to delete an hourly update by simply setting the "delete" attribute to TRUE on the message element for the hour to delete.

Note that the Unit Hourly messages override all schedule and default values for the day and hour specified.

7.10.1.2 Mandatory and Optional Fields

To create or modify an hourly override entry for a unit, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the unit
No	No	time	DateTime; yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
Yes	No	delete	Boolean; true(1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
Yes ²⁴	No	Claim10	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 10 minutes

²⁴ Claim10 and Claim30 are optional only if the unit is not Claim10 or Claim30 capable.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes ²⁴	No	Claim30	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 30 minutes
No	No	EmerMin	MWType; Decimal 6.1	The emergency minimum generation limit of the unit
No	No	EcoMax	MWType; Decimal 6.1	The economic maximum generation limit of the unit
No	No	EcoMin	MWType; Decimal 6.1	The economic minimum generation limit of the unit
No	No	RTHighOperLimit	MWType; Decimal 6.1	The real time high operating limit of the unit
No	No	ResourceStatus	Enumerated String; Economic, MustRun, Unavailable	The operation status of the unit
Yes	No	ShortName	String; 8	The short name of the unit. Read Only

A limited number of unit IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitUnitHourly message in the eMktMessages.xsd.

7.10.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitUnitHourly party="?">
      <!--1 or more repetitions:-->
      <mes:UnitHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--0 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <!--Optional:-->
            <mes:Claim10?></mes:Claim10>
            <!--Optional:-->
            <mes:Claim30?></mes:Claim30>
            <mes:EmerMin?></mes:EmerMin>
            <mes:EcoMax?></mes:EcoMax>
            <mes:EcoMin?></mes:EcoMin>
            <mes:RTHighOperLimit?></mes:RTHighOperLimit>
            <mes:ResourceStatus?></mes:ResourceStatus>
          </mes:HourlyValues>
        </mes:HourlyProfile>
      </mes:UnitHourly>
    </mes:SubmitUnitHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitUnitHourly>
  <UnitHourly day="2010-07-07" ID="12345678">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T14:00:00-04:00">
        <Claim10>30.0</Claim10>
        <Claim30>60.0</Claim30>
        <EmerMin>150</EmerMin>
        <EcoMax>232</EcoMax>
        <EcoMin>155</EcoMin>
        <RTHighOperLimit>250</RTHighOperLimit>
        <ResourceStatus>Economic</ResourceStatus>
      </HourlyValues>
    </HourlyProfile>
  </UnitHourly>
</SubmitUnitHourly>
```

Sample of Deleting Hourly Data

```
<SubmitUnitHourly>
  <UnitHourly day="2010-07-07" ID="12345678">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T00:00:00-04:00" delete="true"/>
      <HourlyValues time="2010-07-07T11:00:00-04:00" delete="true"/>
    </HourlyProfile>
  </UnitHourly>
</SubmitUnitHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.10.1.4 Additional Value Restrictions

- $EcoMax \geq CSO$ if $CSO > 0$
i.e. for a resource with a Capacity Supply Obligation, its Economic Maximum Limit **should** be at least as high as its Capacity Supply Obligation. A warning will be returned if the rule is violated; it's only a warning because the resource may not be fully available.
- $0 \leq EmerMin \leq EcoMin \leq EcoMax \leq RTHighOpLimit$. This is the **limit nesting rule**. The data submitted is rejected if this rule is violated. Note that all must be provided otherwise an error will be reported.
- For units capable of providing regulation, the limit nesting rule is extended to include the regulation limits, i.e.
 $EmerMin \leq EcoMin \leq Reg\ Low < Reg\ High \leq EcoMax \leq RTHighOpLimit$
where Reg Low and Reg High are the regulation low limit and the regulation high limit for the corresponding hour as defined using the Regulation Offer message and overridden using the Regulation Hourly

message for the selected market day. A warning message is returned when this limit nesting rule is violated.

- Claim10 value must be greater than or equal to zero. It must be provided if the asset is Claim10 capable.
- Claim30 value must be greater than or equal to zero. It must be provided if the asset is Claim30 capable.
- When a unit self schedules (ResourceStatus is set to MustRun) for a number of hours (not necessarily consecutive), the MaxDailyEnergy (if defined in the applicable DA and RT Schedule Details of the schedules that are flagged as available) **should** be greater than or equal to EcoMin x the total number of self scheduled hours; a warning will be returned if it is not.
- If the ResourceStatus is set to MustRun (self-scheduled) for an hour (through the combination of the flag for the day and the hourly overrides of the flag), it **should** be set for a consecutive number hours sufficient to cover the unit's MinRunTime; if not, a warning will be returned.
- If the ResourceStatus is set to Unavailable for any hour (through the combination of the flag for the day and the hourly overrides of the flag), it **should** be set for a consecutive number of hours sufficient to cover the unit's MinDownTime; if not, a warning will be returned.
- A warning will be returned if the unit is Claim10 capable and any of the following conditions exist
 - Claim10 = 0
 - Claim10 > Claim10Cap when the demonstrated capability that appears in the GetUnitResponse message is defined (not NULL)
 - Claim10 > Claim30
 - The sum of the unit's Cold Notification and Cold Startup times exceeds the Claim10 criteria, i.e. $\geq 10\text{min}$. The notification and startup times shall be taken from the schedule designated as available for DA and/or RT.
- A warning will be returned if the unit is Claim30 capable and any of the following conditions exist
 - Claim30 = 0
 - Claim30 > Claim30Cap when the demonstrated capability that appears in the GetUnitResponse message is defined (not NULL)
 - The sum of the unit's Cold Notification and Cold Startup times exceeds Claim30 criteria, i.e. $\geq 30\text{min}$. The notification and startup times shall be taken from the schedule designated as available for DA and/or RT.
- For Claim 10 capable Units and Claim 30 capable Units, a warning will be returned when submitting either a Claim10 or a Claim30 value while MinRun time > 1 or MinDown time > 1 because time values greater than 1 hour makes the unit ineligible for off-line reserve capability. The MinRun and MinDown times shall be taken from the schedule designated as available for DA and/or RT.

Deleting data for an hour is equivalent to removing the override for that market hour.

7.10.1.5 Checking of Override Values against Market Monitoring Thresholds

A warning message will be returned if any of the following hourly override values exceed the corresponding threshold value established by the ISO-NE Market Monitor²⁵: RTHighOperLimit, EcoMax, EcoMin, and EmerMin. See rules defined in 3.3 Mitigation Threshold Checking.

7.10.2 Query Message

7.10.2.1 Purpose of Message

The purpose of this message is to get the dispatch and capability limits and availability status data for units that override the daily or schedule values for a given hour of a market day. The response will contain data that was entered with a corresponding Submit message or NULL values, meaning that there are no override values. Data returned includes the date and time, the short name and ID of the unit, the economic min/max, emergency minimum and real time high operating limits, the unit's off-line reserve capabilities, and the availability status.

7.10.2.2 Mandatory and Optional Fields

To get or query the hourly overrides on units, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time of the day
Yes	No	ID	Long	The ID of the unit

- Only data for one market date at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A limited number of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the unit ID element is not specified then the query returns data for all the units that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

²⁵ Refer to Market Rule 1, Appendix A, Section III.A.6.1.

7.10.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetUnitHourly party=""?>
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetUnitHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetUnitHourly>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <Hour>2010-07-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetUnitHourly>
```

7.10.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in ascending hour value.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	Claim10	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 10 minutes
No	Yes	Claim30	MWType; Decimal 6.1	The claimed MW generation that can be achieved from an offline state in 30 minutes
No	Yes	EmerMin	MWType; Decimal 6.1	The emergency minimum generation limit of the unit
No	Yes	EcoMax	MWType; Decimal 6.1	The economic maximum generation limit of the unit
No	Yes	EcoMin	MWType; Decimal 6.1	The economic minimum generation limit of the unit
No	Yes	RTHighOperLimit	MWType; Decimal 6.1	The real time high operating limit of the unit
No	Yes	ResourceStatus	Enumerated String; Economic, MustRun, Unavailable	The operation status of the unit

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the Unit

Response Elements when there are Overrides to report

```

<GetUnitHourlyResponse>
  ...
  <HourlyValues time="2011-07-20T00:00:00-04:00">
    <EmerMin>155</EmerMin>
    <EcoMax>232</EcoMax>
    <EcoMin>156</EcoMin>
    <RTHighOperLimit>250</RTHighOperLimit>
    <ResourceStatus>Economic</ResourceStatus>
  </HourlyValues>
  <ShortName>ShrtNme</ShortName>
  ...
</GetUnitHourlyResponse>

```

Response Elements when there are no Overrides to report²⁶

```

< GetUnitHourlyResponse >
  ...
  <HourlyValues time="2011-07-20T00:00:00-04:00">
    <Claim10 xsi:nil="true"/>
    <Claim30 xsi:nil="true"/>
    <EmerMin xsi:nil="true"/>
    <EcoMin xsi:nil="true"/>
    <EcoMax xsi:nil="true"/>
    <RTHighOperLimit xsi:nil="true"/>
    <ResourceStatus xsi:nil="true"/>
  </HourlyValues>
  <ShortName>ShrtNme</ShortName>
  ...
</GetUnitHourlyResponse>

```

²⁶ The namespace declaration xmlns:xsi=<http://www.w3.org/2001/XMLSchema-instance> has been removed to improve readability.

7.11 Unit Ramp Rate

This operation involves the details of a unit's generation ramp rate curve for a given day. A ramp rate defines how quickly a unit can increase/decrease its generation in MW per min. See section 7.9 for a definition of a Unit.

7.11.1 Submit Message

7.11.1.1 Purpose of Message

The purpose of this message is to submit information about a generation unit's ramp rate curve. This message will submit information that will be used to create the unit's ramp rate curve; specifically the effective market day, the unit's ID, a ramp rate value in MW/min, and the MW generation range upper bound for the defined ramp rate.

7.11.1.2 Mandatory and Optional Fields

To create or modify a generation unit ramp rate, you need to provide the following. The optional elements RampRateCap and ShortName are read-only elements. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the unit
Yes	Yes	RampRateCap	MWType; Decimal 6.1	The ramp rate capability of the unit from the most recent audit. Read Only
No	No	rate	MWRateType; 5.1	The generation rate of change (MW/min)
No	No	MW	MWType; Decimal 6.1	The upper bound of the MW generation range to which the "rate" applies. The lower bound is defined by another ramp rate curve element where the upper bound is less than this value or zero if none is defined.
Yes	No	ShortName	String; 8	The short name of the unit. Read Only

A limited number of unit IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitUnitRampRate message in the eMktMessages.xsd.

7.11.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitUnitRampRate party="?">
      <!--1 or more repetitions:-->
      <mes:UnitRampRate day="?" ID="?">
        <!--Optional:-->
        <mes:RampRateCap?</mes:RampRateCap>
        <mes:RampRateCurve>
          <!--0 to 10 repetitions:-->
          <mes:RampRatePoint rate="?" MW="?"/>
        </mes:RampRateCurve>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:UnitRampRate>
    </mes:SubmitUnitRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitUnitRampRate>
  <UnitRampRate day="2010-07-07" ID="1234">
    <RampRateCurve>
      <RampRatePoint rate="11.0" MW="1.0"/>
    </RampRateCurve>
  </UnitRampRate>
</SubmitUnitRampRate>
```

Sample of Deleting Ramp Rate Curve

```
<SubmitUnitRampRate>
  <UnitRampRate day="2010-07-07" ID="1234">
    <RampRateCurve/>
  </UnitRampRate>
</SubmitUnitRampRate>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.11.1.4 Additional Value Restrictions

- $0 \leq \text{rate}$; $0 < \text{MW}$
- A warning will be issued if the rate value exceeds the RampRateCap when the demonstrated capability that appears in the GetUnitRampRateResponse message is defined (not NULL)
- The RampRatePoint data can be submitted in any order; they don't need to be listed in increasing order of the MW values

- Complete sets of ramp rate curve segments must be submitted; they cannot be submitted separately and independently. A maximum of ten segments can be submitted.

7.11.2 Query Message

7.11.2.1 Purpose of Message

The purpose of this message is to get a unit's ramp rate curve value(s). Data returned includes the day, unit ID, short name, ramp rate cap, and the ramp rate and MW range upper bound of each curve segment.

7.11.2.2 Mandatory and Optional Fields

To get or query a unit ramp rate curve data, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the unit requested

- Data for only one market date at a time can be queried.
- A limited number of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the unit ID element is not specified then the query returns data for all the units that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

7.11.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetUnitRampRate party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetUnitRampRate>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetUnitRampRate>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetUnitRampRate>
```

7.11.2.4 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal. The data in the response message will be in increasing MW value order. If there is no ramp rate curve defined for an asset, nothing is returned.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the unit. Read Only
No	Yes	RampRateCap	MWType; Decimal 6.1	The ramp rate capability of the unit from the most recent audit. Read Only

Response Elements

```
<GetUnitRampRateResponse>
  ...
  <ShortName>ShrtNme</ShortName>
  <RampRateCap>14</RampRateCap>
  ...
</GetUnitRampRateResponse>
```

7.12 Unit Ramp Rate Hourly

This operation involves the details of a unit's generation ramp rate curve for a given hour. A ramp rate defines how quickly a unit can increase/decrease its generation in MW per min. See section 7.9 for a definition of a Unit.

