AEMO
AUSTRALIAN ENERGY MARKET OPERATOR

Guide to API Energy, FCAS, and MNSP Bids and Offers

1.00 Final

November 2020

Provides assistance to developers and users of the Bidding API

Important Notice

PURPOSE

This Guide to API Energy, FCAS, and MNSP Bids and Offers (Guide), prepared by AEMO, provides guidance for Dispatch Bidding API under the National National Electricity Rules (Rules).

NO RELIANCE OR WARRANTY

This document does not constitute legal or business advice, and should not be relied on as a substitute for obtaining detailed advice about the National Gas or Electricity Law, the Rules or any other applicable laws, procedures or policies. While AEMO has made every effort to ensure the quality of the information in this Guide, neither AEMO, nor any of its employees, agents and consultants make any representation or warranty as to the accuracy, reliability, completeness, currency or suitability for particular purposes of that information.

LIMITATION OF LIABILITY

To the maximum extent permitted by law, AEMO and its advisers, consultants and other contributors to this Guide (or their respective associated companies, businesses, partners, directors, officers or employees) are not liable (whether by reason of negligence or otherwise) for any errors, omissions, defects or misrepresentations in this document, or for any loss or damage suffered by persons who use or rely on the information in it.

TRADEMARK NOTICES

Microsoft is a trademark of Microsoft Corporation in the United States and/or other countries.
Oracle and Java are registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.
© 2015 Google Inc, used with permission. Google and the Google logo are registered trademarks of Google Inc
postmanlabs/postman is licensed under the Apache License 2.0
YAML: Copyright © 2001-2009 Oren Ben-Kiki, Clark Evans, Ingy döt Net
Copyright (c) 2002 JSON.org.
Postman is a trademark of Postman Inc

DISTRIBUTION

Available to the public.

DOCUMENT IDENTIFICATION

Business custodian: Manager, Electricity Market Monitoring and Change
IT custodian: Team Lead, NEM Scheduling and Operations Forecasting Systems
Prepared by: Technical Writers, Digital and Technology

VERSION HISTORY

Version 1.00. Refinements after business feedback.
Last update: Friday, 27 November 2020 5:44 PM

DOCUMENTS MADE OBSOLETE

The release of this document changes any previous versions of Guide to API Energy, FCAS, and MNSP Bids and Offers.

FEEDBACK

Your feedback is important and helps us improve our services and products. To suggest improvements, please contact AEMO's Support Hub.

Contents

Introduction

Purpose

Explains the Dispatch Bidding API and its use.

Audience

This guide is for developers, business analysts and IT staff in participant companies, developing and using the Bidding API and its Endpoints.

A secondary audience is Participant Administrators wanting to know the user rights management (URM) entity to grant for Participant Users to access the API Gateway.

What’s in this guide

Chapter 1 Need to know on page 1 explains what you need to know before you start using Bidding APIs.

Chapter 2 About the Bidding API on page 7 explains the Bidding APIs web application, who it is for, and how to access it, and how to use the common interface features such as selecting a Unit ID, date and so on.

Chapter 3 Bidding API on page 12 explains the Bidding API Endpoints, validation rules, error messages, request, response structures, and examples.

Needing Help on page 1 provides information to assist participants with IT related issues and requesting assistance from AEMO.

Glossary on page 38 explains the terms and abbreviations used throughout this guide.

References on page 44 contains a list of resources mentioned throughout this guide.

How to use this guide

- This guide is written in plain language for easy reading.
- Where there is a discrepancy between the National Electricity Rules, and information or a term in this document, the National Electricity Rules takes precedence.
- Where there is a discrepancy between the Procedures, and information or a term in this document, the Procedures take precedence.
- **Text in this format** indicates there is a resource on AEMO’s website, for details, see References.
- **Text in this format** indicates a link to a related resource.
- Actions to complete in the web portal interface are **bold and dark grey**.
- Rules terms used throughout this guide are capitalised and listed in Rules Terms on page 37. You can find definitions in the National Electricity Rules.
- Glossary terms are capitalised and have the meanings listed against them in the Glossary on page 38.
- References to time are Australian Eastern Standard Time (AEST).

Chapter 1 Need to know

Related rules and procedures

The following rules and procedures relate to Bidding APIs:

Name	Purpose
MNSP Convexity Rule	AEMC website > NER Clause 3.8.6A (e)
Introduction to Market Rules	AEMC website > NER Chapter 3
Market Floor Price	AEMC website > NER 3.9.6
Market Price Cap	AEMC website > NER 3.9.4
Rebidding and Technical Parameters Guideline	**https://www.aer.gov.au/wholesale-markets/market-guidelines-reviews/rebidding-and-technical-parameters-guideline-amendments-for-5-minute-settlement-2019**

Related resources

Please read the following resources with this guide:

You can find resources on AEMO's website.

