APX MarketSuite® ISO Scheduling API - xMarket

Version 0.1



Table of Contents

[1. Revision History 3](#_Toc26320016)

[2. Summary 4](#_Toc26320017)

[2.1 ISO/Market Scope 4](#_Toc26320018)

[3. Data Transfer 5](#_Toc26320019)

[3.1 Data Lifecycle 5](#_Toc26320020)

[4. File Structure and XSD 6](#_Toc26320021)

[4.1.1 Scope 6](#_Toc26320022)

[4.1.2 Inbound Data Variances 6](#_Toc26320023)

[4.1.3 Outbound Data Variances 6](#_Toc26320024)

[4.2 Inbound Data 7](#_Toc26320025)

[4.2.1 Example Submission for one bid hour 7](#_Toc26320026)

[4.2.2 ‘Submit’ root element 7](#_Toc26320027)

[4.3 Outbound Data 8](#_Toc26320028)

[4.3.1 Example Response for One Bid Hour (GetScheduleData method) 8](#_Toc26320029)

[4.3.2 ‘Response’ root element 9](#_Toc26320030)

[5. MarketBidData 10](#_Toc26320031)

[5.1 MarketBidData Complex Type 10](#_Toc26320032)

[5.2 BidsOffers 10](#_Toc26320033)

[5.2.1 SelfSchedule 11](#_Toc26320034)

[5.2.2 MarketSchedule 11](#_Toc26320035)

[5.3 ResourceParameters 12](#_Toc26320036)

[5.3.1 ResourceParameters Value 12](#_Toc26320037)

[6. MarketTradeData 13](#_Toc26320038)

[6.1 MarketTradeData Complex Type 13](#_Toc26320039)

[6.2 BilateralSchedules 13](#_Toc26320040)

[6.3 BilateralScheduleDetail 13](#_Toc26320041)

[7. ISO-Specific Transactions, Parameters, and Bilateral Schedules 15](#_Toc26320042)

[7.1 PJM 15](#_Toc26320043)

[7.1.1 BidsOffers 15](#_Toc26320044)

[7.1.2 ResourceParameters 15](#_Toc26320045)

[7.1.3 BilateralSchedules 16](#_Toc26320046)

[7.2 MISO 16](#_Toc26320047)

[7.2.1 BidsOffers 16](#_Toc26320048)

[7.2.2 ResourceParameters 17](#_Toc26320049)

[7.2.3 BilateralSchedules 17](#_Toc26320050)

[7.3 SPP 17](#_Toc26320051)

[7.3.1 BidsOffers 17](#_Toc26320052)

[7.3.2 ResourceParameters 18](#_Toc26320053)

[7.3.3 BilateralSchedules 19](#_Toc26320054)

[7.4 ISONE 19](#_Toc26320055)

[7.4.1 BidsOffers 19](#_Toc26320056)

[7.4.2 ResourceParameters 20](#_Toc26320057)

[7.4.3 BilateralSchedules 20](#_Toc26320058)

[7.5 NYISO 20](#_Toc26320059)

[7.5.1 BidsOffers 20](#_Toc26320060)

[7.5.2 ResourceParameters 21](#_Toc26320061)

[7.5.3 BilateralSchedules 21](#_Toc26320062)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Description | Updated By |
| 0.1 | 12/3/2019 | * Initial Version | Adam Barrett |
| 1.0 | 6/4/2020 | * Allowing multiple days and participants for all scheduling data submissions * Market awards available. | Adam Barrett |

# Summary

The purpose of this document is to describe the data structures and processes for inbound and outbound Independent System Operator (ISO) scheduling-related data for the APX MarketSuite® for the following ISOs: PJM, MISO, SPP, ISONE, and NYISO.

This data includes ISO wholesale scheduling transactions such as generator bids, load schedules, generator parameters (such as operating limits) and bilateral schedules.

This document will describe the content of the files exchanged between the APX MarketSuite® and external users’ systems.