7.12.1 Submit Message

7.12.1.1 Purpose of Message

The purpose of this message is to submit hourly data relevant to a unit's generation ramp rate. This message will submit information that will be used to override the asset's daily ramp rate curve; specifically the effective market day and hour, the asset's ID, and either a ramp rate value in MW/min or a ramp rate curve. The message can also be used to delete an hourly update by simply setting the "delete" attribute to TRUE on the message element for the hour to delete.

7.12.1.2 Mandatory and Optional Fields

To create or modify a unit's ramp rate, you need to provide the following. The optional element ShortName is a read-only element. Refer to section 2.4 for information about read-only elements.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the unit
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	delete	Boolean; true (1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values
Yes	No	FixedRampRate	MWRateType; 5.1	The ramp rate (MW/min) to use if a curve is not provided.
Yes	No	rate	MWRateType; 5.1	The generation rate of change (MW/min) when provided as part of a curve.
Yes	No	MW	MWType; Decimal 6.1	The upper bound of the MW generation range to which the "rate" applies. The lower bound is defined by another ramp rate curve element where the upper bound is less than this value or zero if none is defined.
Yes	No	ShortName	String; 8	The short name of the unit

A limited number of unit IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitUnitRampRateHourly message in the eMktMessages.xsd.

7.12.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitUnitRampRateHourly party="?">
      <!--1 to 100 repetitions:-->
      <mes:UnitRampRateHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <!--You have a CHOICE of the next 2 items:-->
            <mes:FixedRampRate?</mes:FixedRampRate>
            <!--1 to 10 repetitions:-->
            <mes:RampRateCurve rate="?" MW="?"/>
          </mes:HourlyValues>
        </mes:HourlyProfile>
        <!--Optional:-->
        <mes:ShortName?</mes:ShortName>
      </mes:UnitRampRateHourly>
    </mes:SubmitUnitRampRateHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitUnitRampRateHourly party="AllMighty Holding Co">
  <UnitRampRateHourly day="2013-10-09" ID="326">
    <HourlyProfile>
      <HourlyValues time="2013-10-09T02:00:00-04:00">
        <FixedRampRate>10</FixedRampRate>
      </HourlyValues>
    </HourlyProfile>
  </UnitRampRateHourly>
</SubmitUnitRampRateHourly>
```

Sample of Deleting the Overrides for Particular Hours

```
<SubmitUnitRampRateHourly party="AllMighty Holding Co">
  <UnitRampRateHourly day="2013-10-09" ID="326">
    <HourlyProfile>
      <HourlyValues time="2013-10-09T02:00:00-04:00"
delete="true">
    </HourlyProfile>
  </UnitRampRateHourly>
</SubmitUnitRampRateHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.12.1.4 Additional Value Restrictions

- The XSD imposes a Choice for either a fixed ramp rate value or a ramp rate curve.
- $0 \leq \text{rate}$; $0 < \text{MW}$
- A warning will be issued if the rate value exceeds the RampRateCap when the demonstrated capability that appears in the GetUnitRampRateResponse message is defined (not NULL)
- The RampRatePoint data can be submitted in any order; they don't need to be listed in increasing order of the MW values
- Complete sets of ramp rate curve segments must be submitted; they cannot be submitted separately and independently. A maximum of ten segments can be submitted.

7.12.2 Query Message

7.12.2.1 Purpose of Message

The purpose of this message is to retrieve a submitted unit's hourly ramp rate data that override the daily values of a market day. Data returned includes the day, the unit's name and ID, the unit's hourly ramp rate data if any exists. The ramp rate data can be either a single ramp rate value or a ramp rate curve.

7.12.2.2 Mandatory and Optional Fields

To get or query for unit hourly ramp rate data, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in a time format
Yes	No	ID	Long	The ID of the unit

- Data for only one market date at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A list of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayIDAndHour type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If the unit ID element is not specified then the query returns data for all the units that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetUnitRampRateHourly party="??">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetUnitRampRateHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetUnitRampRateHourly>
  <QueryFilters>
    <Day>2013-08-07</Day>
    <Hour>2013-08-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetUnitRampRateHourly>
```

7.12.2.3 Data Returned

Except for the outermost tags and additional elements shown below, the data returned in the Query Response matches the data described for the Submittal.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	ShortName	String; 8	The short name of the unit. Read Only.

Response Elements

```
<GetUnitRampRateHourlyResponse>
  <UnitRampRateHourly day="2010-07-07" ID="1234">
    <HourlyProfile>
      <HourlyValues time="2013-08-07T16:00:00-04:00">
        <RampRateCurve rate="11.0" MW="1.0"/>
      </HourlyValues>
      <HourlyValues time="2013-08-07T17:00:00-04:00">
        <RampRateCurve rate="11.0" MW="2.0"/>
      </HourlyValues>
    </HourlyProfile>
    <ShortName>ShortName</ShortName>
  </UnitRampRateHourly>
</GetUnitRampRateHourlyResponse>
```

7.13 Fuel Price Adjustment

Participants can submit Fuel Price Adjustments for a specific unit, fuel, and hours of a market day. Participants can also query Fuel Price Adjustments for a given unit, fuel, and market day.

Refer to the [eMarket Users Guide](#), Section 4.13, for a detailed description of the purpose, workings and validation of Fuel Price Adjustment requests.

7.13.1 Submit Message

7.13.1.1 Purpose of Message

The purpose of this message is to submit a Fuel Price Adjustment for a unit, fuel, and range of hours for a market day.

7.13.1.2 Mandatory and Optional Fields

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The id of the unit
No	No	FuelIndex	String; 40	Name of the fuel index
No	No	StartHour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The start hour
No	No	EndHour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The end hour
No	No	RequestedPrice	PriceType; Decimal 6.2	The requested price
Yes	No	FileName	String; 255	Name of the attached file
Yes	No	FileData	Content id	The content id of the MTOM binary file attachment

- The unit specified must belong to the Participant submitting the request. The Participant is determined by the user's authentication certificate or as identified by the "party" attribute.
- The fuel index must be one that the unit is configured to use. The list of fuel indices that a unit is configured to use can be obtained using the GetUnit query message described in §7.9.2

7.13.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitFuelPriceAdjustment party="??">
      <mes:FuelPriceAdjustment day="??" ID="??">
        <mes:FuelIndex?</mes:FuelIndex>
        <mes:StartHour?</mes:StartHour>
        <mes:EndHour?</mes:EndHour>
        <mes:RequestedPrice?</mes:RequestedPrice>
        <mes:Attachments>
          <mes:Attachment>
            <mes:FileName?</mes:FileName>
            <mes:FileData?</mes:FileData>
          </mes:Attachment>
        </mes:Attachments>
      </mes:FuelPriceAdjustment>
    </mes:SubmitFuelPriceAdjustment>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitFuelPriceAdjustment party="P1">
  <FuelPriceAdjustment day="2014-05-15" ID="12345">
    <FuelIndex>Fuel1</FuelIndex>
    <StartHour>2014-05-15T22:00:00-04:00</StartHour>
    <EndHour>2014-05-15T23:00:00-04:00</EndHour>
    <RequestedPrice>123.45</RequestedPrice>
    <Attachments>
      <Attachment>
        <FileName>SampleFile.txt</FileName>
        <FileData>Sample_File_Data</FileData>
      </Attachment>
    </Attachments>
  </FuelPriceAdjustment>
</SubmitFuelPriceAdjustment>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.13.1.4 Additional Value Restrictions

- The market day must be today or tomorrow
- The start hour must be within the specified market day and at least the next hour from the current hour and the message must be received at least 30 minutes before the start of that hour.

- The end hour must be within the specified market day, with due consideration given to DST transition days, and later than the start hour.
- The requested price must be less than \$1000.
- The requested price must be greater than the most recent published Fuel Index Price of the fuel plus a threshold established by the ISO-NE Market Monitor²⁷ for all hours spanned by the request at the time the request is submitted. The FIP of gas fuel indices change at 10:00 every day because the gas fuel day is from 10:00 of one day to 10:00 of the next. Note that the Participant is required by the Market Rule to submit documentation attached to the request justifying the requested price.
- Fuel price adjustment request for a unit that has been locked out by the ISO Internal Market Monitor from submitting requests will be rejected.
- Any file attachments must be “safe” file types. The following file types are deemed “safe”:
 - MS Word documents (.doc, .docx) but not macro-enabled documents (.docm)
 - MS Excel documents (.xls, .xlsx) but not macro-enabled documents (.xlsm)
 - MS Power Point documents (.ppt, .pptx) but not macro-enabled documents (.pptm)
 - Adobe Acrobat document files (.pdf)
 - JPEG and JPG images, GIF images
 - Rich-Text-Format documents (.rtf)
 - Simple text documents (.txt)

7.13.2 Query Message

7.13.2.1 Purpose of Message

The purpose of this message is to get the Fuel Price Adjustments for a given market day, for a list of units, and for a list of fuels.

7.13.2.2 Mandatory and Optional Fields

To get or query for fuel price adjustment data, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The id of the unit
Yes	No	FuelIndex	String; 40	The name of the fuel index

- Data for only one market date at a time can be queried.

²⁷ Refer to Market Rule 1, Appendix A, Section III.A.3.4

- Data for only one fuel index at a time can be explicitly queried.
- A limited number of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayIdAndFuelIndex type in the eMktMessages.xsd. They must all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If no unit ID and no fuel index are specified, then the query returns data on every FPA for all every unit that belongs to the Participant.
- If no unit IDs and a fuel index are specified, then the query returns data on each FPA for that fuel index on every unit that belongs to the Participant.
- If unit IDs and no fuel index are specified, then the query returns data on each FPA for every fuel indices used by each of the unit IDs listed.
- If unit IDs and a fuel index are specified, then the query returns data on each FPA for the list fuel index used by each of the unit IDs listed. The fuel index must be used for the unit IDs listed, otherwise nothing is returned.

7.13.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetFuelPriceAdjustmentUnit party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <mes:ID?</mes:ID>
        <mes:FuelIndex?</mes:FuelIndex>
      </mes:QueryFilters>
    </mes:GetFuelPriceAdjustmentUnit>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetFuelPriceAdjustmentUnit party="P1">
  <QueryFilters>
    <Day>2014-05-14</Day>
  </QueryFilters>
</GetFuelPriceAdjustmentUnit>
```

7.13.2.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The id of the unit

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	UnitLongName	String; 40	The long name of the unit
No	No	FuelIndex	String; 40	The name of the fuel index
No	No	FPAID	Long	The id of the fuel price adjustment
No	No	StartHour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The start hour
No	No	EndHour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The end hour
No	No	RequestedPrice	PriceType; Decimal 6.2	The requested price
No	Yes	EffectivePrice	PriceType; Decimal 6.2	The effective price. Omitted if the effective price has not been determined or is the Fuel Index price.
No	Yes	EffectivePriceIsIndex	Boolean	True if the effective price is the index. If true, the effective price element is omitted.
No	No	SubmittedBy	String; 50	The name of the user that submitted the FPA.
No	No	SubmittedTime	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time that the submitted FPA was received.
No	Yes	WithdrawnBy	String; 50	The name of the user that withdrew the FPA.
No	Yes	WithdrawnHour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The market hour that the FPA was withdrawn.
No	Yes	WithdrawnAcceptedTime	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time that the FPA withdrawal request was received.
No	No	Status	String; 10	The status of the request: Pending / Approved / Capped / Withdrawn / Pending / WithdrawnPending ²⁸
Yes	No	AttachmentID	Long	The id of the file attachment.
Yes	No	FileName	String; 255	The name of the attached file.
Yes	Yes	FileData	Content id	The content id of the MTOM binary attachment.
Yes	No	UploadedBy	String; 50	The name of the user that attached the file.
Yes	No	UploadedTime	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time that the file attachment was received.

²⁸ Refer to the eMarket Users Guide, Section 4.13 for the definition of these status.