Guide to AEMO's e-Hub APIs: provides details about using AEMO’s e-Hub as an interface to communicate information with AEMO. It assists Wholesale electricity and gas participants developing their own APIs.

Format and Validation for Energy, FCAS, and MNSP Bids and Offers: Describes the interface to submit file-based Energy, FCAS, and MNSP Dispatch Bids and Offers.

Prerequisites

To use Bidding APIs you must complete the following:

1. Register with AEMO to use APIs.
2. If required, set up your Participant API Gateway. For help, see Participant implementation on page 1.
3. Build the APIs needing implementation at the Participant API Gateway according to the specifications in this guide.
4. Participant Administrators (PA) use the MSATS Web Portal to grant the required Participant User access to the User ID accessing the AEMO APIs. For help, see User rights access on page 1.

Assumed knowledge

This guide assumes you have knowledge of:

- JSON basics
- REST API standards
- Web-based technologies
- The operating system you are using
- Using AEMO's e-Hub
- **Connecting to AEMO’s Electricity IT Systems**

User rights access

Participant administrators provide access to the Bidding API using the following URM entity:

- **EMMS - Offers and Submissions - Energy FCAS MNSP Bids**

API account passwords are reset every 90 days.

Steps to set up URM rights

The steps to set up URM rights for API access are:

1. If required, the Participant Administrator (PA) creates a new Participant User in MSATS.
2. The PA assigns the URM entity to the Participant User.

For help with user rights access, see Guide to User Rights Management.

API e-Hub addresses

You can find API details, including OAS (Swagger) files in the API Portal.

You submit and retrieve APIs using AEMO's and your Participant API Gateway. For more details, see **Guide to AEMO's e-Hub APIs.**

Table 1 API Gateway addresses

Environment	Internet direct base URL	MarketNet base URL
Pre-production API portal base	https://dev.preprod.aemo.com.au	n/a
Production API portal	https://dev.prod.aemo.com.au//guides	n/a
Staging gateway	https://api.5msstaging.aemo.com.au:9319	https://apis.5msstaging.marketnet.net.au:9319/
Pre-production gateway	https://apis.preprod.aemo.com.au:9319	https://apis.preprod.marketnet.net.au:9319
Production gateway	https://apis.prod.aemo.com.au:9319	https://apis.prod.marketnet.net.au:9319

Chapter 2 About the Bidding API

The Dispatch Bidding API allows submission and retrieval of Dispatch Bids/Offers. AEMO's e-Hub provides this API using the current API standards. For information about e-Hub access and using APIs, see **Guide to AEMO's e-Hub APIs**.

This API has the following Endpoints:

1. **submitBids**: Submit one or more Energy, FCAS, or MNSP Bids/Offers .
2. **getBids**: Retrieve one or more Energy, FCAS, or MNSP Bids/Offers. Not including Interval Energy Data.
3. **getBid** Retrieve details for a specific Bid/Offer, including interval, day, and Submission information.
4. **getSubmissions**: Retrieve a list of accepted Submissions for a Participant ID.
5. **getSubmission**: Retrieve a previous Submission for a Participant ID.

Authentication methods

This API uses the following security policies:

- Authorization
- TLS certificate

The TLS certificates used to access pre-production and production environments are different.

Authorization

When calling APIs, you authenticate your identity using Base64 encoding of your username and password.

Your company's participant administrator (PA) provides your username and password. You must encode it into a Base64 authorisation token using an application such as Postman (for help, see https://www.getpostman.com/).

The HTTP Basic authentication header takes the following format:

Authorization: Basic {Base64 hash of user:password}, for example:

```
Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
```

TLS certificate

To obtain a TLS certificate, you must generate a Certificate Signing Request (CSR) that identifies your server. For details, see TLS Certificates in https://dev.preprod.aemo.com.au/tls-certificates#generate-a-csr.

Common header parameters

Parameter	Required	Format
Content-type	Yes	application/json
Content-encoding	Yes	gzip or deflate
Accept-encoding	Yes	gzip or deflate
Accept	Yes	application/json
Authorization	Yes	Base64 encoding of the URM username and password, concatenated with a colon.

Parameter	Required	Format
X-initiatingParticipantID	Yes	Participant ID
X-market	Yes	NEM

Example error response

```
HTTP/1.1 404 Resource Not Found
Content-Length: nnn
Date: Mon, 01 May 2017 18:00:00 GMT
Connection: close
Content-Type: application/json
{
  "transactionId": "GUID",
  "data": {
},
  "errors": [
   {
    "code": 404,
    "title": "Not Found",
    "detail": "Resources for the endpoint URI not found. Endpoint URI:
&lt;Resource&gt;",
    "source": null
    }
  ]
}
```

Common response

Error Code	Description
200	OK
404	Invalid resource used in the URL

Error Code	Description
405	Method Not Allowed
500	No payload

Endpoint format

API Endpoint URLs have the following format:

```
https://<host>/<business_name>/<business_
function>/<APIversion>/<Endpoint>?querystring parameters
```

