

› eex group

EEX Transparency Data  
- API User Guide

01.11.2018  
Leipzig

Ref. 001

› eex

› pxe

› ecc

› cltx

› epexspot

› nodal

› powernext

› nodalclear

› pegas

› gaspoint  
nordic

## Table of Contents

<b>1.</b>	<b>Versions</b>	<b>5</b>
<b>2.</b>	<b>Glossary</b>	<b>6</b>
<b>3.</b>	<b>Introduction</b>	<b>7</b>
3.1	Scope	7
3.2	Overview	7
3.3	Subscription	7
<b>4.</b>	<b>Included Transparency Data</b>	<b>8</b>
4.1	Transparency Data Types	8
4.2	Detailed definition of Line Types	9
<b>5.</b>	<b>Transparency Data API</b>	<b>11</b>
5.1	Request method	11
5.2	EndPoint	12
5.3	Authentication	12
5.4	Definition of API Formats	12
5.5	Definition of API Data Fields	13
5.5.1	Specific Range of Data Fields	18
5.6	Field specification for API Responses	20
5.6.1	Time series Data	20
5.6.1.1	ExAnteInformationAvailableCapacityPower	20
5.6.1.2	ExAnteInformationNonUsableCapacityPower	20
5.6.1.3	ExPostInformationActualPlantGenerationPower	20

---

5.6.1.4	ExPostInformationActualUnitGenerationPower	21
5.6.1.5	ExAntelInformationPlannedGenerationPower	21
5.6.1.6	ExPostInformationGenerationSolarPower	21
5.6.1.7	ExAntelInformationGenerationSolarPower	22
5.6.1.8	ExPostInformationGenerationWindPower	22
5.6.1.9	ExAntelInformationGenerationWindPower	23
5.6.1.10	ExPostInformationGenerationEuroSolarPower	23
5.6.1.11	ExAntelInformationGenerationEuroSolarPower	23
5.6.1.12	ExPostInformationGenerationEuroWindPower	23
5.6.1.13	ExAntelInformationGenerationEuroWindPower	24
5.6.1.14	ExPostInformationActualConsumptionPower	24
5.6.1.15	ExAntelInformationPlannedConsumptionPower	24
5.6.1.16	ExAntelInformationWorkingCapacityStoragePower	25
5.6.1.17	ExPostInformationActualStoragePower	25
5.6.1.18	ExPostInformationActualConsumptionGas	25
5.6.1.19	ExAntelInformationPlannedConsumptionGas	25
5.6.2	Event Data	26
5.6.2.1	NonUsabilityGenerationPower	26
5.6.2.2	NonUsabilityConsumptionPower	26
5.6.2.3	NonUsabilityStoragePower	27
5.6.2.4	NonUsabilityConsumptionGas	27
5.6.2.5	AdhocMessages	28
5.6.3	Master Data	28
5.6.3.1	Company	28
5.6.3.2	ProdCons	29
5.6.3.3	Unit	29
5.6.3.4	Capacity	29
<b>6.</b>	<b>API Methods</b>	<b>31</b>
6.1	getTimeSeries	31
6.1.1	getTimeSeries Parameters	31
6.2	getEvent	32

6.2.1	getEvent Parameters	32
6.3	getMasterDataCompany	33
6.3.1	getMasterDataCompany Parameters	33
6.4	getMasterDataProdCons	34
6.4.1	getMasterDataProdCons Parameters	34
6.5	getMasterDataUnit	34
6.5.1	getMasterDataUnit Parameters	34
6.6	getMasterDataCapacity	35
6.6.1	getMasterDataCapacity Parameters	35
<b>7.</b>	<b>API Calls</b>	<b>36</b>
7.1	API Requests	36
7.1.1	URL Example Request	36
7.1.2	Java Example Request	37
7.1.3	C# Example Request	39
7.2	API Responses	40
7.3	JSON sample return	41
7.4	XML sample return	41
7.5	Examples	42
7.5.1	getTimeSeries	42
7.5.2	getEvent	42
7.5.3	getMasterData	42
<b>8.</b>	<b>Known Issues</b>	<b>43</b>
8.1	General known issues	43
8.2	Symbol specific known issues	43
8.2.1	ExPostInformationActualPlantGenerationPower	43
8.2.2	ExPostInformationActualUnitGenerationPower	43

## 1. Versions

Version	Description	Date	Author
0001	Initial Draft	01.11.2018	EEX, Erik Hanns

## 2. Glossary

Term	Description
<b>A</b>	
<b>API</b>	Application Programming Interface
<b>C</b>	
<b>CSV</b>	Comma-separated values
<b>E</b>	
<b>EEX</b>	European Energy Exchange
<b>Endpoint</b>	A URL address which services the API requests
<b>J</b>	
<b>JSON</b>	JavaScript Object Notation
<b>M</b>	
<b>MDS</b>	Market Data Services
<b>Method</b>	An operation that takes parameters to create a specific request for data
<b>T</b>	
<b>TPE</b>	Transparency Platform of EEX
<b>U</b>	
<b>URL</b>	Uniform Resource Locator
<b>UTC</b>	Universal Time Coordinated
<b>X</b>	
<b>XML</b>	eXtensible Markup Language

Table 1: Glossary of Terms

## 3. Introduction

### 3.1 Scope

The scope of this document is to provide a programmer who is familiar with web service Application programming interface (web API) with the information required to retrieve transparency data via programmatically.

### 3.2 Overview

The API User Guide provides information about how to retrieve transparency data of the EEX Transparency Platform in a programmatic way. This service provides real time updates of the transparency data available from EEX. The web service API returns XML and JSON responses. In order to access the API users will require credentials. These are available from EEX as part of a subscription package.

### 3.3 Subscription

The API solution allows users to receive real-time or delayed market data and transparency data. Get connected to EEX Group DataSource with fast and robust data directly from the source. Please find all information regarding ordering, prices and available modules of the API on our website.

#### How to subscribe:

<https://www.eex.com/en/market-data/market-data-download/subscription/subscription-process>

#### Contact:

Information Services  
T +49 341 2156-288  
[datasource@eex-group.com](mailto:datasource@eex-group.com)

## 4. Included Transparency Data

EEX Group DataSource offers the following APIs for transparency data:

- Transparency Data Power
- Transparency Data Gas
- Transparency Data Ad-hoc Messages

### 4.1 Transparency Data Types

The products available and their symbol are listed below – full details of the returned data for these symbols and the parameters that can be used, can be found in section 6.