Full SOAP Format

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soap:Body>
    <mes:GetFuelPriceAdjustmentUnitResponse>
      <mes:FuelPriceAdjustmentUnit day="" ID="">
        <mes:UnitLongName?></mes:UnitLongName>
        <mes:FuelPriceAdjustmentFuel>
          <mes:FuelIndex?></mes:FuelIndex>
          <mes:FuelPriceAdjustment FPAID="">
            <mes:StartHour?></mes:StartHour>
            <mes:EndHour?></mes:EndHour>
            <mes:RequestedPrice?></mes:RequestedPrice>
            <mes:EffectivePrice?></mes:EffectivePrice>
            <mes:EffectivePriceIsIndex?</
mes:EffectivePriceIsIndex>
            <mes:SubmittedBy?></mes:SubmittedBy>
            <mes:SubmitAcceptedTime?></mes:SubmitAcceptedTime>
            <mes:WithdrawnBy?></mes:WithdrawnBy>
            <mes:WithdrawnHour?></mes:WithdrawnHour>
            <mes:WithdrawnAcceptedTime?</
mes:WithdrawnAcceptedTime>
            <mes:Status?></mes:Status>
            <mes:Attachments>
              <mes:Attachment>
                <mes:AttachmentID?></mes:AttachmentID>
                <mes:FileName?></mes:FileName>
                <mes:FileData?></mes:FileData>
                <mes:UploadedBy?></mes:UploadedBy>
                <mes:UploadedTime?></mes:UploadedTime>
              </mes:Attachment>
            </mes:Attachments>
          </mes:FuelPriceAdjustment>
        </mes:FuelPriceAdjustmentFuel>
      </mes:FuelPriceAdjustmentUnit>
    </mes:GetFuelPriceAdjustmentUnitResponse>
  </mes:soap:Body>
</mes:soap:Envelope>
```

Sample of Query Response

```
<GetFuelPriceAdjustmentUnitResponse>
  <FuelPriceAdjustmentUnit day="2014-05-23" ID="123">
    <UnitLongName>Unit 1</UnitLongName>
    <FuelPriceAdjustmentFuel>
      <FuelIndex>Fuel1</FuelIndex>
      <FuelPriceAdjustment FPAID="12345">
        <StartHour>2014-05-23T13:00:00-04:00</StartHour>
        <EndHour>2014-05-23T15:00:00-04:00</EndHour>
        <RequestedPrice>123.45</RequestedPrice>
        <EffectivePrice xsi:nil="true"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
        <EffectivePriceIsIndex>false</EffectivePriceIsIndex>
        <SubmittedBy>GenUser</SubmittedBy>
        <SubmitAcceptedTime>2014-05-22T15:51:00-
04:00</SubmitAcceptedTime>
        <WithdrawnBy xsi:nil="true"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
        <WithdrawnHour xsi:nil="true"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
        <WithdrawnAcceptedTime xsi:nil="true"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
        <Status>Approved</Status>
        <Attachments>
          <Attachment>
            <AttachmentID>123456789</AttachmentID>
            <FileName>File1.txt</FileName>
            <FileData xsi:nil="true"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
            <UploadedBy>GenUser</UploadedBy>
            <UploadedTime>2014-05-22T15:51:00-04:00</UploadedTime>
          </Attachment>
        </Attachments>
      </FuelPriceAdjustment>
    </FuelPriceAdjustmentFuel>
  </FuelPriceAdjustmentUnit>
</GetFuelPriceAdjustmentUnitResponse>
```

7.14 Fuel Price Adjustment Unit Hourly

Participants can query which if any of the submitted Fuel Price Adjustments is used to establish the fuel price for each hour of a given market day.

7.14.1 Query Message

7.14.1.1 Purpose of Message

The purpose of this message is to get the Fuel Price Adjustments that are in effect for each of the hours of one or more market days, units, and an optional fuel index. This is a sparse result; no hourly elements will be returned for hours that do not have FPAs or the FPA has been withdrawn.

7.14.1.2 Mandatory and Optional Fields

To get or query a fuel price adjustment hourly, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The id of the unit
Yes	No	FuelIndex	String; 40	The name of the fuel index

- Data for only one market date at a time can be queried.
- Data for only one fuel index at a time can be explicitly queried.
- A limited number of unit IDs can be specified; see the maxOccurs attribute of the QueryByDayIdAndFuelIndex type in the eMktMessages.xsd. They must however all be units that belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute.
- If no unit ID and no fuel index are specified, then the query returns hourly data on every FPA for all every unit that belongs to the Participant.
- If no unit IDs and a fuel index are specified, then the query returns hourly data on each FPA for that fuel index on every unit that belongs to the Participant.
- If unit IDs and no fuel index are specified, then the query returns hourly data on each FPA for every fuel indices used by each of the unit IDs listed.
- If unit IDs and a fuel index are specified, then the query returns hourly data on each FPA for the list fuel index used by each of the unit IDs listed. The fuel index must be used for the unit IDs listed, otherwise nothing is returned.

7.14.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetFuelPriceAdjustmentUnitHourly party="??">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <mes:ID?</mes:ID>
        <mes:FuelIndex?</mes:FuelIndex>
      </mes:QueryFilters>
    </mes:GetFuelPriceAdjustmentUnitHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetFuelPriceAdjustmentUnitHourly party="P1">
  <QueryFilters>
    <Day>2014-05-14</Day>
  </QueryFilters>
</GetFuelPriceAdjustmentUnitHourly>
```

7.14.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The id of the unit
No	No	UnitLongName	String; 40	The long name of the unit
No	No	FuelIndex	String; 40	The name of the fuel index
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour the fuel price adjustment is in effect
No	No	FPAID	long	The id of the fuel price adjustment
Yes	No	EffectivePrice	PriceType; Decimal 6.2	The effective price. Omitted if the effective price is null or is the index
No	No	EffectivePriceIsIndex	Boolean	True if the effective price is the index. If true, the effective price element is omitted
No	No	Status	String; 10	The status of the request: Approved / Capped / Withdrawn ²⁹

²⁹ Refer to the eMarket Users Guide, Section 4.13 for the definition of these status.

Full SOAP Format

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soap:Body>
    <mes:GetFuelPriceAdjustmentHourlyResponse>
      <mes:FuelPriceAdjustmentHourly day="?" ID="?">
        <mes:HourlyProfile>
          <mes:HourlyValues time="?">
            <mes:FPAID?</mes:FPAID>
            <mes:EffectivePrice?</mes:EffectivePrice>
            <mes:EffectivePriceIsIndex?</
mes:EffectivePriceIsIndex>
            <mes:Status?</mes:Status>
          </mes:HourlyValues>
        </mes:HourlyProfile>
        <mes:ShortName?</mes:ShortName>
      </mes:FuelPriceAdjustmentHourly>
    </mes:GetFuelPriceAdjustmentHourlyResponse>
  </soap:Body>
</soap:Envelope>
```

Sample of Query Response

```
<GetFuelPriceAdjustmentUnitHourlyResponse>
  <FuelPriceAdjustmentUnitHourly day="2014-05-23" ID="12345">
    <UnitLongName>Unit 1</UnitLongName>
    <FuelPriceAdjustmentFuelHourly>
      <FuelIndex>Fuel1</FuelIndex>
      <HourlyProfile>
        <HourlyValues time="2014-05-23T00:00:00-04:00">
          <FPAID>123456789</FPAID>
          <EffectivePrice>123.45</EffectivePrice>
          <EffectivePriceIsIndex>false</EffectivePriceIsIndex>
          <Status>Capped</Status>
        </HourlyValues>
      </HourlyProfile>
    </FuelPriceAdjustmentFuelHourly>
  </FuelPriceAdjustmentUnitHourly>
</GetFuelPriceAdjustmentUnitHourlyResponse>
```

7.15 Fuel Price Adjustment Withdrawal

Participants can withdraw a previously submitted Fuel Price Adjustment.

7.15.1 Submit Message

7.15.1.1 Purpose of Message

The purpose of this message is to allow a Participant to withdraw one or more previously submitted Fuel Price Adjustments by providing a list of their IDs.

7.15.1.2 Mandatory and Optional Fields

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	FPAID	Long	The id of the Fuel Price Adjustment

The FPA ID must be associated with the Participant that is submitting the request. The Participant is determined by the user's authentication certificate or identified by the "party" attribute of the message.

7.15.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitFuelPriceAdjustmentWithdrawal party="?">
      <mes:FuelPriceAdjustmentWithdrawal FPAID="?"/>
    </mes:SubmitFuelPriceAdjustmentWithdrawal>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<SubmitFuelPriceAdjustmentWithdrawal party="P1">
  <FuelPriceAdjustmentWithdrawal FPAID="123456789"/>
</SubmitFuelPriceAdjustmentWithdrawal>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

7.16 Fuel Price Adjustment Attachment

Participants can submit, delete, and query file attachments for an existing Fuel Price Adjustment.

7.16.1 Submit Message

7.16.1.1 Purpose of Message

The purpose of this message is to allow Participants to attach files to existing Fuel Price Adjustments. The binary files are supplied with the web message using the W3C MTOM (Message Transmission Optimization Mechanism) standard.

7.16.1.2 Mandatory and Optional Fields

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	FPAID	Long	The id of the fuel price adjustment
No	No	FileName	String; 255	Name of the file
No	No	FileData	Content id	The content id of the MTOM binary attachment.

The FPA ID must be associated with the Participant that is submitting the request. The Participant is determined by the user's authentication certificate or identified by the "party" attribute of the message.

7.16.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitFuelPriceAdjustmentAttachment party="?">
      <mes:FuelPriceAdjustment FPAID="?">
        <mes:Attachments>
          <mes:Attachment>
            <mes:FileName?</mes:FileName>
            <mes:FileData?<mes:FileData>
          </mes:Attachment>
        </mes:Attachments>
      </mes:FuelPriceAdjustment>
    </mes:SubmitFuelPriceAdjustmentAttachment>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
-----=_Part_120_541309213.1400780147794
Content-Type: application/xop+xml; charset=UTF-8; type="text/xml"
Content-Transfer-Encoding: 8bit
Content-ID: <rootpart>
<SubmitFuelPriceAdjustmentAttachment party="P1">
  <FuelPriceAdjustment FPAID="123456789">
    <Attachments>
      <Attachment>
        <FileName>SampleFile.txt</FileName>
        <FileData>
          <inc:Include href="cid:SampleFile.txt"
xmlns:inc="http://www.w3.org/2004/08/xop/include"/>
        </FileData>
      </Attachment>
    </Attachments>
  </FuelPriceAdjustment>
</SubmitFuelPriceAdjustmentAttachment>
-----=_Part_120_541309213.1400780147794
Content-Type: text/plain; charset=us-ascii; name=SampleFile.txt
Content-Transfer-Encoding: 7bit
Content-ID: <SampleFile.txt>
Content-Disposition: attachment; name="SampleFile.txt";
filename="SampleFile.txt"
Sample_File_Data
-----=_Part_120_541309213.1400780147794--
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses

7.16.1.4 Additional Value Restrictions

- File attachments must be “safe” file types. The following file types are deemed “safe”:
 - MS Word documents (.doc, .docx) but not macro-enabled documents (.docm)
 - MS Excel documents (.xls, .xlsx) but not macro-enabled documents (.xlsm)
 - MS Power Point documents (.ppt, .pptx) but not macro-enabled documents (.pptm)
 - Adobe Acrobat document files (.pdf)
 - JPEG and JPG images, GIF images
 - Rich-Text-Format documents (.rtf)
 - Simple text documents (.txt)

7.16.2 Delete Message

7.16.2.1 Purpose of Message

The purpose of this message is to allow Participants to remove an existing binary file attachment from a Fuel Price Adjustment by supplying the ID of the FPA and the ID of the attachment.

7.16.2.2 Mandatory and Optional Fields

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	FPAID	Long	The id of the Fuel Price Adjustment
No	No	AttachmentID	Long	The id of the attachment

- The FPA ID must be associated with the Participant that is submitting the request. The Participant is determined by the user's authentication certificate or identified by the "party" attribute of the message.
- The Attachment ID must be associated with the FPA and not already removed

7.16.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:DeleteFuelPriceAdjustmentAttachment party="?">
      <mes:FPAID?</mes:FPAID>
      <mes:AttachmentID?</mes:AttachmentID>
    </mes:DeleteFuelPriceAdjustmentAttachment>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<DeleteFuelPriceAdjustmentAttachment party="P1">
  <FPAID>123456789</FPAID>
  <AttachmentID>123456790</AttachmentID>
</DeleteFuelPriceAdjustmentAttachment>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses

7.16.3 Query Message

7.16.3.1 Purpose of Message

The purpose of this message is to get the attachments for a given Fuel Price Adjustment by the FPA id.

There are two variations of the query message. The first message, **GetFuelPriceAttachment** will return all the data for an attachment, except the file data itself. The second message, **GetFuelPriceAttachmentData** returns the same results as GetFuelPriceAttachment, but also includes the file data as an MTOM attachment to the response.

7.16.3.2 Mandatory and Optional Fields

To get or query a fuel price adjustment attachment, for either **GetFuelPriceAdjustmentAttachment** or **GetFuelPriceAdjustmentAttachmentData** you need to provide the following.

GetFuelPriceAdjustmentAttachment

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	FPAID	Long	The id of the Fuel Price Adjustment

GetFuelPriceAdjustmentAttachmentData

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	FPAID	Long	The id of the Fuel Price Adjustment
No	No	AttachmentID	Long	The id of the attachment

- The FPA ID must be associated with the Participant that is submitting the request. The Participant is determined by the user's authentication certificate or identified by the "party" attribute of the message.
- If an Attachment ID is entered, it must be associated with the FPA ID and the Attachment must not have been removed.
- If no Attachment ID is specified, the response will include all (not yet removed) attachments of the FPA.

7.16.3.3 SOAP Format

Full SOAP Message - GetFuelPriceAdjustmentAttachment

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetFuelPriceAdjustmentAttachment party="?">
      <mes:QueryFilters>
        <mes:FPAID?</mes:FPAID>
        <mes:AttachmentID?</mes:AttachmentID>
      </mes:QueryFilters>
    </mes:GetFuelPriceAdjustmentAttachment>
  </soapenv:Body>
</soapenv:Envelope>
```