For example:

```
https://apis.prod.aemo.com.au:9319/NEMWholesale/bidding/v1/getBids
```

Table 2 Definition

Parameter	Description
<protocol>	HTTPS
<host>	Names the server hosting the service or an external proxy For details, see API e-Hub addresses on page 6
<business_name>	A business area such as NEMRetail or NEMWholesale
<business_function>	API Name - The AEMO system providing the services. e.g. bidding

Parameter	Description
<endpoint>	Entities of a Business Function e.g. /getBids
?querystring parameters	Query string parameters for GET method

Submission and throttling

To control the traffic, AEMO implements Throttling on API Endpoint requests. If Participant ID requests exceed the Throttling limit a HTTP 503 response returns.

Endpoint	Limits
submitBids	1 request per second per participantId. The participantId is identified from the X-initiatingParticipantID request header parameter. This restriction is due to an existing legacy limit in the bidding tables in the data model. The bidding tables rely on the field OfferDate in the primary key, which is a date/time field supporting precision only to the second. Participants systems must manage this Throttling limit because if they submit multiple JSON requests for the same participantId at the same time, AEMO may reject some.
getBid getBids getSubmission getSubmissions	1 request per second per participantId

Chapter 3 Bidding API

This chapter describes the Bidding API and its Endpoints, validation rules, request and response structure, and examples.

submitBids

The submitBids Endpoint allows participants to submit one or more Energy, FCAS, or MNSP Bids.

If you submit a Bid/Offer with multiple Units and multiple Trading Days in a single request, and one has an error, AEMO rejects the whole Submission.

You provide the Bid/Offer details in the energyBids, fcasBids, or mnspBids arrays in the JSON schema. Every Submission must supply the entire Submission level information. For details, see submitBids parameters on page 14.

The JSON Schema supports the provision of multiple Bids/Offers for a specific Trading Day, DUID (Dispatchable Unit or MSNP Interconnector ID), and FCAS Service Type.

submitBids request

Item	Description
Method	POST
URL	<Base URL>/NEMWholesale/bidding/v1/submitBids
Header	Standard request header attributes, include: Authorization: Basic …. Content-Encoding: gzip or deflate Accept-Encoding: gzip or deflate

Request	{
	"submissionTimeStamp": [date/time],
	"referenceId": [string],
	"comments": [string],
	"authorisedBy": [string],
	"energyBids": [array],
	"fcasBids": [array],
	"mnspBids": [array]
	}

submitBids parameters

Name	Required	DataType	Description
submissionTimeStamp	N	date/time	Participant provided timestamp for the Submission. Expected in the format: yyyy-mm-ddThh:MM:ss[+10:00] e.g. 2021-04-23T20:20:39 2021-04-23T20:20:39+10:00
referenceId	N	string (100)	A participant provided reference. Must be unique for each participant
comment	N	string (100)	Allows the participant to provide a comment or description for this Submission
authorisedBy	N	string (20)	Person authorising this Submission. Used for participant's reference but not validated
energyBids	Y	array	Collection of one or more Energy bids. You must provide at least one array of energyBids/fcasBids/mnspBids
fcasBids	Y	array	Collection of one or more FCAS bids. You must provide at least one array of energyBids/fcasBids/mnspBids
mnspBids	Y	array	Collection of one or more MNSP bids. You must provide at least one array of energyBids/fcasBids/mnspBids

submitBids response

Item	Description
Response code	200

Success response	{
	"transactionId": [string],
	"data":{
	"participantId": [string],
	"offerTimeStamp": [date/time],
	"transactionId": [string],
	"referenceId": [string],
	"comments": [string],
	"filename": [string],
	"authorisedBy": [string],
	"status": [string],
	"method": [string],
	},
	    "errors": [],
	    "warnings": []
	}

Success response example	{
	"transactionId" : "10ad7d61-27fb-4446-98fe-f4cd3622c8f4",
	"data":
	{
	"participantId": "XYZ",
	"offerTimeStamp": "2021-07-01T01:03:11+10:00",
	"transactionId": "rrt-124",
	"referenceId": "rrt-123",
	"comments": "My first bid",
	"filename": "AEMO_constructed.API",
	"authorisedBy": "Max",
	"status": "VALID",
	"method": "API",
	},
	    "errors": [],
	    "warnings": []
	}

Item	Description

Invalid response	{
	"transactionId": [string],
	"data":{
	"participantId": [string],
	"offerTimeStamp": [date/time],
	"transactionId": [string],
	"referenceId": [string],
	"comments": [string],
	"filename": [string],
	"authorisedBy": [string],
	"status": [string],
	"method": [string],
	},
	"errors": [
	{
	"code": [string],
	"source": [string]
	"title": [string],
	"detail": [string],
	}
	],
	"warnings": [
	{
	"code": [string],
	"source": [string]
	"title": [string],
	"detail": [string],
	}
	]
	}