For details about the specific API methods and who they are called, refer to the document ‘APX Technical Interface Specification – REST APIs.’

## ISO/Market Scope

The document is designed to support functionality for markets PJM, MISO, SPP, ISONE, and NYISO. Region-specific details are highlighted throughout the document.

The data structures for the CAISO and ERCOT ISOs are similar, and can be found in a separate document.

# Data Transfer

## Data Lifecycle

Clients may submit data into the APX MarketSuite® from either the UI or the API. In both cases, all the data in the system is visible through the UI and can be submitted to the ISO via the Checkout application. In the case of API submissions, a flag can be set that will result in the data being submitted immediately to the ISO without requiring any user intervention through the UI.

Data statuses are managed in a variety of ways as supported by the regional ISO. For example, some markets perform a credit check after initial submission, and the APX system periodically checks the status of the submissions by querying the ISO systems.

When submitting data via the API through the Upload File method, clients will receive a file handle (a character string) that represents the file that was uploaded. The uploaded data may also use the external\_id field to make it easier to identify and track data being sent to APX MarketSuite®, or to tag it at a level that is more granular than a file. The file handle may be used to query for exception data that was submitted via the API (i.e. data that could not be processed).

Clients may use the GetScheduleData method to obtain the most current schedule data that exists on the APX system.

# File Structure and XSD

Scheduling data can be submitted to the APX MarketSuite® through Inbound file upload, and can be queried from the system with the Outbound GetStatus (exception data) and GetScheduleData methods.

Both the inbound and outbound web-services utilize the same ScheduleData.xsd to form and validate the format of the XML messages, with some variances:

### Scope

Many organizations may represent multiple ‘participants’ within a single market. For this reason, the XML schema allows BidsOffers, ResourceParameters and BilateralSchedules for multiple participants to be submitted (Inbound) and pulled (Outbound) in one file.

Users may also upload up to seven days worth of data in a single submission.

### Inbound Data Variances

The inbound data is similar to the outbound schema with the following exceptions:

* The root element is named ‘Submit’
* The ‘Status’ and ‘Message’ elements are not used for inbound data.
* The ‘FileStatus’ attribute is not used for inbound data.
* The ‘CounterPartyMW’ attribute of the BilateralScheduleDetail element is not used for inbound data.
* The ‘Price’ attribute of the ‘SelfScheduleType’ element is not used for pre-market data, but may be present for outbound post-market awards.

### Outbound Data Variances

The data of the response looks very similar to submitted data with the following differences:

* The root element is named ‘Response’
* The ‘SubmitToISO’ attribute in the Response element is ignored.
* The ‘Status’ and ‘Message’ elements are added throughout to return the status of the row and optionally a message that contains error or informational message regarding the status of that row.
* The ‘FileStatus’ attribute is provided in GetStatus responses to convey the presence or absence of any exception data associated with the submission. This attribute is not used in GetScheduleData responses.
* The ‘CounterPartyMW’ attribute is added to the BilateralScheduleDetail element
* The ‘Price’ attribute is added to the ‘SelfScheduleType’ element for post-market awards.

## Inbound Data

The following section details how the API may be used to submit data to APX MarketSuite® through the Upload File method. Inbound XML files will have a ‘Submit’ root element, plus the ‘MarketBidData’ or ‘MarketTradeData’ complex type. MarketBidData includes BidsOffers and ResourceParameters, and MarketTradeData includes BilateralSchedules. ‘MarketBidData’ and ‘MarketTradeData’ may not be submitted in the same file.

### Example Submission for one bid hour

<Submit Version="2.1" SourceSystem="APX" CreateDate="2019-10-26T14:39:12Z" Region="PJM" SubmitToISO="true" xmlns="http://service.apx.com/ScheduleData">

<MarketBidData Date="2019-12-06">

<BidsOffers MarketParticipant="ACME" Location="DPL" Transaction="DA Fixed Demand Bid">

<SelfSchedule>

<Schedule MW="10" IntervalEndGmt="2019-12-06T06:00:00Z"></Schedule>

</SelfSchedule>

</BidsOffers>

</MarketBidData>

</Submit>

The specifics of the ‘MarketBidData’ and ‘MarketTradeData’ complex types are the same in the inbound and outbound files, and are described later in this document.