#### Power

Time-series data:

- ExAntelInformationAvailableCapacityPower
- ExAntelInformationNonUsableCapacityPower
- ExPostInformationActualPlantGenerationPower
- ExPostInformationActualUnitGenerationPower
- ExAntelInformationPlannedGenerationPower
- ExPostInformationGenerationSolarPower
- ExAntelInformationGenerationSolarPower
- ExPostInformationGenerationWindPower
- ExAntelInformationGenerationWindPower
- ExPostInformationGenerationEuroWindPower
- ExAntelInformationGenerationEuroWindPower
- ExPostInformationGenerationEuroSolarPower
- ExAntelInformationGenerationEuroSolarPower
- ExPostInformationActualConsumptionPower
- ExAntelInformationPlannedConsumptionPower
- ExAntelInformationWorkingCapacityStoragePower
- ExPostInformationActualStoragePower

Event data:

- NonUsabilityGenerationPower
- NonUsabilityConsumptionPower
- NonUsabilityStoragePower

Master data:

- Company
- ProdCons
- Unit
- Capacity



## Gas

Time-series data:

- ExPostInformationActualConsumptionGas
- ExAnteInformationPlannedConsumptionGas

Event data:

- NonUsabilityConsumptionGas

Master data:

- Company
- ProdCons
- Unit
- Capacity

## Adhoc Messages

Event data:

- AdhocMessages

## 4.2 Detailed definition of Line Types

In order to give users the possibility to establish a relation between API responses and the provided transparency files on sFTP, below is an overview of the Line Types and the respective API symbols.

Line Type	Definition	Symbol for API
ACIL	Available Capacity Information Line	ExAnteInformationAvailableCapacityPower
NUIL	Non-Usable Capacity Information Line	ExAnteInformationNonUsableCapacityPower
APGL	Actual Plant Generation Line	ExPostInformationActualPlantGenerationPower
AUGL	Actual Unit Generation Line	ExPostInformationActualUnitGenerationPower
CPGL	Country Planned Generation Line	ExAnteInformationPlannedGenerationPower
ASPL	Actual Solar Power Generation Line	ExPostInformationGenerationSolarPower
ESPL	Expected Solar Power Generation Line	ExAnteInformationGenerationSolarPower
AWPL	Actual Wind Power Generation Line	ExPostInformationGenerationWindPower
EWPL	Expected Wind Power Generation Line	ExAnteInformationGenerationWindPower
ASPL	Actual Solar Power Generation Line	ExPostInformationGenerationEuroSolarPower
ESPL	Expected Solar Power Generation Line	ExAnteInformationGenerationEuroSolarPower
AWPL	Actual Wind Power Generation Line	ExPostInformationGenerationEuroWindPower

Line Type	Definition	Symbol for API
EWPL	Expected Wind Power Generation Line	ExAnteInformationGenerationEuroWindPower
ACPL	Actual Consumption Power Line	ExPostInformationActualConsumptionPower
PCPL	Planned Consumption Power Line	ExAnteInformationPlannedConsumptionPower
WCPL	Working Capacity Power Line	ExAnteInformationWorkingCapacityStoragePower
APSL	Actual Power Storage Line	ExPostInformationActualStoragePower
ACGL	Actual Consumption Gas Line	ExPostInformationActualConsumptionGas
APGL	Planned Consumption Gas Line	ExAnteInformationPlannedConsumptionGas
NUGL	Non-Usability Generation Line	NonUsabilityGenerationPower
NUCL	Non-Usability Consumption Line	NonUsabilityConsumptionPower
NUSL	Non-Usability Storage Line	NonUsabilityStoragePower
NUCL	Non-Usability Consumption Line	NonUsabilityConsumptionGas
AHML	Ad hoc Message Line	AdhocMessages
COIL	Company Information Line	MasterData-Power/Gas
PCIL	Producer Consumer Information Line	MasterData-Power/Gas
GUIL	Generation Unit Information Line	MasterData-Power
CUIL	Consumption Unit Information Line	MasterData-Power/Gas
SUIL	Storage Unit Information Line	MasterData-Power
GCIL	Generation Capacity Information Line	MasterData-Power
CCIL	Consumption Capacity Information Line	MasterData-Power/Gas
SCIL	Storage Capacity Information Line	MasterData-Power

Table 2: Definition of Line Types

The line type is a returned field, however as the line type is defined and linked to the symbol used when requesting data (for example ExAnteInformationAvailableCapacityPower will return line type ACIL) it is not a parameter that can be filtered on.

## 5. Transparency Data API

Criterion	Description
URL	All API calls will be made to: <ul style="list-style-type: none"> <li>• <a href="https://api1.datasources.eex-group.com">https://api1.datasources.eex-group.com</a></li> </ul>
Data Domain	The API provides access to the Transparency data (Time Series, Event and Master Data) stored in the EEX Transparency Platform Database. This covers the following data areas: <ul style="list-style-type: none"> <li>• Power;</li> <li>• Gas;</li> <li>• Ad-hoc Messages.</li> </ul>
Access	Access to the API is only possible if a client has a subscription package purchased from EEX. Furthermore, API customers will be permissioned for specific subscribed Products/Modules.
Availability	Transparency Data and the API are available for customers 24/7. The return formats provided are XML and JSON.
Source of Data	<a href="#">EEX Transparency Platform</a>
Data Depth	Depending on the request parameters, the API will return real time data updates and historical data.

Table 3: API Basics

### 5.1 Request method

The API is a synchronous interface. It supports one method of requesting data:

- **Https Get**

Any date or filter parameters are a part of the URL query string that is submitted (section 6)

In general, the request string is built as follows:

**<https://datasources.eex-group.com/<method>>**? followed by “**Parameter Name**=”**Parameter Value**”, where each pair of parameter name and value are separated by an ampersand (“&”)

Three methods are supported:

- GetTimeSeries
- GetEvent
- GetMasterData

As the Get request is a URL string, special characters will need to be escaped. Escaping means to replace the special character with the escape symbol. For instance to replace the space in 'a link', you have to use this 'a%20link' in the URL. The most common of these special characters including their respective translation can be found in the table below.

Character	Escape Symbol	Character	Escape Symbol
Space	%20	,	%2C
!	%21	/	%2F
#	%23	:	%3A
\$	%24	;	%3B
&	%26	=	%3D
'	%27	?	%3F
(	%28	@	%40
)	%29	[	%5B
*	%2A	]	%5D
+	%2B		

Table 4: URL Escape Characters

This is not a comprehensive list. There are many resources on the internet that provide lookups for translation of URLs or specific special characters.