Full SOAP Message - GetFuelPriceAdjustmentAttachmentData

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetFuelPriceAdjustmentAttachmentData party=""?>
      <mes:QueryFilters>
        <mes:FPAID?</mes:FPAID>
        <mes:AttachmentID?</mes:AttachmentID>
      </mes:QueryFilters>
    </mes:GetFuelPriceAdjustmentAttachmentData>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal (GetFuelPriceAdjustmentAttachment)

```
<GetFuelPriceAdjustmentAttachment party="P1">
  <QueryFilters>
    <FPAID>123456789</FPAID>
  </QueryFilters>
</GetFuelPriceAdjustmentAttachment>
```

Sample of Query Submittal (GetFuelPriceAdjustmentAttachmentData)

```
<GetFuelPriceAdjustmentAttachmentData party="P1">
  <QueryFilters>
    <FPAID>132456789</FPAID>
    <AttachmentID>132456790</AttachmentID>
  </QueryFilters>
</GetFuelPriceAdjustmentAttachmentData>
```

7.16.3.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	No	FPAID	Long	The id of the fuel price adjustment
Yes	No	AttachmentID	Long	The id of the attachment
Yes	No	FileName	String; 255	The name of the attached file
Yes	Yes	FileData	Content id	The content id of the MTOM binary attachment. This element is omitted in the response to GetFuelPriceAttachment
Yes	No	UploadedBy	String; 50	The name of the user that attached the file
Yes	No	UploadedTime	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The time that the file was attached

Full SOAP Format – GetFuelPriceAdjustmentAttachmentResponse

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soap:Body>
    <mes:GetFuelPriceAdjustmentAttachmentResponse>
      <mes:FuelPriceAdjustment FPAID="??">
        <mes:Attachments>
          <mes:Attachment>
            <mes:AttachmentID?</mes:AttachmentID>
            <mes:FileName?</mes:FileName>
            <mes:UploadedBy?</mes:UploadedBy>
            <mes:UploadedTime?</mes:UploadedTime>
          </mes:Attachment>
        </mes:Attachments>
      </mes:FuelPriceAdjustment>
    </mes:GetFuelPriceAdjustmentAttachmentResponse>
  </soap:Body>
</soap:Envelope>
```

Full SOAP Format – GetFuelPriceAdjustmentAttachmentDataResponse

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soap:Body>
    <mes:GetFuelPriceAdjustmentAttachmentDataResponse>
      <mes:FuelPriceAdjustment FPAID="??">
        <mes:Attachments>
          <mes:Attachment>
            <mes:AttachmentID?</mes:AttachmentID>
            <mes:FileName?</mes:FileName>
            <mes:FileData?</mes:FileData>
            <mes:UploadedBy?</mes:UploadedBy>
            <mes:UploadedTime?</mes:UploadedTime>
          </mes:Attachment>
        </mes:Attachments>
      </mes:FuelPriceAdjustment>
    </mes:GetFuelPriceAdjustmentAttachmentDataResponse>
  </soap:Body>
</soap:Envelope>
```

Sample of Query Response to GetFuelPriceAdjustmentAttachment

```
<GetFuelPriceAdjustmentAttachmentResponse>
  <FuelPriceAdjustment FPAID="123456789">
    <Attachments>
      <Attachment>
        <AttachmentID>123456790</AttachmentID>
        <FileName>SampleFile.txt</FileName>
        <FileData xsi:nil="true"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
        <UploadedBy>GenUser</UploadedBy>
        <UploadedTime>2014-05-15T11:51:53-04:00</UploadedTime>
      </Attachment>
    </Attachments>
  </FuelPriceAdjustment>
</GetFuelPriceAdjustmentAttachmentResponse>
```

Sample of Query Response to GetFuelPriceAdjustmentAttachmentData

```
--uuid:c038a2dd-dba7-4b4d-abb0-157146937565
Content-Type: application/xop+xml; charset=UTF-8; type="text/xml"
Content-Transfer-Encoding: binary
Content-ID: <root.message@cx.f.apache.org>
<GetFuelPriceAdjustmentAttachmentDataResponse>
  <FuelPriceAdjustment FPAID="123456789">
    <Attachments>
      <Attachment>
        <AttachmentID>123456790</AttachmentID>
        <FileName>SampleFile.txt</FileName>
        <FileData>
          <xop:Include
xmlns:xop="http://www.w3.org/2004/08/xop/include"
href="cid:579f7654-d692-4d56-8324-5a071e2bbfd8-317@www.markets.iso-ne.com"/>
        </FileData>
        <UploadedBy>GenUser</UploadedBy>
        <UploadedTime>2014-05-15T11:51:53-04:00</UploadedTime>
      </Attachment>
    </Attachments>
  </FuelPriceAdjustment>
</GetFuelPriceAdjustmentAttachmentDataResponse>
--uuid:c038a2dd-dba7-4b4d-abb0-157146937565
Content-Type: */*
Content-Transfer-Encoding: binary
Content-ID: 579f7654-d692-4d56-8324-5a071e2bbfd8-317@www.markets.iso-ne.com
Content-Disposition: attachment;name="InputStreamDataSource"
Sample_File_Data
--uuid:c038a2dd-dba7-4b4d-abb0-157146937565--
```

8. Regulation Web Services

This section defines the submittals and queries used for Regulation Market operations.

The table below shows a summary of the Regulation messages and indicates if data can be submitted, queried or deleted. It also shows when and how long submitted offer data is effective. With the Regulation Offer message, the submitted data is effective starting on the market day specified in the message and for all subsequent days up to the day for which data is submitted in a separate message. Hourly offer data only applies to a specific market day and hour.

Message	Brief Description of Usage	Actions			Data Remains Effective	
		Submit	Query	Delete	Until data is submitted for a future day	Only for the specified day and hour
Regulation Portfolio	Manage Portfolios of regulation assets	X	X	X	n/a See Note 2	n/a
Regulation Offer	Set or view the Regulation offer data.	X	X		X	
Regulation Physical Hourly	Set or view the Hourly Overrides of Physical Regulation Offers for a day.	X	X	See Note 1		X
Regulation Financial Hourly	Set or view the Hourly Overrides of Financial Regulation Offers for a day.	X	X	See Note 1		X

Note 1: There is no “Delete” web service operation defined to explicitly delete a set of hourly data previously submitted. Data for a specific hour can however be deleted by submitting a message for the hour with the delete attribute set to TRUE.

Note 2: Portfolios are not effective dated. The data in a portfolio exists from when the portfolio is created until the portfolio is deleted. The data (asset list) in a portfolio can be updated using the SubmitPortfolio request; note however that the complete list of asset IDs must be submitted to replace the previously submitted list, if any.

When submitting regulation offer data, it is important to understand the hierarchy of the Regulation messages and the implications of data overrides. Figure 3 below shows the hierarchy of the Regulation messages used for submittals, where the same bid data appears in different messages, and which messages have precedence over

(overrides) the others. **[Note:** only the data elements applicable to this example are shown. See the following sections for the details on the data in each message.]

The Regulation Offer message is at the root of the hierarchy. Participants who own regulation resources must submit a least one Regulation offer using this message if they wish to participate in the Regulation market. The Regulation Offer message establishes the Participant’s offers for a market day and all future days.

Two messages allow for overriding the regulation offer on an hourly basis.

- The Regulation Physical Hourly Updates message provides the means to profile on an hourly basis the resource's availability to participate in the market, its regulation response rate (ARR) and its regulation limits. These hourly updates can be submitted only for future market days; they cannot be changed intraday, i.e. during the operating day.
- The Regulation Financial Hourly Updates message provides the means to profile on an hourly basis the resource's offer. These hourly updates can be submitted at any time for any hour of future days and for future hours of the current day up to 5 minutes before the top of that hour.

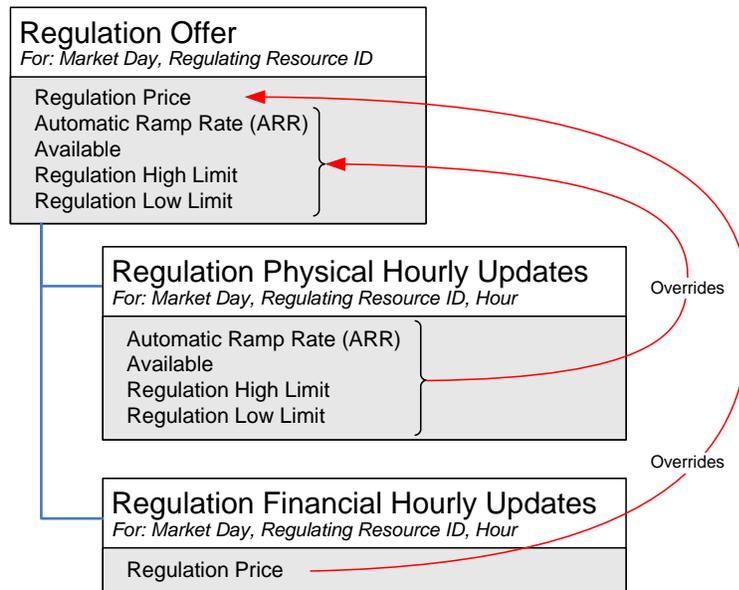


Figure 3 - Illustration of Regulation Message Hierarchy and Data Precedence

8.1 Regulation Portfolio

See section 4.4 for a definition of a portfolio.

8.1.1 Submit Message

This functionality is provided by the Portfolio's Submit Message described in section 4.4.1

8.1.2 Delete Message

This functionality is provided by the Portfolio's Delete Message described in section 4.4.2

8.1.3 Query Message

This functionality is provided by the Portfolio's Query Message described in section 4.4.3

8.2 Regulation Offer

A regulation resource is a specially equipped generating resource that is able to increase or decrease generation output every four seconds in response to signals it receives from ISO New England in order to control changes in the system load and fluctuations on the system tie-lines.

8.2.1 Submit Message

8.2.1.1 Purpose of Message

The purpose of this message is to submit a default regulation offer for a market day and beyond: specifically the market day, the ID of the regulating resource, the automatic ramp rate, the requested price for regulation service and capacity, the regulation limits and the availability of the resource for regulation.

8.2.1.2 Mandatory and Optional Fields

To create or modify an offer, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the resource
No	No	ARR	MWRateType; Decimal 5.1	The automatic ramp rate (ARR) or regulation response rate of the resource
No	No	Available	Boolean; true (1), false (0)	Whether the unit is available for regulation
No	No	RegServicePrice	PriceType; Decimal 6.2	The price requested to cover regulation service cost (\$/MWH)
No	No	RegITOCap	PriceType; Decimal 6.2	The price requested to cover the regulation inter-temporal opportunity capacity cost (\$/MWH)
No	No	RegCapOther	PriceType; Decimal 6.2	The price requested to cover all other regulation capacity cost (\$/MWH)
No	No	RegHighLimit	MWFlowType; Decimal 6.1	The regulation high limit
No	No	RegLowLimit	MWFlowType; Decimal 6.1	The regulation low limit

If the resource ID identifies a generator that is NOT a “RegResource”³⁰, i.e. a unit registered with ISO New England as NOT capable of regulation, the whole message is rejected.

Data for a limited number of resource IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitRegOffer message in the eMktMessages.xsd.

³⁰ Refer to the response of the GetUnit query to get a generator’s RegResource status.

8.2.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitRegOffer party="?">
      <!--1 or more repetitions:-->
      <mes:RegOffer day="?" ID="?">
        <mes:ARR?</mes:ARR>
        <mes:Available?</mes:Available>
        <mes:RegCapOther?</mes:RegCapOther>
        <mes:RegITOCap?</mes:RegITOCap>
        <mes:RegServicePrice?</mes:RegServicePrice>
        <mes:RegHighLimit?</mes:RegHighLimit>
        <mes:RegLowLimit?</mes:RegLowLimit>
      </mes:RegOffer>
    </mes:SubmitRegOffer>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitRegOffer>
  <RegOffer day="2010-07-07" ID="123456">
    <ARR>5.1</ARR>
    <Available>true</Available>
    <RegCapOther>3.78</RegCapOther>
    <RegITOCap>5.01</RegITOCap>
    <RegServicePrice>14.60</RegServicePrice>
    <RegHighLimit>10.0</RegHighLimit>
    <RegLowLimit>6.0</RegLowLimit>
  </RegOffer>
</SubmitRegOffer>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

8.2.1.4 Additional Value Restrictions

- ARR cannot be less than a minimum allowable response rate that is set by ISO New England
- A warning will be issued if the ARR exceeds the resource's ARR demonstrated capability (ARRCap) when the ARRCap value is defined (not NULL). See definition of ARRCap in response to the Regulation Offer query message below.
- For regulation capable generators
 $0 \leq \text{EcoMin} \leq \text{RegLowLimit} < \text{RegHighLimit} \leq \text{EcoMax}$, where the economic limits are the more constraining of the limits among the

daily default (those submitted using the Schedule Detail message) and the hourly updates (those submitted using the Unit Hourly message) for the selected market day. This is the **limit nesting rule**. A warning message is returned when this rule is violated.