<table>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Invalid response example</td><td><pre>
HTTP/1.1 422 Unprocessable Entity
Content-Type: application/json
Content-Length: nnn
Content-Encoding: gzip
{
   "transactionId" : "10ad7d61- 27fb- 4446- 98fe-
f4cd3622c8f4",
 "data":
  {
   "participantId": "PARTID",
   "referenceId": "123",
   "offerTimeStamp": "2021-07-01T01:03:11+10:00",
   "comments": "My first bid",
   "status": "CORRUPT",
   "filename": "ACMECORP_BID_202107019171936410.API",
   "method": "API",
   "authorisedBy": "Max",
  },
 "errors": [
 {
   "code": "NEM-Bidding-Price-ExceedsMPC",
   "title": "Price Exceeds MPC",
   "detail": "Prices must not exceed the market price
cap
   of $15,000",
    "source" : "$.energyBids [? (@.duid = 'UNIT1'
&amp;&amp;
   @.tradingDate= '2021- 04- 21')].energyPeriods
[100].price"
  }
]
</pre></td></tr>
</table>

getBids

The getBids Endpoint returns one or more Energy, FCAS, or MNSP Bid/Offers (not including interval data).

Participants can use the following optional parameters to search for specific submissions (for details, see getBids above):

- fromTradingDate
- toTradingDate
- duid or interconnectorId
- service
- includeSuperseded

getBids request

Item	Description
Method	GET
URL	<Base URL>/NEMWholesale/bidding/v1/getBids
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip or deflate
Request	Not applicable

getBids parameters

For information about how to use these optional parameters, see getBids scenarios on page 22.

Name	Required	Datatype	Description
fromTradingDate	N	date	From Trading Day (inclusive) Default is current Trading Day
toTradingDate	N	date	To Trading Day (inclusive) Default is fromTradingDate + 7 days A seven day range is returned. If you need a greater range, submit this Endpoint multiple times. For the selected date range when there is no explicit Bid/Offer, the last valid Bid/Offer displays
duid	N	string	Comma separated list of Dispatchable Unit IDs (DUID) or MNSP Interconnector IDs
service	N	string	Comma separated list of services: 1. ENERGY 2. MNSP 3. RAISE6SEC 4. RAISE60SEC 5. RAISE5MIN 6. RAISEREG 7. LOWER6SEC 8. LOWER60SEC 9. LOWER5MIN 10. LOWEREG
includeSuperseded	N	boolean	Default = FALSE If set to true includes all versions of bids, otherwise the current effective bids are returned

getBids response

Item	Description
Response code	200
Header	Standard request header attributes, include: Authorization: Basic …. Content-Encoding: gzip or deflate

Success response	{
	"transactionId": [string],
	"data": {
	"bids": [
	 {
	  "referenceId": [string],
	  "transactionId: [string],
	  "offerTimeStamp": [date/time],
	  "tradingDate": [string],
	  "duid": [string],
	  "service": [string],
	  "entryType": [string],
	  "rebidExplanation": [object],
	 },
	…
	]
	},
	"errors": [],
	"warnings": []
	}

Success response example	HTTP/1.1 200 OK Content-Type: application/json Content-Length: nnn Content-Encoding: gzip { "transactionId": "4585bb2e-328a-4726-960f-b6ac15d8dc08", "data": { "bids": [

Item	Description

```
   {
     "referenceId": "#1234",
       "transactionId"  :    "4abab6ec- 6aa7- 4f75- bc3f-
9060d83dda83",
     "duid": "UNIT1",
     "tradingDate": "2021-04-25",
     "offerTimestamp ": "2021-04-24T15:03:16",
     "service": "ENERGY"
     "entryType": "REBID",
     "rebidExplanation": {
     "reason": "Unit trip",
     "eventTime": "13:10:22"
 },
},
{
   "referenceId": "#1235"
     "transactionId"  :    "4abab6ec- 6aa7- 4f75- bc3f-
9060d83dda83",
   "duid": "UNIT2",
   "tradingDate": "2021-04-25",
   "offerTimestamp ": "2021-04-24T15:03:16",
   "service": "ENERGY"
   "offerDateTime": "2021-04-24T15:03:16",
   "entryType": "DAILY",
   "rebidExplanation": {}
}
]
},
“errors”:[],
“warnings”:[]
}
```

getBids scenarios

Parameter used	Outcome	Explanation
None	The current Bid/Offer for each Service Type for each owned duid / InterconnectorId between the current tradingDate and the current tradingDate + 7 days	You can use the return values to find the full bid detail or getSubmission to find the full Submission detail
includeSuperseded	All Bids/Offers for each Service for each Partipant ID owned duid / InterconnectorId submitted for the Trading Dates between the current Trading Date and the current Trading Date + 7 days	Returns all Bid/Offers, not only effective Bids/Offers.

getBid

The getBid Endpoint returns all details for a specific Bid/Offer, including interval, day, and Submission information.