## 5.2 EndPoint

The endpoint for the service is:

- <https://api1.datasource.eex-group.com>

Please note, the endpoint supports html only. Furthermore, only https (secure http) address is supported.

## 5.3 Authentication

Authentication uses Basic access authentication. This is a user and password pair.

The user and password will be provided by EEX once a subscription is in place.

## 5.4 Definition of API Formats

The following formats will be used for the data fields of the API.

Term	Description	Example
<dateTime>	Time format – information of a point in time. Please note that all points in time are in UTC.	2017-01-17T01:00:00Z
<string>	Alphanumeric string – used for text information.	Nuclear
<decimal>	The decimal data type is used to specify a numeric value including the zero.	100.9
<boolean>	The boolean data type is used to specify a true or false value.	true
<integer>	The integer data type is used to specify a numeric value without a fractional component.	99

Table 5: URL Escape Characters

## 5.5 Definition of API Data Fields

This is a full list of the data fields and their types for reference. The specific fields returned in a message are defined in section 5.5.1.

Fieldname	Format	Description	example
ActualConsumption	<decimal>	This field contains the actual consumption of consumer. The unit is MW.	80,4
ActualGeneration	<decimal>	This field contains the actual production of generation units. The unit is MW.	82465,4
ActualSolarEnergy	<decimal>	This field contains the actual solar power generation. The unit is MW.	135,5
ActualWindEnergy	<decimal>	This field contains the actual wind power generation. The unit is MW.	18254,6
AvailableCapacity	<decimal>	This field contains the amount of the available capacity (daily average value). The unit is MW.	366,7

Fieldname	Format	Description	example
ControlArea	<string>	This field contains the name of the control area the generation or consumer unit is connected to.	Amprion
Commercialisation	<boolean>	Commercialization of a generation unit.	True
Commodity	<string>	Commodity of the affected Unit.	Power
Country	<string>	Code of the country; Please refer to ISO 3166-1.	DE
CompanyID	<string>	The unique identifier of a company.	POWERHSLTD01
CompanyName	<string>	The name of a company.	Powerhouse Generation Ltd.
CreationTimeStamp	<dateTime>	The timestamp of the file creation.	2009-11-18T18:00:00Z
EndDate	<dateTime>	The end date the data of the generation or consumer unit are delivered.	2011-01-01T00:00:00Z
EventID	<string>	Unique ID for the respective Non-Usability. The EventID is always 29 characters long and is built as follows: <ul style="list-style-type: none"> <li>• Filling (000000000000)</li> <li>• EventID (20180913)</li> <li>• Unique (#2435)</li> <li>• Version number (_013)</li> </ul>	00000000000022246422#2415_001
ExpectedSolarEnergy	<decimal>	This field contains the forecast of the expected generation from solar energy. The unit is MW.	127,5
ExpectedWindEnergy	<decimal>	This field contains the forecast of the expected	2376,5

Fieldname	Format	Description	example
		generation from wind energy. The unit is MW.	
Facility	<string>	Identifier for the value chain of a Unit.	Producer
InstalledCapacity	<decimal>	For generation units the field contains the amount of the installed net bottleneck output. For consumption units the maximum consumption capacity is filled in. The unit is MW.	366,7
LineNumbers	<integer>	Information about the number of lines of the file.	32
MarketArea	<string>	This field contains the name of market area the generation or consumer unit is connected to.	CEGH
MarketParticipantCode	<string>	This field contains the Market Participant Code (ACER Code) of the reporting company.	A0001523E.DE
Message	<string>	Reports additional information.	Revision finished
ModificationTimeStamp	<dateTime>	The timestamp of the modification of this information by the dispatcher.	2009-11-15T11:43:00Z
NonUsableCapacity	<decimal>	This field contains the amount of the non-usable capacity (daily average value). The unit is MW.	366,7
NonavailabilityReason	<string>	Reason of the Non-Availability (i.e. Maintenance, Outage)	Maintenance
NUMCapacity	<decimal>	This field contains the amount of the non-	1265,2

Fieldname	Format	Description	example
		usability of a generation unit. The unit is MW.	
NUMEndDate	<dateTime>	The expected end date of a non-usability.	2011-11-20T14:15:00Z
NUMStartDate	<dateTime>	The start date of a non-usability.	2009-11-19T23:15:12Z
PlannedConsumption	<decimal>	This field contains the forecast of the planned consumption for the next day. The unit is MW.	85405,6
PlannedGeneration	<decimal>	This field contains the forecast of the planned generation for the next day. The unit is MW.	85405,6
PublicationTimeStamp	<dateTime>	The timestamp of the publication of this information on the website.	2009-11-16T00:00:00Z
ProdConsID	<string>	The unique identifier of a generation, consumption or storage plant.	E000001
ProdConsName	<string>	The name of a plant, consumer or storage.	Warp Generator
Quantity	<decimal>	This field contains the actual filling level of power storages. The unit is MWh.	454164,2
RealConsumption	<decimal>	Real consumption of a unit. The unit is MW.	854,6
Remarks	<string>	Any other information that facilitates the full understanding of the potential impact of the event.	Revision finished
ReportingAvailableCapacity	<string>	This field contains information if the production company	True



Fieldname	Format	Description	example
		reports available capacity for all production units.	
Source	<string>	Name of the source.	Biomass
StartDate	<dateTime>	The start date the data of the generation or consumer unit are delivered.	2011-01-01T00:00:00Z
Status	<integer>	The status of a non-usability source.	Active
TimeStamp	<dateTime>	The timestamp of the referring data.	2009-12-31T23:00:00Z
Type	<string>	This field indicates if a generation or consumption unit already was disturbed when the notification was sent.	Planned
UnitID	<string>	The unique identifier of a generation, consumption or storage unit.	E000001-001
UnitName	<string>	The name of a generation, consumption or storage unit.	Core 1
WorkingCapacity	<decimal>	Working capacity of a power storage unit. The unit is MW.	854,6
WGS84Latitude	<decimal>	Latitude gives the location of a place on Earth north or south of the equator.	51.3378
WGS84Longitude	<decimal>	Longitude is the geographic coordinate most commonly used in cartography and global navigation for east-west measurement.	12.3790

Table 6: Definition of API Data Fields

### 5.5.1 Specific Range of Data Fields

Some fields have a fixed set of values that are valid. These are listed below, along with the values that will be returned or can be used for querying.