- For ATRRs, $RLLFloor \leq RegLowLimit < RegHighLimit \leq RHLCeil$, where RLLFloor and RHLCeil are thresholds set by the ISO New England that are returned in the GetRegOfferResponse message.
- Total Regulation Capacity offer price (RegCapOther + RegITOCap) cannot exceed \$100.00 and must be greater than the Regulation Floor Price determined by the ISO New England.
- Regulation Service offer price (RegServicePrice) cannot exceed \$10.00 and must be greater than the Regulation Floor Price determined by the ISO New England.

8.2.2 Query Message

8.2.2.1 Purpose of Message

The purpose of this message is to get regulation service offers that are in place for a market day and for one or more regulation resources. Data returned includes the day, the resource ID, its automatic ramp rate, its demonstrated ramp rate capability, its daily regulation limits along with the daily regulation limit thresholds, its capacity and service offer prices for regulation, the resource's availability for regulation, its calculated regulation capability value, various factors set by the ISO New England and any associated comments.

8.2.2.2 Mandatory and Optional Fields

To get or query a regulation offer, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the resource

- One market date followed by a list of resource IDs may be specified per query.
- A maximum of 100 resource IDs can be specified. They should however all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute and they should be IDs for ATRRs or units that are "RegResources"³¹, i.e. registered with ISO New England as units capable of regulation. If either of these conditions is not satisfied, no data will be returned for the unit ID.
- If the resource ID element is not specified then the query returns data for all the resources that are associated with the Participant as

³¹ Refer to the response of the GetUnit query to get a unit's RegResource status.

determined by the user's authentication certificate or as identified by the "party" attribute and that are "RegResources".

8.2.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetRegOffer party="">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 100 repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetRegOffer>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetRegOffer>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <ID>123456</ID>
  </QueryFilters>
</GetRegOffer>
```

8.2.2.4 Data Returned

The data returned in the Query Response is described in the following table.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	ARR	MWRateType; Decimal 5.1	The automatic ramp rate (ARR, aka regulation response rate) of the resource
No	No	Available	Boolean; true (1), false (0)	Whether the unit is available for regulation
No	Yes	ARRCap	MWRateType; Decimal 5.1	The demonstrated automatic ramp rate capability of the resource from the most recent audit.
No	Yes	Capability	MWType; Decimal 6.1	The MW regulation capability calculated from the ARR, the regulation limits and the derate factor.
No	Yes	DerateComment	String; 512	A comment entered by the Market Administrator the last time the DerateFactor value was changed

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	DerateFactor	DerateFactorType; Decimal 3.2	A factor between 0 and 1 that is calculated based on historical performance that provides an estimate of the resources future capability to respond to an AGC setpoint.
No	Yes	DispatchMethod	Enumerated String; CONT, ENC or ENT	The regulating resource dispatch method: <ul style="list-style-type: none"> • CONT – Continuous (Not energy neutral) • ENC – Energy Neutral Continuous • ENT – Energy Neutral Trinary
No	Yes	ExcludedComment	String; 512	A comment entered by the Market Administrator when the resource was excluded from the regulation market.
No	Yes	Excluded	Boolean; true (1), false (0)	Whether the resource is excluded from the regulation market by the Market Administrator.
No	Yes	ServiceComment	String; 512	A comment entered by the Market Administrator the last time the ServiceFactor value was changed
No	Yes	ServiceFactor	ServiceFactorType; Decimal 6.2	A factor that is calculated based on historical performance that provides an estimate of the resource's regulation service as a multiple of capacity.
No	Yes	RegServicePrice	PriceType; Decimal 6.2	The price requested to cover regulation service cost (\$/MWH)
No	Yes	RegITOCap	PriceType; Decimal 6.2	The price requested to cover the regulation inter-temporal opportunity capacity cost (\$/MWH)
No	Yes	RegCapOther	PriceType; Decimal 6.2	The price requested to cover all other regulation capacity cost (\$/MWH)
No	Yes	RegHighLimit	MWFlowType; Decimal 6.1	The regulation high limit
No	Yes	RegLowLimit	MWFlowType; Decimal 6.1	The regulation low limit
No	Yes	RHLCeil	MWFlowType; Decimal 6.1	The regulation high limit threshold. For generators, this is the generator's daily economic maximum limit.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	RLLFloor	MWFlowType; Decimal 6.1	The regulation low limit threshold. For generators, this is the generator's daily economic minimum limit.

Response Elements

```

<GetRegOfferResponse>
  ...
  <ARR>3</ARR>
  <Available>>true</Available>
  <ARRCap xsi:nil="true"/>
  <Capability>5</Capability>
  <DerateComment>Initial derate factor of 100%</DerateComment >
  <DerateFactor>1.0</DerateFactor>
  <DispatchMethod>CONT</DispatchMethod>
  <ExcludedComment xsi:nil="true"/>
  <Excluded>>false</Excluded>
  <RegHighLimit>176.0</RegHighLimit>
  <RegLowLimit>166.0</RegLowLimit>
  <RegCapOther>8.95</RegCapOther>
  <RegITOCap>6.95</RegITOCap>
  <RegServicePrice>17.10</RegServicePrice>
  <RHLCeil>177.0</RHLCeil>
  <RLLFloor>165.0</RLLFloor>
  <ServiceComment>Initial service factor of 10.0</ServiceComment>
  <ServiceFactor>10.0</ServiceFactor>
  ...
</GetRegOfferResponse>

```

8.3 Regulation Hourly

See section 8.2 for a definition of a regulating resource.

The Regulation Hourly operations are used to submit/modify/view hourly overrides of the daily regulation offer values for a market day maintained using the Regulation Offer operations described in 8.2.

8.3.1 Submit Message

The Submit Regulation Hourly operations are separated in two: one for the regulation financial parameters that can be submitted at any time for any future hour, and one for the regulation physical parameters that can be submitted only for future days.

8.3.1.1 Regulation Hourly Physical Offers

8.3.1.1.1 Purpose of Message

The purpose of this message is to submit or update a resource's regulation hourly physical offer parameters. This message contains data relevant to an hourly update of the resource's physical regulation parameters; specifically the market day and hour, the ID of the resource, and the physical offer parameters: the resource's availability for regulation, its automatic ramp rate and its regulation high/low limits. The message can also be used to delete an hourly update by simply setting the "delete" attribute to TRUE on the message element for the hour to delete.

8.3.1.1.2 Mandatory and Optional Fields

To add or update an hourly update of a resource's physical regulation offer parameters, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the resource
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The market hour in time of the day format.
Yes ³²	No	ARR	MWRateType; Decimal 5.2	The automatic ramp rate (ARR) of the resource
Yes ³²	No	Available	Boolean; true (1), false (0)	Whether the resource is available for regulation
Yes ³²	No	RegHighLimit	MWFlowType; Decimal 6.1	The regulation high limit
Yes ³²	No	RegLowLimit	MWFlowType; Decimal 6.1	The regulation low limit

³² The element is optional but only if the attribute "Delete" is true.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
Yes	No	delete	Boolean; true(1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values

To delete an hourly update of a resource’s physical regulation offer parameters, you need only provide the “delete” attribute set to TRUE and omit the other four elements.

If the resource ID identifies a generator that is NOT a “RegResource”³³, i.e. a unit registered with ISO New England as NOT capable of regulation, the whole message is rejected.

Data for a limited number of resource IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitRegPhysicalHourly message in the eMktMessages.xsd.

8.3.1.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitRegPhysicalHourly party="?">
      <!--1 or more repetitions:-->
      <mes:RegPhysicalHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--0 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <mes:Available?</mes:Available>
            <mes:ARR?</mes:ARR>
            <mes:RegHighLimit?</mes:RegHighLimit>
            <mes:RegLowLimit?</mes:RegLowLimit>
          </mes:HourlyValues>
        </mes:HourlyProfile>
      </mes:RegPhysicalHourly>
    </mes:SubmitRegPhysicalHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

³³ Refer to the response of the GetUnit query to get a generator’s RegResource status.

Sample of Submittal

```
<SubmitRegPhysicalHourly>
  <RegPhysicalHourly day="2010-07-07" ID="12345678">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T16:00:00-04:00">
        <Available>true</Available>
        <ARR>5.1</ARR>
        <RegHighLimit>10.0</RegHighLimit>
        <RegLowLimit>6.0</RegLowLimit>
      </HourlyValues>
    </HourlyProfile>
  </RegPhysicalHourly>
</SubmitRegPhysicalHourly>
```

Sample of Deleting the Overrides for Particular Hours

```
<SubmitRegPhysicalHourly>
  <RegPhysicalHourly day="2010-07-07" ID="12345678">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T16:00:00-04:00"
        delete="true"/>
      <HourlyValues time="2010-07-07T18:00:00-04:00"
        delete="true"/>
      <HourlyValues time="2010-07-07T20:00:00-04:00"
        delete="true"/>
    </HourlyProfile>
  </RegPhysicalHourly>
</SubmitRegPhysicalHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

8.3.1.1.4 Additional Value Restrictions

- ARR cannot be less than a minimum allowable response rate that is set by ISO New England
- A warning will be issued if the ARR exceeds the resource's ARR demonstrated capability (ARRCap) when the ARRCap value is defined (not NULL). See definition of ARRCap in response to the Regulation Offer query message.
- For regulation capable generators
 $0 \leq \text{EcoMin} \leq \text{RegLowLimit} < \text{RegHighLimit} \leq \text{EcoMax}$, where
EcoMin = effective hourly economic minimum MW dispatch limit
EcoMax = effective hourly economic maximum MW dispatch limit
and the effective hourly limit values are those for the corresponding hour derived from the hourly limit values submitted with a SubmitUnitHourly message (see 7.10 Unit Hourly) if an hourly update is defined else from the daily limit values found in the details (see 7.6 Schedule Detail) of all schedules designated as available for the selected market day.

- For ATRRs,
 $RLLFloor \leq RegLowLimit < RegHighLimit \leq RHLCeil$, where
RLLFloor and RHLCeil are thresholds set by the ISO New England that
are returned in the GetRegOfferResponse message.

8.3.1.2 Regulation Hourly Financial Offers

8.3.1.2.1 Purpose of Message

The purpose of this message is to submit or update a resource’s regulation hourly financial offer parameters. This message contains data relevant to an hourly update of the resource’s financial regulation parameters; specifically the market day and hour, the ID of the resource, and the financial offer parameters: the service price, the Inter Temporal Opportunity (ITO) capacity price and the price for all Other capacity. The message can also be used to delete an hourly update by simply setting the “delete” attribute to TRUE on the message element for the hour to delete.

8.3.1.2.2 Mandatory and Optional Fields

To add or update an hourly update of a resource’s financial regulation offer parameters, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the resource
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The market hour in time of the day format.
Yes ³⁴	No	RegServicePrice	PriceType; Decimal 6.2	The price requested to cover regulation service cost (\$/MWH)
Yes ³⁴	No	RegITOCap	PriceType; Decimal 6.2	The price requested to cover the regulation inter-temporal opportunity capacity cost (\$/MWH)
Yes ³⁴	No	RegCapOther	PriceType; Decimal 6.2	The price requested to cover all other regulation capacity cost (\$/MWH)
Yes	No	delete	Boolean; true(1), false (0) Default value: false	When set to true, the update record for the specified market day and hour is deleted, irrespective of the other element values

To delete an hourly update of a resource’s financial regulation offer parameters, you need only provide the “delete” attribute set to TRUE and omit the other three elements.

³⁴ The element is optional but only if the attribute “Delete” is true.

If the resource ID identifies a generator that is NOT a “RegResource”³⁵, i.e. a unit registered with ISO New England as NOT capable of regulation, the whole message is rejected.

Data for a limited number of resource IDs can be submitted in one request; refer to the maxOccurs attribute of the SubmitRegFinancialHourly message in the eMktMessages.xsd.