### Inbound Data Validation

The different types of inbound data are validated differently. Inbound BidsOffers data undergoes two phases of validation:

Phase 1 Validation

The first phase of validation is for file format and syntax (xsd validation). If the file fails this validation, the entire file will be rejected and no data will be saved.

Phase 2 Validation

The second phase validation is for business rules. Data failing phase 2 validation can only be returned by the Get Status method. Reasons for failing phase 2 validation include:

* Invalid Transaction name
* Invalid location name
* Location not valid for transaction

For ResourceParameters and BilateralSchedules, all validations are in one phase, and any failure will result in the entire file being rejected. Note that APX may introduce two-phase validation for these data in the future.

### Root ‘Submit’

The ‘Submit’ section is the header for all submissions regardless of the dataset being passed in the XML file.

|  |  |  |
| --- | --- | --- |
| Header Attribute | Description | Data Type |
| Version | Version of API (XSD Optional) (Default is current version) | String |
| SourceSystem | Represents the system that produced the data (XSD Required) | String |
| CreateDate | Date and time that the data was pulled from the source system (XSD Required) | XML dateTime (GMT) |
| Region | ISO Code for the submission | String |
| SubmitToISO | A flag that indicates that the data should be sent automatically to the ISO. ‘False’ means that the bids/offers will need to be reviewed and submitted interactively using the APX MarketSuite® Checkout application or via a subsequent API submission (XSD Optional) (Default is ‘False’). | Boolean |
| xmlns | xsd | http://service.apx.com/  ScheduleData |

## Outbound Data

The following section details how the API may be used to obtain ISO Scheduling data from the APX MarketSuite®. There are two methods of obtaining ISO Scheduling related data:

* GetStatus – allows clients to programmatically obtain the exception (error) data associated with prior submissions by referencing the uploaded file. This method only reports the status of exception data that was submitted by this file; it will not include the status of data submitted in another file or directly the APX MarketSuite® User Interface or the regional ISO User Interface.
* GetScheduleData – allows a client to programmatically obtain:
  + A complete image of their scheduling data including status.
  + Market awards including MW and the applicable market clearing price

Outbound XML files will have a ‘Response’ root element, plus the ‘MarketBidData’ or ‘MarketTradeData’ complex type. MarketBidData includes BidsOffers and ResourceParameters, and MarketTradeData includes BilateralSchedules. ‘MarketBidData’ and ‘MarketTradeData’ may not be requested or returned in the same request.

### Example Response for One Bid Hour (GetScheduleData method)

<Response SourceSystem="APX" CreateDate="2019-12-03T23:03:07.091-08:00" Region="PJM" xmlns="http://service.apx.com/ScheduleData" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<MarketBidData Date="2019-12-06">

<BidsOffers MarketParticipant="ACME" Location="DPL" Transaction="DA Fixed Demand Bid">

<SelfSchedule>

<Schedule MW="10" IntervalEndGmt="2019-12-06T06:00:00Z" Status="APXCreated"/>

</SelfSchedule>

</BidsOffers>

</MarketBidData>

</Response>

The specifics of the ‘MarketBidData’ and ‘MarketTradeData’ complex types are the same in the inbound and outbound files, and are described later in this document.