Fieldname	Value	Translation
Commercialisation	false	Predominantly not for free marketing
	true	Predominantly for free marketing
ControlArea	APG	APG [10YAT-APG-----L]
	Elia	Elia [10YBE-----2]
	CEPS	CEPS [10YCZ-CEPS-----N]
	SwissGrid	Swissgrid [10YCH-SWISSGRIDZ]
	TransnetBW	TransnetBW (formerly EnBW TNG) [10YDE-ENBW-----N]
	TenneT (DE)	TenneT (DE) (formerly Transpower, E.ON) [10YDE-EON-----1]
	Amprion	Amprion (formerly RWE) [10YDE-RWENET---]
	50Hertz	50 Hertz (formerly Vattenfall) [10YDE-VE-----2]
	RTE	RTE [10YFR-RTE-----C]
	National Grid	National Grid [10YGB-----A]
	MAVIR	MAVIR [10YHU-MAVIR----U]
	Terna	Terna [10YIT-GRTN----B]
TenneT (NL)	TenneT (NL) [10YNL-----L]	
MarketArea	Gaspool	GASPOOL [37Y701133MH0000P]
	NCG	NetConnect Germany (NCG) [21Y-ERTV-----8]
	CEGH	MG-OST-AT - Market Area East AT (CEGH) [21Y00000000025G]
	PSV IT	Virtual Trading Point [21Y---A001A010-A]
	VOB	VOB-CZ (formerly RWE Transgas Net) [21Y---A001A001-B]
ReportingAvailableCapacity	false	The company delivers no information on a voluntary commitment.
	true	The company delivers information on a voluntary commitment.
Source	Biomass	Biomass
	Fossil Hard coal	Hard Coal
	Fossil Coal-derived gas	Coal-derived Gas

Fieldname	Value	Translation
	Waste	Waste
	Fossil Gas	Gas
	Geothermal	Geothermal
	Fossil Brown coal/Lignite	Lignite
	Marine	Marine
	Fossil Oil	Oil
	Fossil Oil shale	Oil shale
	Other	Other
	Other renewable	Other Renewable
	Fossil Peat	Peat
	Hydro Pumped Storage	Pumped Storage
	Hydro Run-of-river and poundage	Run-of-River
	Hydro Water Reservoir	Water Reservoir
	Solar	Solar
	Nuclear	Nuclear
	Wind Offshore	Wind (Offshore)
	Wind Onshore	Wind (Onshore)
Status	Active	The non-usability message is active (not cancelled).
	Inactive	The non-usability message is not active (cancelled).
Type	Planned	If the disturbance takes place after the notification was sent.
	Unplanned	If the disturbance has already begun before the notification was sent.
Reason	External factors	Official instructions, environmental legislation requirements, work stoppages or the like lead to a restriction in output or to the standstill of the production, consumption or storage facility.
	Maintenance	Regular planned outage for the renewal, maintenance and review of components.

Fieldname	Value	Translation
	Other	There is a reason which is not covered by the reasons below.
	Outage	There is a technical malfunction on at least one or several components of the production, consumption or storage unit.

Table 7: Specific Range of Data Fields

## 5.6 Field specification for API Responses

### 5.6.1 Time series Data

The following chapter will list all available data types and the fields that the API can return. The heading for each section is the SYMBOL as passed into the API.

#### 5.6.1.1 ExAnteInformationAvailableCapacityPower

The Available Capacity Information contains information on the available capacity of generation units. The Available Capacity Information will return the following data types:

- Symbol
- LineType
- Country
- Source
- TimeStamp
- AvailableCapacity
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.2 ExAnteInformationNonUsableCapacityPower

The Non-Usable Capacity contains information on the non-usable capacity of generation units. The Non-Usable Capacity Information will return the following data types:

- Symbol
- LineType
- Country
- Source
- TimeStamp
- NonUsableCapacity
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.3 ExPostInformationActualPlantGenerationPower

The Actual Generation contains information on the production of all power generation units per source in the respective country. The Actual Generation information will return the following data types:

- Symbol
- LineType
- Country
- Source
- TimeStamp
- ActualGeneration
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.4 **ExPostInformationActualUnitGenerationPower**

The Actual Unit Generation contains information on the actual production of generation units. The Actual Unit Generation will return the following data types:

- Symbol
- LineType
- Country
- UnitID
- TimeStamp
- ActualGeneration
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.5 **ExAnteInformationPlannedGenerationPower**

The Country Planned Generation Line containing information of the planned generation in the respective country by source. The Country Planned Generation will return the following data types:

- Symbol
- LineType
- Country
- Source
- TimeStamp
- PlannedGeneration
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.6 **ExPostInformationGenerationSolarPower**

The Actual Solar Power Generation contains publications of the actual generation from solar energy per control area. The Actual Solar Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ActualSolarEnergy
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.1.7 **ExAnteInformationGenerationSolarPower**

The Expected Solar Power Generation Line is used for publication of the expected generation from solar energy per control area. The Expected Solar Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ExpectedSolarEnergy
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.1.8 **ExPostInformationGenerationWindPower**

The Actual Wind Power Generation contains publications of the actual generation from wind energy per control area. The Actual wind Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ActualWindEnergy
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.9 **ExAnteInformationGenerationWindPower**

The Expected Wind Power Generation Line is used for publication of the expected generation from wind energy per control area. The Expected wind Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ExpectedWindEnergy
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.10 **ExPostInformationGenerationEuroSolarPower**

The Actual Solar Power Generation contains publications of the actual generation from solar energy per control area provided by EuroWind. The Actual Solar Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ActualSolarEnergy
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.11 **ExAnteInformationGenerationEuroSolarPower**

The Expected Solar Power Generation Line is used for publication of the expected generation from solar energy per control area provided by EuroWind. The Expected Solar Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ExpectedSolarEnergy
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.12 **ExPostInformationGenerationEuroWindPower**

The Actual Wind Power Generation contains publications of the actual generation from wind energy per control area provided by EuroWind. The Actual wind Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ActualWindEnergy
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.13 **ExAntelInformationGenerationEuroWindPower**

The Expected Wind Power Generation Line is used for publication of the expected generation from wind energy per control area provided by EuroWind. The Expected wind Power Generation will return the following data types:

- Symbol
- LineType
- ControlArea
- Source
- TimeStamp
- ExpectedWindEnergy
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.14 **ExPostInformationActualConsumptionPower**

The Actual Consumption Power contains information on the actual consumption of power units in the respective country. The Actual Consumption Power will return the following data types:

- Symbol
- LineType
- Country
- TimeStamp
- ActualConsumption
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.15 **ExAntelInformationPlannedConsumptionPower**

The Planned Consumption Power Line containing information of the planned consumption in the respective country. The Planned Consumption Power will return the following data types:

- Symbol
- LineType
- Country
- TimeStamp
- PlannedConsumption
- PublicationTimeStamp
- ModificationTimeStamp



#### 5.6.1.16 **ExAnteInformationWorkingCapacityStoragePower**

The Working Capacity Power contains information on the working capacity of power units. The Working Capacity Power will return the following data types:

- Symbol
- LineType
- UnitID
- TimeStamp
- WorkingCapacity
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.17 **ExPostInformationActualStoragePower**

The Actual Power Storage contains information on the actual storage of power units in the respective country. The Actual Power Storage will return the following data types:

- Symbol
- LineType
- Country
- TimeStamp
- Quantity
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.18 **ExPostInformationActualConsumptionGas**

The Actual Consumption Gas contains information on the actual consumption of gas units. The Actual Consumption Gas will return the following data types:

- Symbol
- LineType
- MarketArea
- TimeStamp
- ActualConsumption
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.1.19 **ExAnteInformationPlannedConsumptionGas**

The Planned Consumption Gas Line containing information of the planned consumption in the respective market area. The Planned Consumption Gas will return the following data types:

- Symbol
- LineType
- MarketArea
- TimeStamp
- ActualConsumption
- PublicationTimeStamp
- ModificationTimeStamp

## 5.6.2 Event Data

### 5.6.2.1 NonUsabilityGenerationPower

The Non-Usability Generation Line is used for publication of Non-Usability Messages of power generation units. The Non-Usability Generation will return the following data types:

- Symbol
- LineType
- Country
- CompanyID
- ProdConsID
- UnitID
- Commodity
- Facility
- ControlArea
- Source
- Type
- EventID
- NUMStartDate
- NUMEndDate
- NUMCapacity
- NonavailabilityReason
- Remarks
- TimeStamp
- Status
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.2.2 NonUsabilityConsumptionPower

The Non-Usability Consumption Line is used for publication of Non-Usability Messages of power consumption units. The Non-Usability Consumption will return the following data types:

- Symbol
- LineType
- Country
- CompanyID
- ProdConsID
- UnitID
- Commodity
- Facility
- ControlArea
- Type
- EventID
- NUMStartDate
- NUMEndDate
- NUMCapacity
- NonavailabilityReason

- Remarks
- TimeStamp
- Status
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.2.3 **NonUsabilityStoragePower**

The Non-Usability Storage Line is used for publication of Non-Usability Messages of power storage units. The Non-Usability Generation will return the following data types:

- Symbol
- LineType
- Country
- CompanyID
- ProdConsID
- UnitID
- Commodity
- Facility
- ControlArea
- Type
- EventID
- NUMStartDate
- NUMEndDate
- NUMCapacity
- NonavailabilityReason
- Remarks
- TimeStamp
- Status
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.2.4 **NonUsabilityConsumptionGas**

The Non-Usability Consumption Line is used for publication of Non-Usability Messages of gas consumption units. The Non-Usability Consumption will return the following data types:

- Symbol
- LineType
- Country
- CompanyID
- ProdConsID
- UnitID
- Commodity
- Facility
- MarketArea
- Type
- EventID
- NUMStartDate
- NUMEndDate

- NUMCapacity
- NonavailabilityReason
- Remarks
- TimeStamp
- Status
- PublicationTimeStamp
- ModificationTimeStamp

#### 5.6.2.5 **AdhocMessages**

The Ad hoc Message contains information on the ad hoc messages. The Ad hoc Message will return the following data types:

- Symbol
- LineType
- Country
- CompanyID
- CompanyName
- EventID
- Status
- AdhocType
- StartDate
- EndDate
- Message
- TimeStamp
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.3 Master Data

#### 5.6.3.1 **Company**

The Company Information Line is used for publication of company information. The Company Information will return the following data types:

- Symbol
- LineType
- CompanyID
- CompanyName
- MarketParticipantCode
- ReportingAvailableCapacity
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.3.2 **ProdCons**

In the Producer and Consumer Information Line information to producers and consumers can be found. The Producer and Consumer Information will return the following data types:

- Symbol
- LineType
- CompanyID
- ProdConsID
- ProdConsName
- Commodity
- Facility
- WGS84Latitude
- WGS84Longitude
- Country
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.3.3 **Unit**

The Consumption Unit Information Line is used for publication of information regarding to consumption units. The Consumption Unit Information will return the following data types:

- Symbol
- LineType
- CompanyID
- ProdConsID
- UnitID
- UnitName
- Commodity
- Facility
- ControlArea (Power) ; MarketArea (Gas)
- Source
- Commercialisation
- StartDate
- EndDate
- PublicationTimeStamp
- ModificationTimeStamp

### 5.6.3.4 **Capacity**

The Generation Capacity Information Line is used for publication of information referring to the installed capacity of generation (installed net bottleneck output) units. The Generation Capacity Information will return the following data types:

- Symbol
- LineType
- UnitID
- TimeStamp
- Capacity

- PublicationTimeStamp
- ModificationTimeStamp

## 6. API Methods

### 6.1 getTimeSeries

The method returns time series data from the EEX Transparency Platform database:

#### 6.1.1 getTimeSeries Parameters

Name	Optional	Description
Symbol	No	Identifier for time series data package.
Start	No	A date in the format, depending on the time resolution of the time series requested (e.g. 'YYYY-MM-DD hh:mm'). Is equal to the first output TimeStamp.
End	No	A date in the format, depending on the time resolution of the time series requested (e.g. 'YYYY-MM-DD hh:mm'). Is equal to the last output TimeStamp.
DaysBack	No	Synonymous with the START/ END pair, with the proviso that END is the current date. So 3, would count 3 days back from today.
Country	Yes	Country which data is requested for (e.g. DE, AT, CH, ...).
ControlArea	Yes	Control Area which data is requested for (e.g. Amprion, APG, ...).
MarketArea	Yes	Market Area which data is requested for (e.g. CEGH, VOB, ...).
Source	Yes	Fuel type which data is requested for (e.g. Biomass, Fossil Gas, ...).
UnitID	Yes	Unit which data is requested for (e.g. E110235-001, S110328-001, ...).
TimeStamp	Yes	Is related to quantity value. Number of time stamps depends on request.
Quantity	Yes	Value for a specific time stamp in the specific unit (e.g. MW, MWh, ...).
PublicationTimeStamp	Yes	Date and time of the publication into database of the respective value.
ModificationTimeStamp	Yes	Date and time of the last update of the respective value.