If the parameters match, the specific Bid/Offer details return, else an error displays.

This Enpoint only returns one of the three types of bids - Energy, FCAS, or MNSP.

getBid request

Item	Description
Method	GET
URL	<Base URL>/NEMWholesale/bidding/v1/getBid
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip or deflate Accept-Encoding: gzip or deflate
Request	Use required parameters (see getBid parameters on the next page) For example: **/NEMWholesale/bidding/v1/getBi?DUID=P01S01D1&TradingDate=2020-07-30&Service=ENERGY&OfferTimestamp=2020-10-09 14:36:31**

getBid parameters

Name	Required	Data type	Description	Example
tradingDate	Y	date	The trading day for the Bid/Offer	2021-04-25
duid	Y	string	The Unit or MNSP interconnectorId A successful MNSP getBid response returns Link IDs in the DUID objects, not Interconnector IDs	UNIT1 or T-V-MNSP1
offerTimeStamp	Y	date/time	Bid/Offer date and time	2021-04-24T15:03:16
service	Y	string	Enter only 1 of the following Service Types: 1. ENERGY 2. MNSP 3. RAISE6SEC 4. RAISE60SEC 5. RAISE5MIN 6. RAISEREG 7. LOWER6SEC 8. LOWER60SEC 9. LOWER5MIN 10. LOWERREG	Energy

getBid response

Item	Description
Response code	200
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip

Success response	`{`
	`"transactionId": [string],`
	`"data":`
	`{`
	` "participantId": [string],`
	` "offerTimeStamp": [date/time],`
	` "transactionId": [string],`
	` "referenceId": [string],`
	` "filename": [string],`
	` "status": [string],`
	` "submissionTimeStamp": [string],`
	` "comments": [string],`
	` "authorisedBy": [string],`
	` "method": [string],`
	` "energyBid": [object]`
	` "fcasBid": [object]`
	` "mnspBid": [object]`
	`},`
	` "errors": [],`
	` "warnings": []`
	`}`

Success response example	`Content-Encoding: gzip`
	`{`
	`“transactionId”: “4abab6ec-6aa7-4f75-bc3f-9060d83dda83”,`
	`“data”:`
	` {`

Item	Description

```
  “participantId”: “ACMECORP”,
  “offerTimeStamp”: “2021-04-24T15:03:16”,
  “transactionId”: “4585bb2e-328a-4726-960f-b6ac15d8dc08”,
  “referenceId”: “#12345”,
  “comments”: “Daily offer”,
  “filename”: “ACMECORP_BID_19991211132538651.API”,
  “authorisedBy”: “Max”,
  “Status”: “VALID”,
  “method”: “API”,
  “energyBid”: {
  “entryType”: “DAILY”,
  “duid”: “UNIT1”,
  “tradingDate”: “2021-04-25”,
  "fastStartProfile": {},
  "rebidExplanation": {},
  “prices”:                                                       [-
1000.00
,
0.00
,100.00,200.00,500.00,1000.00,5000.00,10000.00,11000.00,12000.00],
  “energyPeriods”: [
 {
  “periodId”: 1,
  “rampUpRate”: 3,
  “rampDownRate”: 3,
  “pasaAvail”: 100,
  “maxAvail”: 100,
  “bandAvail”: [0,0,0,50,50,0,0,0,0,0]
 },
 {
  “PeriodId”: 2,
  “rampUpRate”: 3,
  “rampDownRate”: 3,
  “pasaAvail”: 100,
  “maxAvail”: 100,
  “bandAvail”: [0,0,0,50,50,0,0,0,0,0]
}
]
}
}
“errors”:[]
“warnings”:[]
}
```

getSubmissions

The getSubmissions Endpoint retrieves a list of accepted submissions where the requesting participantId is the submitting participantId.

You can use the following optional parameters to search for specific submissions (for details, see getSubmissions parameters on the next page):

- fromTradingDate
- toTradingDate
- transactionId
- fromOfferTimeStamp
- toOfferTimeStamp
- referenceId
- comments

getSubmissions request

Item	Description
Method	GET
URL	<Base URL>/NEMWholesale/bidding/v1/getSubmissions
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip or deflate
Request	Not applicable

getSubmissions parameters

For more information on how to use these optional parameters, refer to getSubmissions scenarios on page 31.