### Root ‘Response’

The ‘Response’ section is the header for all outbound data regardless of the data being requested.

|  |  |  |
| --- | --- | --- |
| Header Attribute | Description | Data Type |
| SourceSystem | Represents the system that produced the data (XSD Required) | String |
| CreateDate | Date and time that the data was pulled from the source system (XSD Required) | XML dateTime (GMT) |
| Region | ISO Code for the submission | String |
| FileStatus | Get Status method only. Conveys the header-level status of the results. Possible responses include:   * ‘SUCCESS: No exception data found’ * ‘SUCCESS’ [File will include details on exceptions for the submission] * ‘ERROR: The file is not a valid Scheduling file’ * 'File load in progress' | String |
| xmlns | xsd | http://service.apx.com/  ScheduleData |

# MarketBidData

This section will explain the contents of the MarketBidData complex type. This element may contain BidsOffers and/or ResourceParameters data, which are each also a complex type.

## MarketBidData Complex Type

The parent element and attributes.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| MarketBidData | Container for BidsOffers and Resource Parameters | N | Complex type |
| Date | First trade date of submitted data. | Y | Date |
| EndDate | Last trade date of submitted data | N | Date |
| BidsOffers | See section 5.2 | N | Complex type |
| ResourceParameters | See section 5.3 | N | Complex type |

## BidsOffers

The BidsOffers complex type may contain two types a bids: SelfSchedule and MarketSchedule. The ISO-specific types of transactions that may be scheduled under each of these are described later in this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| BidsOffers | Container for interval level sub-data. Attributes for this element include business keys and other data that apply to all child intervals. | N | Complex type |
| MarketParticipant | ISO scheduling code of the participant. | Y | String |
| Transaction | Name of the transaction being scheduled. The ISO-specific transactions that may be scheduled are described later in this document. | Y | String |
| Location | Name of the ISO location being scheduled (this should be the readable name, not the ISO location code) | Y | String |
| ReferenceCode | Unique identifier for the bid. Required for some transactions, such as PJM Generator market bids | N (required for some transactions) | String |

### SelfSchedule

Depending on the Transaction in the parent BidsOffers, the SelfSchedule complex type may be required for the interval-level data.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| SelfSchedule | Interval-level schedule data | N | Complex Type |
| MW | MW of the interval schedule.  To cancel a previously scheduled transaction for the interval, omit this attribute. | N | Decimal |
| IntervalEndGMT | Interval End time for the schedule, in GMT | Y | XML dateTime |
| Status | (Outbound Only) The status associated with the Transaction/Interval | N | String |
| Message | (Outbound Only) The status message associated with the Transaction/Interval | N | String |

### MarketSchedule

Depending on the Transaction selected in BidsOffers, the SelfSchedule complex type may be required for the interval-level data.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| MarketSchedule | Market schedule data | N | Complex type |
| Curve | Interval-level market schedule bid curve | Y | Complex Type |
| CurveType | Type of curve. May depend on ISO rules for the transaction, but possible values include:   * Block * Slope * Fixed | Y (Not required for single-point Bids, e.g. PJM UpToCongestion) | String |
| IntervalEndGMT | Interval End time for the schedule, in GMT | Y | XML dateTime |
| Status | (Outbound Only) The status associated with the Transaction/Interval | N | String |
| Message | (Outbound Only) The status message associated with the Transaction/Interval | N | String |
| CurvePoint | Price/MW pairs. May be repeated for each point in a bid curve.  To cancel a previously scheduled transaction for the interval, omit this element. | N | Element |
| MW | MW for the curve point | Y | Decimal |
| Price | Price for the curve point | Y | Decimal |

## ResourceParameters

The ResourceParameters complex type is used to submit operating parameters, such as MW limits, ramp rates, etc. The ISO-specific types of transactions that may be scheduled under each of these are described later in this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| ResourceParameters | Container for interval level sub-data. Attributes for this element include business keys and other data that apply to all child intervals. | N | Complex type |
| MarketParticipant | ISO scheduling code of the participant. | Y | String |
| Location | Name of the ISO location being scheduled (this should be the readable name, not the ISO location code) | Y | String |
| ParameterType | Name of the parameter being scheduled. The ISO-specific parameters that may be are described later in this document. | Y | String |
| ReferenceCode | Unique identifier for the bid. Required for some transactions, such as PJM Schedule parameters. | N | String |

### ResourceParameters Value

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| Value | Interval-level schedule for parameters | N | Complex type |
| IntervalEndGMT | Interval End time for the parameter, in GMT | Y | Datetime |
| Value | Parameter value | N | String |
| Status | (Outbound Only) The status associated with the Parameter/Interval | N | String |
| Message | (Outbound Only) The status message associated with the Parameter/Interval | N | String |

# MarketTradeData

This section will explain the contents of the MarketTradeData complex type. This element may only contain BilateralSchedules.