Table 8: getTimeSeries Parameters

## 6.2 getEvent

The method returns event data from the EEX Transparency Platform database:

### 6.2.1 getEvent Parameters

Name	Optional	Description
Symbol	No	Identifier for event data package.
Event_Start	No	A date in the format, 'YYYY-MM-DD hh:mm'. Is equal to the NUMStartDate.
Event_End	No	A date in the format, 'YYYY-MM-DD hh:mm'. Is equal to the NUMEndDate.
Update_Start	No	Expressed as date in the format, 'YYYY-MM-DD hh:mm'. Requests events with ModificationTimeStamp $\geq$ 'YYYY-MM-DD hh:mm'. If specified, Update_End needs be specified as well.
Update_End	No	Expressed as date in the format, "YYYY-MM-DD hh:mm". Requests events with ModificationTimeStamp $\leq$ 'YYYY-MM-DD hh:mm'. If specified, Update_Start needs be specified as well.
Event_Back	No	Expressed as an integer. Returns the most recent events based on the ModificationTimeStamp for the requested parameters.
Update_Back	No	Expressed as an integer. Synonymous with the Update_Start/ Update_End pair, with the proviso that Update_End is the current date.
Country	Yes	Country which data is requested for (e.g. DE, AT, CH, ...).
CompanyID	Yes	Affected company
ProdConsID	Yes	Affected facility
UnitID	Yes	Affected unit
Commodity	Yes	Commodity of the affected Unit (Power, Gas).
Facility	Yes	Identifier if the affected Unit is Producer, Storage or Consumer.
ControlArea	Yes	Power: Control Area of the affected unit (e.g. Amprion, APG, ...).
MarketArea	Yes	Gas: Market Area of the affected unit (e.g. CEGH, VOB, ...).
Source	Yes	Fuel type of the affected unit (e.g. Biomass, Fossil Gas, ...).
Type	Yes	Type of Non-Usability (Planned or Unplanned).
EventID	Yes	Unique identifier of the specific event.



Name	Optional	Description
NUMStartDate	Yes	Begin of the event (same as Event_Start).
NUMEndDate	Yes	Stop of the event (same as Event_End).
NUMCapacity	Yes	Value for non-usable capacity in MW.
NonavailabilityReason	Yes	Reason of the event (e.g. Outage, Maintenance, ...).
TimeStamp	Yes	Date and time of the initial message.
Status	Yes	Status of the event (Active or Inactive).
PublicationTimeStamp	Yes	Date and time of publication into database.
ModificationTimeStamp	Yes	Date and time of the last modification of the respective event.

Table 9: getEvent Parameters

## 6.3 getMasterDataCompany

The method returns master data from the EEX Transparency Platform database:

### 6.3.1 getMasterDataCompany Parameters

Name	Optional	Description
Symbol	No	Identifier for Company data package
CompanyID	Yes	ID of the company.
CompanyName	Yes	Name of the company.
MarketParticipantCode	Yes	Market Participant Code (ACER Code) of the reporting company.
ReportingAvailableCapacity	Yes	Flag for the reporting of the available capacity (true or false).
PublicationTimeStamp	Yes	Date and time of publication into database.
ModificationTimeStamp	Yes	Date and time of the last modification of the entries.

Table 10: getMasterDataCompany Parameters

## 6.4 getMasterDataProdCons

The method returns master data from the EEX Transparency Platform database:

### 6.4.1 getMasterDataProdCons Parameters

Name	Optional	Description
Symbol	No	Identifier for ProdCons data package.
CompanyID	Yes	ID of the company.
ProdConsID	Yes	ID of the facility.
ProdConsName	Yes	Name of the facility.
Commodity	Yes	Commodity of the affected unit (Power, Gas).
Facility	Yes	Identifier if the affected unit is Producer, Storage or Consumer.
WGS84Latitude	Yes	location of a place on earth north or south of the equator
WGS84Longitude	Yes	geographic coordinate navigation for east-west
Country	Yes	Country of the facility.
PublicationTimeStamp	Yes	Date and time of publication into database.
ModificationTimeStamp	Yes	Date and time of the last modification of the entries.

Table 11: getMasterDataProdCons Parameters

## 6.5 getMasterDataUnit

The method returns master data from the EEX Transparency Platform database:

### 6.5.1 getMasterDataUnit Parameters

Name	Optional	Description
Symbol	No	Identifier for unit data package.
Commodity	No	Valid values are Gas or Power.
CompanyID	Yes	ID of the company.
ProdConsID	Yes	ID of the facility.
UnitID	Yes	ID of the unit.
UnitName	Yes	Name of the unit.

Name	Optional	Description
Commodity	Yes	Commodity of the affected unit (Power, Gas).
Facility	Yes	Identifier if the affected unit is Producer, Storage or Consumer.
ControlArea	Yes	Control Area of the affected unit (e.g. Amprion, APG, ...).
MarketArea	Yes	Market Area of the affected unit (e.g. CEGH, VOB, ...).
Source	Yes	Fuel type which data is requested for (e.g. Biomass, Fossil Gas, ...).
Commercialisation	Yes	Is the output free for marketing (true or false).
StartDate	Yes	Begin of reporting.
EndDate	Yes	End of reporting.
PublicationTimeStamp	Yes	Date and time of publication into database.
ModificationTimeStamp	Yes	Date and time of the last modification of the entries.

Table 12: getMasterDataUnit Parameters

The compulsory parameter Commodity will determine whether ControlArea or MarketArea is returned in the response. If the Commodity=Power, then ControlArea will be returned. If the Commodity=Gas, then the MarketArea will be returned. If the request specifies Power but tries to filter using the MarketArea parameter, no results will be returned. Similarly, if the request specifies Gas but tries to filter using the ControlArea parameter, no results will be returned.