8.3.1.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:SubmitRegFinancialHourly party="?">
      <!--1 or more repetitions:-->
      <mes:RegFinancialHourly day="?" ID="?">
        <mes:HourlyProfile>
          <!--0 to 25 repetitions:-->
          <mes:HourlyValues time="?" delete="?">
            <mes:RegCapOther?</mes:RegCapOther>
            <mes:RegITOCap?</mes:RegITOCap>
            <mes:RegServicePrice?</mes:RegServicePrice>
          </mes:HourlyValues>
        </mes:HourlyProfile>
      </mes:RegFinancialHourly>
    </mes:SubmitRegFinancialHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Submittal

```
<SubmitRegFinancialHourly>
  <RegFinancialHourly day="2010-07-07" ID="12345678">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T16:00:00-04:00">
        <RegCapOther>3.78</RegCapOther>
        <RegITOCap>5.01</RegITOCap>
        <RegServicePrice>14.60</RegServicePrice>
      </HourlyValues>
    </HourlyProfile>
  </RegFinancialHourly>
</SubmitRegFinancialHourly>
```

³⁵ Refer to the response of the GetUnit query to get a generator’s RegResource status.

Sample of Deleting the Overrides for Particular Hours

```
<SubmitRegFinancialHourly>
  <RegFinancialHourly day="2010-07-07" ID="12345678">
    <HourlyProfile>
      <HourlyValues time="2010-07-07T16:00:00-04:00"
        delete="true"/>
      <HourlyValues time="2010-07-07T18:00:00-04:00"
        delete="true"/>
      <HourlyValues time="2010-07-07T20:00:00-04:00"
        delete="true"/>
    </HourlyProfile>
  </RegFinancialHourly>
</SubmitRegFinancialHourly>
```

Response Message

This message has a standard response as defined in section 2.1 Submit and Query Responses.

8.3.1.2.4 Additional Value Restrictions

- Total Regulation Capacity offer price (RegCapOther + RegITOCap) cannot exceed \$100.00 and must be greater than the Regulation Floor Price determined by ISO New England.
- Regulation Service offer price (RegServicePrice) cannot exceed \$10.00 and must be greater than the Regulation Floor Price determined by the ISO New England.

8.3.2 Query Message

8.3.2.1 Purpose of Message

The purpose of this message is to get the regulation hourly offer override values for a given market day, for a list of market hours and for a list of resources. Data returned includes the resource's regulation capability and its short name.

8.3.2.2 Mandatory and Optional Fields

To get or query a regulation resource, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	Hour	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The market hour in time of the day format.
Yes	No	ID	Long	The ID of the resource

- Data for a single market data at a time can be queried.
- If the Hour element is not specified then the query returns data for every hour of the market day.
- A limited number of resource IDs can be specified; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd. They

should all belong to the Participant as determined by the user's authentication certificate or as identified by the "party" attribute. If a generator ID is on the list, the generator must be a RegResource³⁶, i.e. registered with ISO New England as generator capable of regulation. If either of these conditions is not satisfied, no data will be returned for the resource ID.

- If the resource ID element is not specified then the query returns data for all the resources that are associated with the Participant as determined by the user's authentication certificate or as identified by the "party" attribute and for generators that are flagged as RegResource.

8.3.2.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetRegHourly party="?">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--0 to 25 repetitions:-->
        <mes:Hour?</mes:Hour>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetRegHourly>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetRegHourly>
  <QueryFilters>
    <Day>2010-07-07</Day>
    <Hour>2010-07-07T14:00:00-04:00</Hour>
    <ID>123456</ID>
  </QueryFilters>
</GetRegHourly>
```

8.3.2.4 Data Returned

The data returned in the Query Response is shown below. If an hourly override value does not exist for an hour, a "nil" value is returned. The data in the response message will be in ascending hour value.

³⁶ Refer to the response of the GetUnit query to get a unit's RegResource status.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	Yes	ARR	MWRateType; Decimal 5.1	The automatic ramp rate (ARR) of the resource for the hour
No	Yes	Available	Boolean; true (1), false (0)	Whether the unit is available for regulation for the hour.
No	Yes	Capability	MWType; Decimal 6.1	The MW regulation capability for the hour calculated from the ARR for the hour, the regulation limits for the hour and the derate factor for the day. In the absence of ARR and regulation limits for the hour, the daily values are used.
No	Yes	RegHighLimit	MWFlowType; Decimal 6.1	The regulation high limit for the hour.
No	Yes	RegLowLimit	MWFlowType; Decimal 6.1	The regulation low limit for the hour.
No	Yes	RegCapOther	PriceType; Decimal 6.2	The price requested to cover all other regulation capacity cost (\$/MWH) for the hour.
No	Yes	RegITOCap	PriceType; Decimal 6.2	The price requested to cover the regulation inter-temporal opportunity capacity cost (\$/MWH) for the hour.
No	Yes	RegServicePrice	PriceType; Decimal 6.2	The price requested to cover regulation service cost (\$/MWH) for the hour.
No	No	ShortName	String; 8	The short name of the resource.

Response Elements

```
<GetRegHourlyResponse>
  <!--1 or more repetitions for each resource ID:-->
  <RegHourly day="?" ID="?">
    <HourlyProfile>
      <!--0 to 25 repetitions:-->
      <HourlyValues time="?">
        <ARR?</ARR>
        <Available?</Available>
        <Capability?</Capability>
        <RegHighLimit?</RegHighLimit>
        <RegLowLimit?</RegLowLimit>
        <RegCapOther?</RegCapOther>
        <RegITOCap?</RegITOCap>
        <RegServicePrice?</RegServicePrice>
      </HourlyValues>
      <HourlyValues time="?">
        ...
      </HourlyValues>
    </HourlyProfile>
    <ShortName?</ShortName>
  </RegHourly>
  <RegHourly day="?" ID="?">
    ...
  </RegHourly>
</GetRegHourlyResponse>
```

9. Public Web Services

This section defines the queries used for Public Web Service operations, most of which provide information on what cleared in the Day-Ahead market. Information on what cleared is available only after the Day-Ahead Market has been cleared and the time for the opening of the Re-Offer Period of the selected day has passed.

9.1 Bid Types

This operation is used to get the various bid types that a Market Participant can submit. The possible bid types are demand, fixed, increment, and price sensitive.

9.1.1 Query Message

9.1.1.1 Purpose of Message

The purpose of this message is to get the bid types that can be submitted by the Market Participant.

9.1.1.2 Mandatory and Optional Fields

None

9.1.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetBidTypes party="?" />
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetBidTypes/>
```

9.1.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Type	Enumerated String; Demand, Fixed, Increment, PriceSensitive	The type of bid
No	No	Demand	Boolean; true (1), false (0)	Whether the bid is a demand bid
No	No	Description	String; 40	The description of the bid

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetBidTypesResponse>
      <mes:BidTypes>
        <!--1 or more repetitions:-->
        <mes:BidType>
          <mes:Type?</mes:Type>
          <mes:Demand?</mes:Demand>
          <mes:Description?</mes:Description>
        </mes:BidType>
      </mes:BidTypes>
    </mes:GetBidTypesResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetBidTypesResponse>
  <BidTypes>
    <BidType>
      <Type>Increment</Type>
      <Demand>>false</Demand>
      <Description>Increment</Description>
    </BidType>
    <BidType>
      <Type>Decrement</Type>
      <Demand>>true</Demand>
      <Description>Decrement</Description>
    </BidType>
    <BidType>
      <Type>Fixed</Type>
      <Demand>>true</Demand>
      <Description>Fixed Demand</Description>
    </BidType>
    <BidType>
      <Type>PriceSensitive</Type>
      <Demand>>true</Demand>
      <Description>Price Sensitive Demand</Description>
    </BidType>
  </BidTypes>
</GetBidTypesResponse>
```

9.2 Binding Limits

Binding limits are constraints applied for a particular hour on a particular date, and can be such things as transmission or generation constraints.

9.2.1 Query Message

9.2.1.1 Purpose of Message

The purpose of this message is to get the constraints for each hour that were applied in the approved Day-Ahead market case of a particular date. This response will contain information about the constraints such as the day/hour, branch name, and contingency name.

The data for this message is available after the Re-Offer period for the market day opens.

9.2.1.2 Mandatory and Optional Fields

To get binding limits, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

9.2.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetBindingLimits party="?">
      <mes:Day?</mes:Day>
    </mes:GetBindingLimits>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetBindingLimits>
  <Day>2010-07-07</Day>
</GetBindingLimits>
```

9.2.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	BranchName	String; 55	The branch name
No	No	ContingencyName	String; 40	The contingency name
No	No	Hour	DateTime; yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in time of day format

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetBindingLimitsResponse>
      <!--Zero or more repetitions:-->
      <mes:BindingLimits day="">
        <!--Zero or more repetitions:-->
        <mes:BindingLimit>
          <mes:BranchName?</mes:BranchName>
          <mes:ContingencyName?</mes:ContingencyName>
          <mes:HoursImpacted>
            <!--1 to 25 repetitions:-->
            <mes:Hour?</mes:Hour>
          </mes:HoursImpacted>
        </mes:BindingLimit>
      </mes:BindingLimits>
    </mes:GetBindingLimitsResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetBindingLimitsResponse>
  <BindingLimits day="2010-05-07">
    <BindingLimit>
      <BranchName>SCOBIE_N124_A_LN</BranchName>
      <ContingencyName>Actual</ContingencyName>
      <HoursImpacted>
        <Hour>2010-05-07T21:00:00-04:00</Hour>
        <Hour>2010-05-07T23:00:00-04:00</Hour>
      </HoursImpacted>
    </BindingLimit>
  </BindingLimits>
</GetBindingLimitsResponse>
```

9.3 Interface Limits

Interface limits are operating limits imposed on interfaces between zones within the ISO New England control area.

9.3.1 Query Message

9.3.1.1 Purpose of Message

The purpose of this message is to get the flow and limit values on the interfaces for each hour of a particular market day from the approved Day-Ahead market case. This response will contain information about interface limits, such as the day/time, name, flow, and limit of an interface.

The data for this message is available after the Re-Offer period for the market day opens.

9.3.1.2 Mandatory and Optional Fields

To get interface limits, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

9.3.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetInterfaceLimits party="?">
      <mes:Day?</mes:Day>
    </mes:GetInterfaceLimits>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetInterfaceLimits>
  <Day>2010-07-07</Day>
</GetInterfaceLimits>
```

9.3.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	Name	String; 40	The name of the interface
No	No	time	DateTime; yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in time of day format

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	flow	MWFlowType; Decimal 6.1	The MW flow through the interface for that hour
No	No	limit	MWFlowType; Decimal 6.1	The MW flow limit for that hour

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetInterfaceLimitsResponse>
      <!--Zero or more repetitions:-->
      <mes:InterfaceLimits day="?">
        <!--Zero or more repetitions:-->
        <mes:InterfaceLimit>
          <mes:Name?</mes:Name>
          <mes:HourlyProfile>
            <!--1 to 25 repetitions:-->
            <mes:HourlyValues time=?" flow=?" limit=?"/>
          </mes:HourlyProfile>
        </mes:InterfaceLimit>
      </mes:InterfaceLimits>
    </mes:GetInterfaceLimitsResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetInterfaceLimitsResponse>
  <InterfaceLimits day="2010-05-07">
    <InterfaceLimit>
      <Name>NHME</Name>
      <HourlyProfile>
        <HourlyValues time="2010-05-07T11:00:00-04:00"
          flow="518" limit="9999"/>
        <HourlyValues time="2010-05-07T12:00:00-04:00"
          flow="520" limit="9999"/>
      </HourlyProfile>
    </InterfaceLimit>
  </InterfaceLimits>
</GetInterfaceLimitsResponse>
```

9.4 Net Tie Schedules

Net tie schedules show the scheduled amount of power flowing between ISO New England and its neighbors for each hour of a specified day.

9.4.1 Query Message

9.4.1.1 Purpose of Message

The purpose of this message is to get the net MW flow on the ties for each hour of a market day from the approved Day-Ahead market case. The response will contain information about the cleared flow, such as the day/time, area, and MW flow amount.

The data for this message is available after the Re-Offer period for the market day opens.

9.4.1.2 Mandatory and Optional Fields

To get net tie schedules, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

9.4.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNetTieSchedules party="?">
      <mes:Day?</mes:Day>
    </mes:GetNetTieSchedules>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetNetTieSchedules>
  <Day>2010-07-07</Day>
</GetNetTieSchedules>
```

9.4.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	area	String; 40	The area name
No	No	time	DateTime; yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in time of day format
No	No	MW	MWFlowType; Decimal 6.1	The MW amount flowing