## MarketTradeData Complex Type

The parent element and attributes.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| MarketTradeData | Container for BilateralSchedules | N | Complex type |
| Date | First trade date of submitted data. | Y | Date |
| EndDate | Last trade date of submitted data | N | Date |
| BilateralSchedules | See section 6.2 | Y | Complex type |

## BilateralSchedules

Parent for each transaction.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| BilateralSchedules | Bilateral schedule data | Y | Complex type |
| MarketParticipant | ISO scheduling code of the participant. | Y | String |
| Location | Name of the ISO location being scheduled (this should be the readable name, not the ISO location code) | Y | String |
| SinkLocation | Some bilaterals require a sink location (this should be the readable name, not the ISO location code) | N | String |
| ReferenceCode | ISO code for the Contract | Y | String |

## BilateralScheduleDetail

Interval-level detail for bilateral schedules.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Required | Type |
| BilateralScheduleDetail | Interval-level schedule for bilateral schedules | Y | Complex type |
| IntervalEndGMT | Interval End time for the parameter, in GMT | Y | Datetime |
| MW | MW value for the interval schedule.  To cancel the BilateralSchedule for this interval, exclude this attribute. | N | Decimal |
| CounterPartyMW | (Outbound Only) The MW quantity submitted by the counterparty associated with the Parameter/Interval, if available | N | Decimal |
| Status | (Outbound Only) The status associated with the Parameter/Interval | N | String |
| Message | (Outbound Only) The status message associated with the Parameter/Interval | N | String |

# ISO-Specific Transactions, Parameters, and Bilateral Schedules

This section will identify the specific ISO transactions, Resource Paramaters, and Bilateral Schedules supported under this API, along with the available attributes or other information necessary to successfully submit the data.

## PJM

### BidsOffers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transaction | Description | Schedule Type | Reference Code | Attributes | Notes |
| ***DA Gen Energy Market*** | Generator Energy Market Bid for DAM. Minimum one, max ten Price/MW rows | MarketSchedule | Numeric PJM Schedule ID, e.g. ‘1’ or ‘99’ - required | ‘CurveType’ (Block or Slope) |  |
| ***RT Gen Energy Market*** | Generator Energy Market Bid for RTBM. Minimum one, max ten Price/MW rows | MarketSchedule | Numeric PJM Schedule ID, e.g. ‘1’ or ‘99’ - required | ‘CurveType’ (Block or Slope) |  |
| ***DA Decrement Bid*** | DAM Virtual Bid | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |
| ***DA Increment Offer*** | DAM Virtual Offer | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |
| ***DA Fixed Demand Bid*** | Load Energy value in the DAM. | SelfSchedule | N/A |  |  |
| ***DA Price Sensitive Demand Bid*** | Load Energy Market Bid for DAM. Minimum one, max ten Price/MW rows | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |
| ***DA SourceSink Congestion Market*** | Known in PJM as “UpToCongestion” bids. Single price/MW pair with SourceLocation, SinkLocation | MarketSchedule | Required.  This is used as the PJM TransactionID. Can be int or decimal. |  |  |

### ResourceParameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Data Type/format | Description | ParameterType | Notes |
| ***Commitment Status*** | String | Commitment status of the generator. Available values:   * Unavailable * Economic * Emergency * MustRun | Value |  |
| ***Economic Max MW*** | Decimal | Optional element specifying the economic maximum limit for the hourly interval. | Value |  |
| ***Economic Min MW*** | Decimal | Optional element specifying the economic minimum limit for the hourly interval. | Value |  |
| ***Emergency Max MW*** | Decimal | Optional element specifying the emergency maximum limit for the hourly interval. | Value |  |
| ***Emergency Min MW*** | Decimal | Optional element specifying the emergency minimum limit for the hourly interval. | Value |  |
| ***Fixed Gen*** | boolean | Optional element specifying the fixed generation indicator for the unit for this hour. Must be ‘true’ or ‘false’ | Value |  |

### BilateralSchedules

|  |  |  |
| --- | --- | --- |
| Transaction | Description | Notes |
| ***DA Sell Energy IBT***  ***DA Buy Energy IBT*** | Day-Ahead energy bilateral |  |
| ***RT Sell Energy IBT***  ***RT Buy Energy IBT*** | Real-Time energy bilateral |  |
| ***RT Sell Energy RLR***  ***RT Buy Energy RLR*** | Real-Time Retail Load Responsibility bilateral |  |
| ***RT Sell Energy WLR***  ***RT Buy Energy WLR*** | Real-Time Wholesale Load Responsibility bilateral |  |

## MISO

### BidsOffers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transaction | Description | Schedule Type | Reference Code | Available Attributes | Notes |
| ***DA Load Energy Self*** | Load Energy value in the DAM. | SelfSchedule | N/A |  |  |
| ***DA Load Energy Market*** | Load Energy Market Bid for DAM. Minimum one, max ten Price/MW rows | MarketSchedule | N/A |  |  |
| ***DA Virtual Bid*** | Generator Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block) |  |
| ***DA Virtual Offer*** | Generator Non-Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block) |  |

### ResourceParameters

### BilateralSchedules

|  |  |  |
| --- | --- | --- |
| Transaction | Description | Notes |
| ***DA Sell Energy FinSchedule***  ***DA Buy Energy FinSchedule*** | Day-Ahead energy bilateral |  |
| ***RT Sell Energy FinSchedule***  ***RT Buy Energy FinSchedule*** | Real-Time energy bilateral |  |

## SPP

### BidsOffers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transaction | Description | Schedule Type | Reference Code | Available Attributes | Notes |
| ***DA Gen Energy Market*** | Generator Energy Market Bid for DAM. Minimum one, max ten Price/MW rows | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |
| ***RT Gen Energy Market*** | Generator Energy Market Bid for RTBM. Minimum one, max ten Price/MW rows | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |
| ***DA Virtual Bid*** | Generator Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |
| ***DA Virtual Offer*** | Generator Non-Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block or Slope) |  |

### ResourceParameters

The following parameters are supported in SPP. These are all part of the ‘Energy Operating Limits’ template.

Notes:

* Parameters must be submitted for a specific market stage—DAM or RTBM. As such, each parameter has a “-DA” or “-RT’ suffix to indicate the market.
* Only submit the parameters that you wish to change. The APX MarketSuite® will take existing value for non-submitted parameters and use those for the submission to SPP. The APX MarketSuite® regularly pulls down data from the SPP portal, so should have the latest info.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Data Type/format | Description | ParameterType | Notes |
| ***MaxEconomicLimit-DA MaxEconomicLimit-RT*** | Decimal | An economic MW level at or below a Resource’s Maximum Normal Capacity Operating Limit used for constraining Energy dispatch and Contingency Reserve clearing during normal system conditions. | Value |  |
| ***MaxEmergencyLimit-DA***  ***MaxEmergencyLimit-RT*** | Decimal | The maximum MW level at which a Resource other than a Block Demand Response Resource may operate under Emergency system conditions. | Value |  |
| ***MaxEmergencyRuntime-DA MaxEmergencyRuntime-RT*** | string (hh:mm) | The maximum length of time a Resource can operate above its Maximum Normal Capacity Operating Limit up to its Maximum Emergency Capacity Operating Limit. | Value |  |
| ***MaxNormalLimit-DA***  ***MaxNormalLimit-RT*** | Decimal | The maximum MW level at which a Resource may operate continuously. | Value |  |
| ***MaxRegulationLimit-DA***  ***MaxRegulationLimit-RT*** | Decimal | The maximum MW level at which a Regulation Qualified Resource, a Regulation-Up Qualified Resource or a Regulation-Down Qualified Resource may operate while providing Regulation Deployment. | Value |  |
| ***MinEconomicLimit-DA***  ***MinEconomicLimit-RT*** | Decimal | A MW level at or above a Resource’s Minimum Normal Capacity Operating Limit used for energy dispatch at a minimum level during normal operating conditions. | Value |  |
| ***MinEmergencyLimit-DA***  ***MinEmergencyLimit-RT*** | Decimal | The minimum MW level at which a Resource other than a Block Demand Response Resource may operate under Emergency system conditions. | Value |  |
| ***MinEmergencyRuntime-DA MinEmergencyRuntime-RT*** | string (hh:mm) | The maximum length of time a Resource can operate below its Minimum Normal Capacity Operating Limit down to its Minimum Emergency Capacity Operating Limit. | Value |  |
| ***MinNormalLimit-DA MinNormalLimit-RT*** | Decimal | The minimum MW level at which a Resource may operate continuously. | Value |  |
| ***MinRegulationLimit-DA MinRegulationLimit-RT*** | Decimal | The minimum MW level at which a Regulation Qualified Resource, a Regulation-Up Qualified Resource or a Regulation-Down Qualified Resource may operate while providing Regulation Deployment. | Value |  |
| ***MQSRLimit-DA MQSRLimit-RT*** | Decimal | The maximum amount of Supplemental Reserve that can be provided by a Quick-Start Resource from an off-line state. | Value |  |
| ***TARRFactor-DA TARRFactor-RT*** | Decimal | Turn-Around Ramp Rate Factor | Value |  |

### BilateralSchedules

## ISONE

### BidsOffers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transaction | Description | Schedule Type | Reference Code | Available Attributes | Notes |
| ***DA Load Energy Self*** | Load Energy value in the DAM. | SelfSchedule | N/A |  |  |
| ***DA Load Energy Market*** | Load Energy Market Bid for DAM. Minimum one, max ten Price/MW rows | MarketSchedule | N/A | ‘CurveType’ (Block) |  |
| ***DA Virtual Bid*** | Generator Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block) |  |
| ***DA Virtual Offer*** | Generator Non-Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block) |  |

### ResourceParameters

### BilateralSchedules

|  |  |  |
| --- | --- | --- |
| Transaction | Description | Notes |
| ***DA Sell Energy IBT***  ***DA Buy Energy IBT*** | Day-Ahead energy bilateral |  |
| ***RT Sell Energy IBT***  ***RT Buy Energy IBT*** | Real-Time energy bilateral |  |
| ***RT Sell LoadResp IBT***  ***RT Buy LoadResp IBT*** | Real-Time Retail Load Responsibility bilateral |  |

## NYISO

### BidsOffers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transaction | Description | Schedule Type | Reference Code | Available Attributes | Notes |
| ***DA Load Energy Self*** | Load Energy self-Schedule in the DAM. | SelfSchedule | N/A |  |  |
| ***DA Load Energy Forecast*** | Load Energy forecast in the DAM. | SelfSchedule |  |  |  |
| ***DA Load Energy Market*** | Load Energy Market Bid for DAM. Minimum one, max ten Price/MW rows | MarketSchedule | N/A |  |  |
| ***DA Virtual Bid*** | Generator Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block) |  |
| ***DA Virtual Offer*** | Generator Non-Spin Market Bid. One row with values for Price and MW | MarketSchedule | N/A | ‘CurveType’ (Block) |  |

### ResourceParameters

### BilateralSchedules