Name	Required	Datatype	Description
fromTradingDate	N	date	Date of the Trading Day to query from (inclusive) No default
toTradingDate	N	date	Date of the Trading Day to query to (inclusive) No default
transactionId	N	string	Filter based on a partial match to the transactionId
fromOfferTimeStamp	N	date/time	Date/time of the offerTimeStamp to query from (inclusive) Default is current NEM date-time - 90 days
toOfferTimeStamp	N	date/time	Date/time of the offerTimeStamp to query until (inclusive) Default is current NEM date-time
referenceId	N	string	Filter based on a partial match to referenceId because the query is range bound by the primary key fields, ParticipantId and OfferTimeStamp
comments	N	string	Filter based on a partial match to comments because the query is range bound by the primary key fields, ParticipantId and OfferTimeStamp

getSubmissions response

Item	Description
Response code	200
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip or deflate

Success response	`{`
	`"transactionId": [string],`
	`"data":`
	`{`
	` "submissions": [`
	` {`
	`  "participantId": [string]`
	`  "transactionId": [string],`
	`  "referenceId": [string],`
	`  "offerTimeStamp": [date/time],`
	`  "submissionTimeStamp": [string],`
	`  "comments": [string],`
	`  "status": [string],`
	`  "filename": [string],`
	`  "method": [string],`
	`  "authorisedBy": [string]`
	` },`
	`…`
	`]`
	`},`
	`"errors": [],`
	`"warnings": []`
	`}`

Success response example	`HTTP/1.1 200 OK`
	`Content-Type: application/json`
	`Content-Length: nnn`
	`Content-Encoding: gzip`
	`{`
	`“transactionId”: “76e454bb-ee36-40a5-9b55-709abef50fbe”,`

Item	Description

```
“data”:
{
 “submissions”: [
  {
   “participantId”: “ACMECORP”,
   “offerTimeStamp”: “2021-04-25T12:11:54”,
   “transactionId”:   “ 6 a337759 - 81 dd - 49 c0- ae88 -
3c3046ce653d”,
   “referenceId”: “#1234”,
   “submissionTimeStamp”: “2021-04-25T04:06:46”,
   “comments”: “Test”,
   “filename”: “ACMECORP_BID_20210418121155117.API”,
   “authorisedBy”: “Max”,
   “status”: “VALID”,
   “method”: “API”
 },
 {
   “participantId”: “ACMECORP”,
   “offerTimeStamp”: “2021-04-26T12:11:54”,
   “transactionId”:   “ 6 a337759 - 81 dd - 49 c0- ae88 -
3c3046ce653e”,
   “referenceId”: “#12345”,
   “submissionTimeStamp”: “2021-04-25T04:06:46”,
   “comments”: “Test”,
   “filename”: “ACMECORP_BID_20210426121155117.API”,
   “authorisedBy”: “Max”,
   “status”: “VALID”,
   “method”: “API”
 },
]
},
“errors”:[],
“warnings”:[]
}
```

getSubmissions scenarios

Parameter used	Outcome	Explanation
None	All Submissions made by the requesting participantId from today minus 90 days.	The default toOfferTimeStamp is the current time The default fromOfferTimeStamp is the current day minus 90 days
fromOfferTimeStamp toOfferTimeStamp	All Submissions from the given fromOfferTimeStamp until the given toOfferTimeStamp, made by the requesting participantId	You must provide the fromOfferTimeStamp and the toOfferTimeStamp. The range must not exceed 90 days
transactionId fromOfferTimeStamp toOfferTimeStamp	All Submissions from the given fromOfferTimeStamp until the given toOfferTimeStamp exactly matching the given transactionId, made by the requesting participantId	This example also applies for referenceId, transactionId, and comments because all are partial match variables
fromTradingDate	All submissions from the default fromOfferTimeStamp to the default toOfferTimeStamp with Bids/Offers for Trading Dates equal to or greater than the given fromTradingDate, made by the requesting participantId	This is not a definitive list of all Bids/Offers for the given Trading Date range because there is a restriction with the OfferTimeStamp range

getSubmission

The getSubmission Endpoint retrieves a previous Submission where the requesting participantId is the submitting participantId.

Participants can use the following optional parameters to search for a specific submission (for details, see page 33):

- transactionId
- referenceId

You must provide at least one optional parameter otherwise no results return.

getSubmission request

Item	Description
Method	GET
URL	<Base URL>/NEMWholesale/bidding/v1/getSubmission
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip or deflate Accept-Encoding: gzip or deflate
Request	Use one of these parameters (see getSubmission parameters on the next page) For example, /NEMWholesale/bidding/v1/getSubmission?TransactionId=cf81a11e-cb03-446d-97d9-09fd558349c2

getSubmission parameters

For more information on how to use these optional parameters, refer to getSubmission scenarios on page 35.