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetNetTieSchedulesResponse>
      <!--Zero or more repetitions:-->
      <mes:NetTieSchedules day="?">
        <!--Zero or more repetitions:-->
        <mes:NetTieSchedule area="?">
          <!--1 to 25 repetitions:-->
          <mes:HourlyValue time=?" MW=?"/>
        </mes:NetTieSchedule>
      </mes:NetTieSchedules>
    </mes:GetNetTieSchedulesResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetNetTieSchedulesResponse>
  <NetTieSchedules day="2010-05-07">
    <NetTieSchedule area=".I.HQ_P1_P2345 5">
      <HourlyValue time="2010-05-07T20:00:00-04:00" MW="1138"/>
      <HourlyValue time="2010-05-07T21:00:00-04:00" MW="1138"/>
    </NetTieSchedule>
  </NetTieSchedules>
</GetNetTieSchedulesResponse>
```

9.5 Pricing

Pricing provides the Day-Ahead market price of energy for every hour of a market day at the pricing nodes in the ISO control area.

9.5.1 Query Message

9.5.1.1 Purpose of Message

The purpose of this message is to get the DA Clearing prices for each hour of a market day at all or specific public pricing nodes.

The data for this message is available after the Re-Offer period for the market day opens.

9.5.1.2 Mandatory and Optional Fields

To get the DA prices for a particular market day, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day
Yes	No	ID	Long	The ID of the node requested

- Data for only one market date at a time can be queried.
- A limited number of IDs can be listed; see the maxOccurs attribute of the QueryByDayAndId type in the eMktMessages.xsd.
- If the ID element is not specified then the query returns data for all the public pricing nodes (Pnodes). These are the nodes listed in the response to the GetNode query request described in section 4.2. It is the same as the report obtained by selecting the Day-Ahead Prices on the Public XML Download display tab of the eMarket MUI.

9.5.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetPrices party="">
      <mes:QueryFilters>
        <mes:Day?</mes:Day>
        <!--Zero or more repetitions:-->
        <mes:ID?</mes:ID>
      </mes:QueryFilters>
    </mes:GetPrices>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetPrices>
  <Day>2010-07-07</Day>
  <ID>123456</ID>
</GetPrices>
```

9.5.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	ID	Long	The ID of the node
No	No	name	String; 40	The name of the node
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in time of day format
No	No	price	PriceType; Decimal 6.2	The DA market price of energy (\$ per MWH) at that node for that hour

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetPricesResponse>
      <!--Zero or more repetitions:-->
      <mes:Prices day="?">
        <!--Zero or more repetitions:-->
        <mes:NodePrices ID=?" name=?">
          <!--1 to 25 repetitions:-->
          <mes:HourlyPrice time=?" price=?"/>
        </mes:NodePrices>
      </mes:Prices>
    </mes:GetPricesResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetPricesResponse>
  <Prices day="2010-05-07">
    <NodePrices ID="4000" name=".H.INTERNAL_HUB">
      <HourlyPrice time="2010-05-07T00:00:00-04:00"
        price="32.62"/>
      <HourlyPrice time="2010-05-07T01:00:00-04:00"
        price="32.68"/>
    </NodePrices>
  </Prices/>
</GetPricesResponse>
```

9.6 System Summary

System summary provides the following hourly information for a given market day: the MW amount of demand that cleared, the forecasted MW load, and the amount of MW reserve for every hour of the day from the approved Day-Ahead case.

9.6.1 Query Message

9.6.1.1 Purpose of Message

The purpose of this message is to get the following hourly values from an approved DA case for a specific market day:

- Forecasted MW load
- Cleared MW Demand
- Amount of MW Reserves

The data for this message is available after the Re-Offer period for the market day opens.

9.6.1.2 Mandatory and Optional Fields

To get a system summary data, you need to provide the following.

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	Day	Date; yyyy-mm-dd	The market day

9.6.1.3 SOAP Format

Full SOAP Message

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetSystemSummary party="?">
      <mes:Day?</mes:Day>
    </mes:GetSystemSummary>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Submittal

```
<GetSystemSummary>
  <Day>2010-07-07</Day>
</GetSystemSummary>
```

9.6.1.4 Data Returned

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	day	Date; yyyy-mm-dd	The market day
No	No	name	String; 40	The name of the area

Opt.	Nil.	Element or Attribute	Data Type; Format	Comments
No	No	time	DateTime: yyyy-mm-ddThh:mm:ss-hh:mm	The hour of the day in time of day format
No	No	DemandBidMW	MWType; Decimal 6.1	The total amount of demand in MW that cleared for that hour
No	No	DemandForecastMW	MWType; Decimal 6.1	The forecasted load in MW for that hour
No	Yes	Reserve	MWType; Decimal 6.1	The total amount of reserve in MW

Full SOAP Format

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mes="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
  <soapenv:Header/>
  <soapenv:Body>
    <mes:GetSystemSummaryResponse>
      <!--Zero or more repetitions:-->
      <mes:SystemSummary day="??">
        <!--Zero or more repetitions:-->
        <mes:AreaSummary name="??">
          <!--1 to 25 repetitions:-->
          <mes:HourlyValues time="??">
            <mes:DemandBidMW?</mes:DemandBidMW>
            <mes:DemandForecastMW?</mes:DemandForecastMW>
            <mes:Reserve?</mes:Reserve>
          </mes:HourlyValues>
        </mes:AreaSummary>
      </mes:SystemSummary>
    </mes:GetSystemSummaryResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Sample of Query Response

```
<GetSystemSummaryResponse>
  <SystemSummary day="2010-05-07">
    <AreaSummary name="NEPEX">
      <HourlyValues time="2010-05-07T00:00:00-04:00">
        <DemandBidMW>9629</DemandBidMW>
        <DemandForecastMW>10540</DemandForecastMW>
        <Reserve>1687</Reserve>
      </HourlyValues>
      <HourlyValues time="2010-05-07T01:00:00-04:00">
        <DemandBidMW>9410</DemandBidMW>
        <DemandForecastMW>10010</DemandForecastMW>
        <Reserve>1687</Reserve>
      </HourlyValues>
    </AreaSummary>
  </SystemSummary>
</GetSystemSummaryResponse>
```

Appendix A. Web Service Client – C#

Below is an example of how to access the eMarket web services using C#. The code below is abbreviated for readability. The complete example and an electronic copy of the sample code is available from ISO New England Customer Support services.

```
using System.ServiceModel; //EndpointAddress
using System.Security.Cryptography.X509Certificates; //X509 classes
using System.Net; //ServicePointManager
using System.Net.Security; //SslPolicyErrors
using System.Xml.Serialization; //XmlSerializer

namespace MUISample
{
    ///<summary>
    /// This sample program shows how to interact with eMarket Market User
    /// Interface (MUI) using web services. The program relies on the
    /// presence of the EnergyMarketMUI interface which is generated from
    /// the eMktMUI.wsdl using the Microsoft svcutil program.
    ///
    /// The following command was used to generate code for this program:
    /// svcutil eMktMUI.wsdl eMktMessages.xsd
    /// This generates the EnergyMarketMUIService.cs file which contains most
    /// of the classes used in this example.
    ///</summary>

    class Program
    {
        static void Main(string[] args)
        {
            try
            {
                // Setup the proxy to the web service.
                EnergyMarketMUI proxy = CreateWebServiceClient();
                // Try a simple Query and Submit type operation.
                TestGetNode(proxy);
                TestSubmitPortfolio(proxy);
            }
            catch (FaultException<MUIFault> soapFault)
            {
                // Handle fault
            }
            catch (Exception e)
            {
                // Handle fault
            }
            Environment.Exit(1);
        }

        //-----
        // Executes a "GetNode" operation and prints out the results.
        //-----
        private static void TestGetNode(EnergyMarketMUI proxy)
        {

```

```

// The XML structure for this message is as follows:
//   <GetNode party="SomeParty"/>

// Create the containing request object
GetNodeRequest request = new GetNodeRequest();

// Create the GetNode object which maps to the <GetNode> element.
request.GetNode = new GetNode();
request.GetNode.party = "SomeParty";

// Call the service
GetNodeResponse response = proxy.GetNode(request);

// Print out response detail
}

//-----
// Executes a "SubmitPortfolio" operation and prints out the results.
//-----
private static void TestSubmitPortfolio(EnergyMarketMUI proxy)
{
    // The XML structure for this message is as follows:
    // <SubmitPortfolio party="SomeParty">
    //   <!-- 1 or more -->
    //   <Portfolio name="SomeName" type="TheType">
    //     <!-- 1 or more -->
    //     <Resource name="SomeResource" resourceID="TheID"/>
    //   </Portfolio>
    // </SubmitPortfolio>

    // Create the containing request object
    SubmitPortfolioRequest request = new SubmitPortfolioRequest();

    // Create the SubmitPortfolio object. This maps to the
    // <SubmitPortfolio> element. Link it to the request object.
    SubmitPortfolio submitPortfolio = new SubmitPortfolio();
    submitPortfolio.party = "SomeParty";
    request.SubmitPortfolio = submitPortfolio;

    // Create the portfolio container object. It will contain the
    // a single <Portfolio> element in this sample.
    PortfolioInOut[] portfolioContainer = new PortfolioInOut[1];
    submitPortfolio.Portfolio = portfolioContainer;

    // Create the portfolio object. This maps to the <Portfolio>
    // element. Assign it to the portfolio container.
    PortfolioInOut portfolio = new PortfolioInOut();
    portfolio.name = "PortfolioName";
    portfolio.type = PortfolioTypeEnum.Pnode;
    portfolioContainer[0] = portfolio;

    // Create the container for portfolio entries. It will contain
    // a single <Resource> element in this sample.
    PortfolioEntry[] portfolioEntries = new PortfolioEntry[1];

    // Create the PortfolioEntry object. This maps to the <Resource>
    // element.

```

```

PortfolioEntry portfolioEntry = new PortfolioEntry();
portfolioEntry.name = "PnodeName";
portfolioEntry.resourceID = 12345;

// Assign the portfolio entry to the array of entries. Assign
// the array of entries to the Portfolio object.
portfolioEntries[0] = portfolioEntry;
portfolio.Resource = portfolioEntries;

// Here is the request in XML form
Console.WriteLine("SubmitPortfolio XML message:\n");
XmlSerializer x = new XmlSerializer(request.GetType());
x.Serialize(Console.Out, request);

// Submit the request and print out the response
SubmitPortfolioResponse response =
    proxy.SubmitPortfolio(request);
Console.WriteLine("Transaction ID = {0}",
    response.SubmitConfirmation.transactionId);
}

//-----
// Setup the endpoints and certificates and return the proxy object
// for the web service.
//-----
private static EnergyMarketMUI CreateWebServiceClient()
{
    EndpointAddress address =
        new EndpointAddress("https://SomeURL");

    // The client will use a 509 certificate.
    X509Certificate2 clientKey = new X509Certificate2(
        "FullPathToKeyFile", "TheKeyPassword" );
    // And the server must have trusted certificates.
    X509Certificate2 serverCertifications = new X509Certificate2(
        "FullPathToCrtFile" );

    // Establish a proxy for the service
    EnergyMarketMUI proxy = GetService(
        address,
        clientKey,
        serverCertifications
    );
    return proxy;
}

//-----
// Establishes a connection to the specified service. This can also
// be specified in a WCF configuration file.
// See http://msdn.microsoft.com/en-us/library/ms735103.aspx
//-----
private static EnergyMarketMUI GetService(
    EndpointAddress address, X509Certificate2 clientCert,
    X509Certificate2 serverTrust)
{
    // Binding for proxy
    BasicHttpBinding binding = new BasicHttpBinding();

```

```
// set security mode to Transport
binding.Security.Mode = BasicHttpSecurityMode.Transport;

// set credential type to require certificates
binding.Security.Transport.ClientCredentialType =
    HttpClientCredentialType.Certificate;

// create channel using binding and address
ChannelFactory<EnergyMarketMUI> factory =
    new ChannelFactory<EnergyMarketMUI>(binding, address);

// assign client side certificate
factory.Credentials.ClientCertificate.Certificate = clientCert;

// assign server trusted certificates
factory.Credentials.ServiceCertificate.DefaultCertificate =
    serverTrust;

// create proxy connection to the server
return factory.CreateChannel();
}
```

Appendix B. Web Service Client – Java

Below is an example of how to access the eMarket web services using Java. The code below is abbreviated for readability. The complete example and an electronic copy of the sample code is available from ISO New England Customer Support services.