## 6.6 getMasterDataCapacity

The method returns master data from the EEX Transparency Platform database:

### 6.6.1 getMasterDataCapacity Parameters

Name	Optional	Description
Symbol	No	Identifier for time capacity data package.
UnitID	Yes	ID of the unit.
TimeStamp	Yes	Describes the date from which this value is valid.
Capacity	Yes	Value for the installed Capacity (MW).
PublicationTimeStamp	Yes	Date and time of publication into database.
ModificationTimeStamp	Yes	Date and time of the last modification of the entries.

Table 13: getMasterDataCapacity Parameters

## 7. API Calls

Below you will find examples for accessing the API.

### 7.1 API Requests

As described in section 5.1, subscribed customers are able to request the web service API. In the following the major request methods as well as example responses are described.

#### 7.1.1 URL Example Request

The simplest example is using an https request and should illustrate the structure of the request and the expected form of response.

The requests below query the results for the Power Consumption Non-Usabilities for the company ID ENBWTRADING0 with Update time between 2018-01-01 11:20 and 2018-08-11 11:15 UTC.

#### JSON Request (additional /JSON qualifier in request):

- `https://api1.datasource.eex-group.com/getEvent/json?Symbol=NonUsabilityConsumptionPower&Update_Start=2018-01-01%2011:20&Update_End=2018-08-11%2015:00&CompanyID=ENBWTRADING0`

#### XML Request (no qualifier):

- `https://api1.datasource.eex-group.com/getEvent/?Symbol=NonUsabilityConsumptionPower&Update_Start=2018-01-01%2011:20&Update_End=2018-08-11%2011:15&CompanyID=ENBWTRADING0`

The data format required to be passed into the call is UTC standard date format: YYYY-MM-DD hh:mm. In the response the times returned are also in UTC, in standard format.

The API call must always include one of these start/end parameters as follows:

- Update\_Start/End pair
- Event\_Start/End pair
- Update\_Back
- Event\_Back

Additionally, the following conditions need to be considered:

- If an Update\_Start parameter is provided, an Update\_End parameter must be provided (and vice-versa)

- If an Event\_Start parameter is provided, an Event\_End parameter must be provided (and vice versa)
- An Update\_Back parameter is a numeric value representing number of days back
- An Event\_Back parameter is a number value representing the number of events back (for the events method) to retrieve data for

date parameters take precedence over the Update\_Back and Event\_Back parameters. If dates are provided, the relevant “Back” parameter is ignored. If dates are not provided, the “Back” parameter(s) calculate days back using 00:00:00 UTC time as a start time and 23:59:59 as an end time.

Please note, if all parameters are left blank, an error is returned noting all of the parameters cannot be left blank.

Additional filter parameters can be included. The example above uses the CompanyID to filter the results to return only those from ENBWTRADING0.

The range parameters should be the first parameters in the API call, after that the order of parameters is not important.

A full list of parameters for each call is detailed in the section 6 (API Methods).

### 7.1.2 Java Example Request

Below is the same example request written using Java.

```
package com.company.ps.test;

import java.io.DataOutputStream;
import java.io.IOException;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.Scanner;

import sun.misc.BASE64Encoder;

public class EEXRequest {

    // Main method
    public static void main(String[] args) {

        EEXRequest client = new EEXRequest();
        if (args.length >= 3) {
            try {
                client.execute(args[0], args[1], args[2]);
            } catch (Exception e) {
```

```

        e.printStackTrace();
    }
} else {
    System.out.println("Usage : EEXRequest <URL> <user>
<password>");
    // Example https://api1.datasources.eex-
group.com/getEvent/csv?Symbol=NonUsabilityConsumptionPower&Update_Start=2018-01-
01T11:20Z&Update_End=2018-08-11T11:15Z user1 password123
}
}

public void execute(String serverURL, String user, String password) throws
MalformedURLException, IOException {

    // If you need to go through proxy by setting system properties

    // Proxy server
//
//      System.getProperties().setProperty("proxySet", "true");
//
//      System.getProperties().setProperty("proxyHost", proxy);
//
//      System.getProperties().setProperty("proxyPort", proxyPort);

    // Proxy server credentials if needed
//
//      System.getProperties().setProperty("proxy.authentication.username",
proxyUsername);
//
//      System.getProperties().setProperty("proxy.authentication.password",
proxyPassword);

    // Construct request
URL httpsFile = new URL(serverURL);
HttpsURLConnection connection =
(HttpsURLConnection)httpsFile.openConnection();
connection.setRequestMethod("GET");
connection.setRequestProperty("Content-Type", "application/x-www-form-
urlencoded");

    // Add credential to the request.
BASE64Encoder enc = new sun.misc.BASE64Encoder();
String encodedUsernameAndPassword = enc.encode((user + ":" +
password).getBytes());
connection.setRequestProperty("Authorization", "Basic " +
encodedUsernameAndPassword);

    //Send request
connection.setUseCaches(false);
connection.setDoInput(true);
connection.setDoOutput(true);
DataOutputStream wr = new
DataOutputStream(connection.getOutputStream());

```

```

        wr.flush();
        wr.close();

        // Read response and write to screen
        Scanner scanner = new Scanner(connection.getInputStream());
        String responseBody = scanner.useDelimiter("\\A").next();
        System.out.println(responseBody);
    }
}

```

The same rules apply to the parameters, so users are advised to read section 7.1.1– Basic Example.

### 7.1.3 C# Example Request

Below is an example using C# to interact with the API.

```

using System;
using System.IO;
using System.Net;
using System.Net.Http;
using CsvHelper;
using CsvHelper.Configuration;

public class Event {
    public string Symbol { get; set; }
    public string LineType { get; set; }
    public string Country { get; set; }
    public string CompanyID { get; set; }
    public string ProdConsID { get; set; }
    public string UnitID { get; set; }
    public string ControlArea { get; set; }
    public string Source { get; set; }
    public string Type { get; set; }
    public string EventID { get; set; }
    public DateTime NUMStartDate { get; set; }
    public DateTime NUMEndDate { get; set; }
    public string NUMCapacity { get; set; }
    public string NonavailabilityReason { get; set; }
    public string Remarks { get; set; }
    public DateTime TimeStamp { get; set; }
    public string Status { get; set; }
    public DateTime PublicationTimeStamp { get; set; }
    public DateTime ModificationTimeStamp { get; set; }
    public string Commodity { get; set; }
    public string Facility { get; set; }
}