Name	Required	Datatype	Description
referenceId	N	string	Filter based on an exact match to referenceId because the query is range bound by the primary key fields, ParticipantId and OfferTimeStamp
transactionId	N	string	Filter based on an exact match to the transactionId because the query is range bound by the primary key fields, ParticipantId and OfferTimeStamp

getSubmission response

Item	Description
Response code	200
Header	Standard request header attributes, include: Authorization: Basic .... Content-Encoding: gzip or deflate

Item	Description

Success response	{
	"transactionId": [string],
	"data":
	{
	"participantId": [String]
	"transactionId": [string],
	"referenceId": [string],
	"offerTimeStamp": [date/time],
	"submissionTimeStamp": [string],
	"comments": [string],
	"status": [string],
	"filename": [string],
	"method": [string],
	"authorisedBy": [string],
	"energyBids": [array],
	"fcasBids": [array],
	"mnspBids": [array]
	},
	“errors”:[],
	“warnings”:[]
	}

Success response example	HTTP/1.1 200 OK Content-Type: application/json Content-Length: nnn Content-Encoding: gzip { “transactionId”: “76e454bb-ee36-40a5-9b55-709abef50fbe”, “data”: { “participantId”: “ACMECORP”, “offerTimeStamp”: “2021-04-25T12:11:54”, “transactionId”: “6a337759-81dd-49c0-ae88-3c3046ce653d”, “referenceId”: “#1234”, “submissionTimeStamp”: “2021-04-25T04:06:46”, “comments”: “Test”, “filename”: “ACMECORP_BID_20210418121155117.API”, “authorisedBy”: “Max”, “status”: “VALID”, “method”: “API”, “energyBids”: [{

Item	Description

```
        “entryType”: “DAILY”,
        “duid”: “UNIT1”,
        “tradingDate”: “2021-04-25”,
        "fastStartProfile": {},
        "rebidExplanation": {},
        “prices”:                                   [                  -
    1000.00
    ,
    0.00
    ,100.00,200.00,500.00,1000.00,5000.00,10000.00,11000.00,12000.00],
      }
    ]
    }
    "errors": [ ],
    "warnings": [ ]
    }
```

getSubmission scenarios

Parameter used	Outcome	Explanation
None	No results	You must provide at least one optional value
referenceId	The Submission matching the given referenceId	If no submissions match the reference ID, then it returns a null result
transactionId	The Submission matching the given transactionId	If no submissions match the transaction ID, then it returns a null result
referenceId transactionId	The Submission that matches the given referenceId and the given transactionId	If the Submission for the given data has no data, then it returns a null result

Needing Help

AEMO's Support Hub

IT assistance is requested through one of the following methods:

- Phone: 1300 AEMO 00 (1300 236 600)

  For non-urgent issues, normal coverage is 8:00 AM to 6:00 PM on weekdays, Australian Eastern Standard Time (AEST).

- The Contact Us form on AEMO's website.

Information to provide

Please provide the following information when requesting assistance from AEMO:

- Your contact details
- Company name
- Company ID
- System or application name
- Environment: production or pre-production
- Problem description
- Screenshots

AEMO recommends participants call AEMO's Support Hub for all urgent issues, whether or not you have logged a call using the contact us form.

For AEMO software-related issues please also provide:

- Participant ID (if Data Interchange (DI) problem)
- Version of software
- Properties or log files
- PDR Monitor support dump and DI instance name (if DI problem)

Rules Terms

Glossary

30-min period
New rules term replacing Trading Interval, where the period remains as 30 minutes

AEMO API Gateway
The gateway on AEMO's side providing participant communication options, accessible over the internet or MarketNet. It uses resources and methods to push messages to Participants' API Gateways .

AEST
Australian Eastern Standard Time

API
Application Programming Interface. A set of clearly defined methods of communication between various software components.

API Portal
Where you can view available APIs, manage your API Keys, and obtain OAS files.

API Protocol
An e-Hub delivery method.

Bids/Offers
Relates to the following Dispatch Bids: 1. Energy (Scheduled Loads) The term Offer relates to the following Dispatch Offers: 1. Energy (Generation Dispatch Offer) 2. Frequency Control Ancillary Service (FCAS) 3. Market Network Service Provider (MNSP - Network Dispatch Offer)

CSR
Certificate Signing Request is a block of encoded text given to a Certificate Authority when applying for an SSL Certificate. It also contains the Public Key to include in the certificate. Usually, a Private Key is created at the same time, making a Key Pair.

csv
Comma-separated values; a file format for exchanging data.

Curl
A command line utility used to interact with REST API endpoints.

Data Model
The definition of the interface to participants of data published by AEMO for gas or electricity. A database conforming to the Data Model can contain a local copy of all current participant-specific data recorded in the main database. The Data Model includes database tables, indexes, and primary keys

DUID
Dispatchable Unit or MSNP Interconnector ID

e-Hub
Consists of the API Portal and the API Gateway for both electricity and gas.

EDM
Electricity Data Model

EMMS
Wholesale Electricity Market Management System; software, hardware, network and related processes to implement the energy market.

Endpoint
Where the API request is sent and where the response comes from.

FCAS
Frequency Control Ancillary Services

Fixed Load
Optional MW, not greater than the max. availability

FTP
File transfer protocol; a standard network protocol used for the transfer of computer files between a client and server on a computer network.

Header Parameters
Parameters included in the request header.