```
package com.areva.sample;

// For reading in properties
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.util.Properties;
import java.util.GregorianCalendar;
import java.util.Date;
import java.util.Calendar;

// Security related classes
import java.security.KeyStore;
import javax.net.ssl.KeyManagerFactory;
import javax.net.ssl.TrustManagerFactory;

// For marshaling to XML
import javax.xml.bind.JAXBContext;
import javax.xml.bind.JAXBException;
import javax.xml.bind.Marshaller;
import javax.xml.datatype.XMLGregorianCalendar;
import javax.xml.datatype.DatatypeFactory;
import javax.xml.datatype.DatatypeConfigurationException;
import javax.xml.datatype.DatatypeConstants;

// Web Service related classes
import org.apache.cxf.configuration.jsse.TLSClientParameters;
import org.apache.cxf.frontend.ClientProxy;
import org.apache.cxf.interceptor.LoggingInInterceptor;
import org.apache.cxf.interceptor.LoggingOutInterceptor;
import org.apache.cxf.jaxws.JaxWsProxyFactoryBean;
import org.apache.cxf.transport.http.HTTPConduit;

// Classes generated from wsdl2java
import com.iso_ne.markets.mui.emkt._0_1.EnergyMarketMUI;
import com.iso_ne.markets.mui.emkt._0_1.MUIFaultException;
import com.iso_ne.markets.mui.emkt.messages.NodeOut;
import com.iso_ne.markets.mui.emkt.messages.GetNode;
import com.iso_ne.markets.mui.emkt.messages.GetNodeResponse;
import com.iso_ne.markets.mui.emkt.messages.PortfolioEntry;
import com.iso_ne.markets.mui.emkt.messages.PortfolioTypeEnum;
import com.iso_ne.markets.mui.emkt.messages.SubmitConfirmation;
import com.iso_ne.markets.mui.emkt.messages.SubmitPortfolio;
import com.iso_ne.markets.mui.emkt.messages.PortfolioInOut;
import com.iso_ne.markets.mui.emkt.messages.TransactionVariance;
import com.iso_ne.markets.mui.emkt.messages.QueryByDayIdAndFuelIndex;
```

```

import com.iso_ne.markets.mui.emkt.messages.GetFuelPriceAdjustmentUnit;
import
com.iso_ne.markets.mui.emkt.messages.GetFuelPriceAdjustmentUnitResponse;
import com.iso_ne.markets.mui.emkt.messages.FuelPriceAdjustmentUnit;
import com.iso_ne.markets.mui.emkt.messages.TransactionVariance;

/**
 * This sample program shows how to interact with eMarket Market User
 * Interface (MUI) using web services. The program relies on the having run
 * the eMktMUI.wsdl through the CXF wsdl2java. See the ant build file to see
 * how this is setup.
 */
public class MUIClientSample {

    //-----
    // Configurable settings are driven from a properties file and held
    // in a local properties object.
    //-----

    private static String PROPERTIES_PATH = "./MUISample.properties";
    private static Properties properties;

    /**
     * @param args
     */
    public static void main(String[] args) {

        properties = getPropertiesFromFile(PROPERTIES_PATH);

        try{

            // Setup the proxy to the web service.
            EnergyMarketMUI webService = createWebServiceClient();

            // Try a simple Query and Submit type operation.
            testGetNode(webService);
            testSubmitPortfolio(webService);
            testFPA(webService);
        }
        catch(MUIFaultException soapFault){
            System.err.println("\n*** Operation produced a Fault
***\n");
            // display details of soap message
            for (TransactionVariance tv :
soapFault.getFaultInfo().getError())
                System.err.printf("ErrorMessage: %s", tv.getReason());
        }
        catch(Exception e){
            System.err.println("\n*** Operation produced a Fault
***\n");
            System.err.println(e.toString());
        }

        System.exit(0);
    }
}

```

```

/**
 * Executes a "GetNode" operation and prints out the results.
 * @param webService - the service to be used
 */
private static void testGetNode(EnergyMarketMUI proxy) throws
MUIFaultException
{
    System.out.println("\n-----");
    System.out.println("  Testing GetNode");
    System.out.println("\n-----\n");

    // The XML structure for this message is as follows:
    // <GetNode party="SomeParty"/>

    // Create the GetNode object. This maps to the <GetNode> element.
    GetNode request = new GetNode();
    request.setParty(properties.getProperty("client.party"));

    // Call the service and print out the response.
    GetNodeResponse response = proxy.getNode(request);
    int nodeCount = response.getNode().size();
    System.out.print("\nNodes returned: " + nodeCount);
    System.out.println(". Here are the first 10:\n");
    int loopCount = 0;
    for (NodeOut node : response.getNode())
    {
        System.out.printf("Node ID = %-8s Name = %-25s Type = %s\n",
node.getID(), node.getName(), node.getType());
        loopCount++;
        if (loopCount == 10)
            break;
    }
}

/**
 * Executes a "SubmitPortfolio" operation and prints out the results.
 * @param webService - the service to be used
 */
private static void testSubmitPortfolio(EnergyMarketMUI proxy) throws
MUIFaultException
{
    System.out.println("\n-----");
    System.out.println("  SubmitPortfolio");
    System.out.println("\n-----\n");

    // The XML structure for this message is as follows:
    // <SubmitPortfolio party="SomeParty">
    // <!-- 1 or more -->
    // <Portfolio name="SomeName" type="TheType">
    // <!-- 1 or more -->
    // <Resource name="SomeResource" resourceID="TheID"/>
    // </Portfolio>
    // </SubmitPortfolio>

```

```

// Create the SubmitPortfolio object. This maps to the
// <SubmitPortfolio> element.
SubmitPortfolio submitPortfolio = new SubmitPortfolio();
submitPortfolio.setParty(properties.getProperty("client.party"));

// Create the portfolio object. This maps to the <Portfolio>
// element. Assign to the submitPortfolio object.
PortfolioInOut portfolio = new PortfolioInOut();
portfolio.setName(properties.getProperty("portfolio.name"));
portfolio.setType(PortfolioTypeEnum.PNODE);
submitPortfolio.getPortfolio().add(portfolio);

// Create the PortfolioEntry object. This maps to the <Resource>
// element. Assign to the portfolio object.
PortfolioEntry portfolioEntry = new PortfolioEntry();
portfolioEntry.setName("LD.STEPSTON23");
Long id = new Long(16184);
portfolioEntry.setResourceID(id);
portfolio.getResource().add(portfolioEntry);

// Here is the request in XML form
System.out.println("Calling SubmitPortfolio with this message:\n");
try
{
    JAXBContext jc =
JAXBContext.newInstance("com.iso_ne.markets.mui.emkt.messages");
    Marshaller m = jc.createMarshaller();
    m.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT,
Boolean.TRUE);
    m.marshal( submitPortfolio, System.out );
} catch (JAXBException e) {
    e.printStackTrace();
}

// Submit the request and print out the response
SubmitConfirmation response =
proxy.submitPortfolio(submitPortfolio);
System.out.println("Transaction ID = " +
response.getTransactionId());

}
/**
 * Executes a "SubmitPortfolio" operation and prints out the
results.
 * @param webService - the service to be used
 */
private static void testFPA(EnergyMarketMUI proxy) throws
MUIFaultException
{
    System.out.println("\n-----
-----");
    System.out.println("  Get Fuel Price Adjustment Unit
for the given date");
    System.out.println("\n-----
-----\n");

```

```

// The XML structure for this message is as follows:
//<GetFuelPriceAdjustmentUnit party="Genon Energy
Management_ LLC" xmlns="http://www.markets.iso-ne.com/MUI/eMkt/Messages">
// <QueryFilters>
// <Day>2014-07-08</Day>
// <FuelIndex />
// </QueryFilters>
// </GetFuelPriceAdjustmentUnit>

// Create the FPA object. This maps to the
// <GetFuelPriceAdjustmentUnit> element.
try
{
GregorianCalendar gregorianCalendar = new
GregorianCalendar();

Calendar calendar = Calendar.getInstance();
calendar.setTime(new Date());
calendar.add(Calendar.DATE, 1);

gregorianCalendar.setTime(calendar.getTime());

XMLGregorianCalendar
xmlGregorianCalendar=DatatypeFactory.newInstance().newXMLGregorianCalendar(gr
egorianCalendar);

xmlGregorianCalendar.setTimezone(DatatypeConstants.FIELD_UNDEFINED);

QueryByDayIdAndFuelIndex
queryByDayIdAndFuelIndex=new QueryByDayIdAndFuelIndex();

queryByDayIdAndFuelIndex.setDay(xmlGregorianCalendar);
GetFuelPriceAdjustmentUnit
getFuelPriceAdjustmentUnit=new GetFuelPriceAdjustmentUnit();

getFuelPriceAdjustmentUnit.setQueryFilters(queryByDayIdAndFuelIndex);

getFuelPriceAdjustmentUnit.setParty(properties.getProperty("client.part
y"));

// Call the service and print out the response.
GetFuelPriceAdjustmentUnitResponse
getFuelPriceAdjustmentUnitResponse =
proxy.getFuelPriceAdjustmentUnit(getFuelPriceAdjustmentUnit);
int fuelPriceAdjustmentUnitCount =
getFuelPriceAdjustmentUnitResponse.getFuelPriceAdjustmentUnit().size();
System.out.print("\n FuelPriceAdjustmentUnit
Returned: " + fuelPriceAdjustmentUnitCount);
System.out.println(". Here are the first
10:\n");

int loopCount = 0;
for (FuelPriceAdjustmentUnit
fuelPriceAdjustmentUnit :
getFuelPriceAdjustmentUnitResponse.getFuelPriceAdjustmentUnit())
{

```

```

        System.out.printf("Unit Long Name = %-25s
Day = %-25s ID = %s\n", fuelPriceAdjustmentUnit.getUnitLongName(),
fuelPriceAdjustmentUnit.getDay(), fuelPriceAdjustmentUnit.getID());
        loopCount++;
        if (loopCount == 10)
            break;
    }
}
catch (DatatypeConfigurationException ex)
{
    ex.printStackTrace();
}
}

/** Create the CXF webservice Proxy.
 * @return webService used throughout the project
 */
private static EnergyMarketMUI createWebServiceClient(){
    // factory for creating interface
    JaxWsProxyFactoryBean factory = new JaxWsProxyFactoryBean();

    // logging for in and out. Add these if you want more info spit
out
    factory.getInInterceptors().add(new LoggingInInterceptor());
    factory.getOutInterceptors().add(new LoggingOutInterceptor());

    // add interface and end point
    factory.setServiceClass(EnergyMarketMUI.class);
    factory.setAddress(properties.getProperty("endpoint.link"));

    // instance of our web service message options
    EnergyMarketMUI webService = (EnergyMarketMUI) factory.create();

    // add certificate connectivity properties
    configureSSLOnTheClient(webService);

    return webService;
}

/** This method configures our transport layer to use <i>two way</i>
secure HTTP
 * To do this we're configuring the certificates we trust as well as
the certificate
 * that we are.
 * @param c - the CXF Proxy that we want to wrap with SSL
 */
private static void configureSSLOnTheClient(Object c){
    // end point obtained
    org.apache.cxf.endpoint.Client client = ClientProxy.getClient(c);

    // conduit we're using to transfer data
    HTTPConduit httpConduit = (HTTPConduit)client.getConduit();

    try{
        // holds parameters for configuration
        TLSClientParameters tlsParams = new TLSClientParameters();

```

```

tlsParams.setDisableCNCheck(true);

// format type for server side certificates
KeyStore trustStore = KeyStore.getInstance(
    properties.getProperty("srvr.trust.fmt")
);

// location of server trusted list of certificates
File serverTrusts = new File(
    properties.getProperty("srvr.trust.kstr")
);

// add trusted server certificates to store
String propValue = properties.getProperty("srvr.trust.pswd");
trustStore.load(new FileInputStream(serverTrusts),
    propValue.toCharArray()
);

// trust manager generated
TrustManagerFactory trustFactory =
    TrustManagerFactory.getInstance(
        TrustManagerFactory.getDefaultAlgorithm()
    );// uses default Algorithm

// initialize trust manager with the file acquired
trustFactory.init(trustStore);

// this configures our client to trust these certificates

tlsParams.setTrustManagers(trustFactory.getTrustManagers());

// keys for client side
KeyStore keyStore = KeyStore.getInstance(
    properties.getProperty("client.key.fmt")
);

// location of client key
File clientKey = new File(
    properties.getProperty("client.key.file")
);

// load certificate file into key store using password
propValue = properties.getProperty("client.key.pswd");
keyStore.load(new FileInputStream(clientKey),
    propValue.toCharArray()
);

// manager for keys
KeyManagerFactory keyFactory =
    KeyManagerFactory.getInstance(
        KeyManagerFactory.getDefaultAlgorithm()
    );// uses default Algorithm

// authorize use of key
propValue = properties.getProperty("client.key.pswd");
keyFactory.init(keyStore,
    propValue.toCharArray()
);

```

```

        );

        // this sets our client to use the configured key and
certificate as our credentials
        tlsParams.setKeyManagers(keyFactory.getKeyManagers());

        // add parameters to conduit
        httpConduit.setTlsClientParameters(tlsParams);

        // exceptions are caught and dealt with the same way
    } catch (Exception e) {
        System.out.println("Security configuration failed with the
following: ");
        e.printStackTrace();
        System.out.println("-----");
    } // end catch
} // end

/** Retrieves properties from a file for the use of this program.
 * @param path - a path to the properties file
 * @return properties - the properties to be used
 */
private static Properties getPropertiesFromFile(String path){
    Properties properties = new Properties();
    File file = new File(path);

    // attempt to read the file and absorb the properties contained
within
    try {
        FileInputStream fileReader = new FileInputStream(file);
        properties.load(fileReader);
    } catch (FileNotFoundException e) {
        // print errors
        e.printStackTrace();
        System.exit(1);
    } catch (IOException e) {
        // print errors
        e.printStackTrace();
        System.exit(1);
    }

    return properties;
}

} // end class MUIClientSample

```