```

```

        public override string ToString() {
            return base.ToString();
        }
    }

class Program {
    static void Main(string[] args) {
        var client = new HttpClient(new HttpClientHandler { Credentials =
new NetworkCredential("user", "password") });
        var resp = client.GetAsync("https://api1.datasources.eex-
group.com/getEvent?Symbol=NonUsabilityConsumptionPower&Update_Start=2018-01-
01T11:20Z&Update_End=2018-08-11T11:15Z").Result;
        if (!resp.IsSuccessStatusCode) {
            Console.Error.WriteLine(resp.StatusCode + ": " +
resp.ReasonPhrase);
            Environment.Exit(1);
        }

        using (var csv = new CsvReader(new
StreamReader(resp.Content.ReadAsStreamAsync().Result), new Configuration {
ShouldSkipRecord = arr => arr[0] == "Error" })) {
            csv.Read();
            csv.ReadHeader();
            while (csv.Read()) {
                var record = csv.GetRecord<Event>();
                Console.WriteLine(record);
            }
        }
    }
}

```

The same rules apply to the parameters, so users are advised to read section 7.1.1– Basic Example.

## 7.2 API Responses

API responses are available in two formats:

- a) JSON
- b) XML

Example responses for each type to be found in the following section.



### 7.3 JSON sample return

```
{
  "Symbol": "NonUsabilityConsumptionPower",
  "LineType": "NUCL",
  "Country": "DE",
  "CompanyID": "ENBWTRADING0",
  "ProdConsID": "V000027",
  "Commodity": "Power",
  "Facility": "Consumer",
  "UnitID": "V000027-001",
  "ControlArea": "TransnetBW",
  "Type": "Unplanned",
  "EventID": "000000000000022246422#2415_001",
  "NUMStartDate": "2018-05-13T06:24:00Z",
  "NUMEndDate": "2018-05-13T16:00:00Z",
  "NUMCapacity": 50.0,
  "NonavailabilityReason": "Outage",
  "Remarks": "",
  "TimeStamp": "2018-05-13T06:25:20Z",
  "Status": "Active",
  "PublicationTimeStamp": "2018-05-13T06:25:22Z",
  "ModificationTimeStamp": "2018-05-13T06:25:20Z"
},
```

This is one example record from the retrieved response. It is standard JSON containing name/value pairs for the returned data.

### 7.4 XML sample return

```
<Row>
  <Field Name="Symbol">NonUsabilityConsumptionPower</Field>
  <Field Name="LineType">NUCL</Field>
  <Field Name="Country">DE</Field>
  <Field Name="CompanyID">ENBWTRADING0</Field>
  <Field Name="ProdConsID">V000027</Field>
  <Field Name="Commodity">Power</Field>
  <Field Name="Facility">Consumer</Field>
  <Field Name="UnitID">V000027-001</Field>
  <Field Name="ControlArea">TransnetBW</Field>
  <Field Name="Type">Unplanned</Field>
  <Field Name="EventID">000000000000022246422#2415_001</Field>
  <Field Name="NUMStartDate">2018-05-13T06:24:00Z</Field>
  <Field Name="NUMEndDate">2018-05-13T16:00:00Z</Field>
  <Field Name="NUMCapacity">50.0</Field>
  <Field Name="NonavailabilityReason">Outage</Field>
  <Field Name="Remarks"/>
  <Field Name="TimeStamp">2018-05-13T06:25:20Z</Field>
  <Field Name="Status">Active</Field>
  <Field Name="PublicationTimeStamp">2018-05-13T06:25:22Z</Field>
  <Field Name="ModificationTimeStamp">2018-05-13T06:25:20Z</Field>
</Row>
```

This is standard XML representing the fields returned in the response.

## 7.5 Examples

### 7.5.1 getTimeSeries

- 1) <https://api1.datasource.eex-group.com/getTimeSeries?Symbol=ExPostInformationActualUnitGenerationPower&Country=BE&DaysBack=6>
- 2) <https://api1.datasource.eex-group.com/getTimeSeries?Symbol=ExPostInformationActualUnitGenerationPower&Country=DE&Start=2018-10-18%2022:00&End=2018-10-19%2021:45>
- 3) <https://api1.datasource.eex-group.com/getTimeSeries?Symbol=ExAnteInformationPlannedGenerationPower&Start=2018-10-17%2022:00&End=2018-10-18%2021:45>
- 4) <https://api1.datasource.eex-group.com/getTimeSeries?Symbol=ExAnteInformationPlannedGenerationPower&Country=NL&Start=2018-10-17%2022:00&End=2018-10-18%2021:00>
- 5) <https://api1.datasource.eex-group.com/getTimeSeries?Symbol=ExAnteInformationAvailableCapacityPower&Start=2018-10-01&End=2018-10-02>

### 7.5.2 getEvent

- 1) [https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityGenerationPower&Country=DE&Update\\_Back=1](https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityGenerationPower&Country=DE&Update_Back=1)
- 2) [https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityConsumptionGas&Country=DE&Update\\_Back=100](https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityConsumptionGas&Country=DE&Update_Back=100)
- 3) [https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityConsumptionPower&Update\\_Back=100](https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityConsumptionPower&Update_Back=100)
- 4) [https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityStoragePower&Country=DE&Update\\_Back=5](https://api1.datasource.eex-group.com/getEvent?Symbol=NonUsabilityStoragePower&Country=DE&Update_Back=5)
- 5) [https://api1.datasource.eex-group.com/getEvent?Symbol=AdhocMessages&Update\\_Back=10](https://api1.datasource.eex-group.com/getEvent?Symbol=AdhocMessages&Update_Back=10)

### 7.5.3 getMasterData

- 1) <https://api1.datasource.eex-group.com/getMasterData?Symbol=Company>
- 2) <https://api1.datasource.eex-group.com/getMasterData?Symbol=ProdCons>
- 3) <https://api1.datasource.eex-group.com/getMasterData?Symbol=Unit>
- 4) <https://api1.datasource.eex-group.com/getMasterData?Symbol=Capacity>

## 8. Known Issues

### 8.1 General known issues

- a) Occasional missing values for "ModificationTimeStamp" of historic data only

### 8.2 Symbol specific known issues

#### 8.2.1 ExPostInformationActualPlantGenerationPower

- a) incomplete data in API return for AT and DE (2014-12-31T23:00:00Z to 2015-07-24T21:00:00Z)

#### 8.2.2 ExPostInformationActualUnitGenerationPower

- a) Occasional missing historic data for inactive units only