JSON
Java Standard Object Notation. An agreed format for text files and data exchange. This is now used by AEMO to receive Bids and Offers and provide responses

JSON Schema
Defines the structure and content of the bidding details.

Key Pair
SSL uses a technique called public-key cryptography, based on the concept of a Key Pair. The Key Pair consists of encrypted Public and Private Key data. It is only possible to decrypt the Public Key with the corresponding Private Key.

Link ID
Identifies the MNSP interconnector link in AEMO's systems. A property (linkId) in the MNSPBidLink object in the JSON bidding schema

MarketNet
AEMO’s private network available to participants having a participant ID

Markets Portal
Web portal for access to AEMO's wholesale web-based applications.

Method
The allowed operation for a resource, e.g. GET, POST, PUT, DELETE, and so on. These operations determine whether you’re reading information, creating new information, updating existing information, or deleting information.

MNSP
Market Network Service Provider

MSATS
Retail Market Settlement and Transfer Solution

MSATS Web Portal
MSATS web-based interactive interface

MW
Megawatt

NER
National Electricity Rules

OAS
OpenAPI specification

OpenAPI specification document
The file, either in YAML or JSON, describing your REST API. Follows the OpenAPI specification format.

PA
Participant Administrator who manages participant company's user access and security. The initial PA is set up by the AEMO system administrator as part of the registration process.

Parameters
Parameters are options you pass with the endpoint (such as specifying the response format or the amount returned). There are four types of parameters: header parameters, path parameters, query string parameters, and request body parameters. The different types of parameters are often documented in separate groups on the same page. Not all endpoints contain each type of parameter. See Parameters for more details.

Participant API Gateway
The interface implemented by participants where AEMO pushes messages.

Participant File Server
The publishing point from AEMO systems to participant systems. Each participant is allocated an account and access to private and public areas. Participants are responsible for interfacing with the Participant File Server. If uncollected, files are moved to the archive folder after a couple of days. If your Data Interchange environment is configured properly it automatically retrieves the missing files from the archive. Files are kept in the archive for approximately six months. AEMO’s production and pre-production environments are independently operated, so each environment has its own IP address for its Participant File Server. For help, see Connection to AEMO's IT Systems.

Participant ID
Registered participant identifier

Participant User ID
The user ID you used to login to the system.

Participant Users
Set up by the company's Participant Administrator.

Path
Parameters in the path of the endpoint, before the query string (?). Path parameters are usually set off within curly braces.

Payload
The data sent by a POST request. The Payload section sits after the header.

PID
Participant ID

Pre-production
AEMO’s test system available to participants

Private Key
The secret Private Key is a text file used initially to generate a Certificate Signing Request (CSR), and later to secure and verify connections.

Production
AEMO’s live system

Public Key
The Public Key is included as part of your SSL certificate, and works together with your Private Key to make sure your data is encrypt Key (i.e. the certificate) can verify the digital signature is authentic without having to know the secret Private Key.

Query String Parameters
Parameters in the query string of the endpoint, after the ?.

Request
The way information is returned from an API. In a request, the client provides a resource URL with the proper authorization to an API server. The API returns a response with the information requested.

Request Body Parameters
Parameters in the request body. Usually submitted as JSON.

Response
The information returned by an API after a request is made. Responses are usually in JSON or XML format.

Response Example
The response example shows a sample response from the request example; the response schema defines all possible elements in the response. The response example is not comprehensive of all parameter configurations or operations, but it should correspond with the parameters passed in the request example. The response lets developers know if the resource contains the information they want, the format, and how that information is structured and labeled. The description of the response is known as the response schema. The response schema documents the response in a more comprehensive, general way, listing each property that could possibly be returned, what each property contains, the data format of the values, the structure, and other details.

REST
The Representational State Transfer API architecture

Service Type
ENERGY, MNSP, RAISE6SEC, RAISE60SEC, RAISE5MIN, RAISEREG, LOWER6SEC, LOWER60SEC, LOWER5MIN, LOWERREG

SSL
Secure Sockets Layer, cryptographic protocol providing API communication security

Submission
A Bid/Offer submission can have: 1. Multiple Trading Days 2. Multiple DUIDs/Interconnector IDs 3. All Service Types in the same Submission

Swagger file
Refers to the OpenAPI specification

Throttling
AEMO uses API throttling to prevent overwhelming the API Gateway.

TLS
Transport Layer Security, cryptographic protocol providing API communication security

Unit
Generating Unit

URM
User Rights Management; see the Guide to URM on AEMO's website

zip
The file compression format used for exchanging data with AEMO.

References

You can find resources on AEMO's website.

Guide to AEMO's e-Hub APIs 4

Format and Validation for Energy, FCAS, and MNSP Bids and Offers 4

Connecting to AEMO’s Electricity IT Systems 4

Guide to User Rights Management 5

Guide to AEMO’s e-Hub APIs. 6

Guide to AEMO’s e-Hub APIs 7

Postman 8

TLS Certificates 8

Index