eCAT Web Services

A Content

1. Description

The purpose of this document is to provide a description of web services (WS) available for the electronic Capacity auction Tool (eCAT) and the configuration of web service clients (WS clients) willing to communicate with eCAT.

WS enable WS clients to request data as well as submit and download documents. All operations performed by WS are synchronous. WS are secured by HTTPS communication protocol and performs client authentication and authorization.

There are two different types of WS described in this document. These two types communicate with different parts of eCAT and therefore have been developed differently.

2. Authentication

WS are secured using the following mechanisms:

- HTTPS protocol
- Username and password authentication
- Client certificate authentication

3. WS client requirements

Requirement	Description
SSL authentication by client certificate	SSL handshake is performed when client sends a request to WS. An electronic certificate and login and password authentication is required, therefore the client WS software must support SSL authentication by client certificate.
SOAP version	Messages must be in line with SOAP version 1.1 for the operations “put”, “list” and “get”. Messages must be in line with SOAP version 1.2 for “listOpenauction” and “getOfferedCapacityForauction”

4. Supported scenarios

4.1. Submission of document

Submission of a document can be done via SOAP message as binary base64 encoded strings (this applies for all supported document types). WS process all messages in the pipeline and performs document type recognition and validations among other operations. As a consequence, the operation “put” is not aware of the format or type of submitted message.

4.1.1. Operation “put”

The operation “put” allows authenticated, authorized users to submit documents. The request contains submitted file name together with its binary content (base64 encoded string).

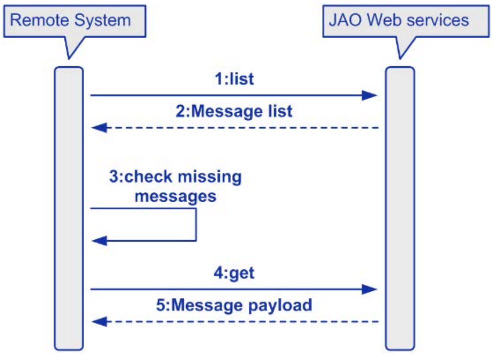
Result of the operation is returned to client as ENTSO acknowledgment v5r0 message within SOAP message (as character data).

4.1.2. Request example for “put”

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:typ="http://ca.jao.eu/types">
   <soapenv:Header/>
   <soapenv:Body>
   <!-- Please go to File - Preferences - SSL Settings and set KeyStore path
to your *.pfx certificate and fill your KeyStore Password -->
      <typ:putRequest>
         <typ:fileName>10X1001A1001A57U_RS-HR-Y-BASE-------180101-
01_1.xml</typ:fileName>
         <!-- In SoapUI: Click '+' below, select file, Select Yes to Cache,
In Part Column select CID ID-->
         <typ:payload>cid:10X1001A1001A57U_RS-HR-Y-BASE-------180101-
01_1.xml</typ:payload>
         </typ:putRequest>
   </soapenv:Body>
</soapenv:Envelope>
```

4.2. Downloading documents

The process of downloading documents is summarized in the following chart:



The steps must be executed by the WS client in the following order:

1. list – remote system will list messages for specified criteria,
2. check missing messages – remote system will check which messages weren't downloaded yet by message ID,
3. get – remote system will request download of missing messages payload by using message ID.

4.2.1. Operation "list"

The operation "list" provides WS client with the ability to get a list of all messages, which fulfils the filtering criteria. Filtering criteria parameters are a combination of code value parameters and time interval parameters.

- dataflow code (required)
- document type (optional)
- sender party EIC code (optional)
- receiver party EIC code (optional)
- domain code (EIC, optional)
- applicable time interval range (required*)
- timestamp range (required*)

Data flow code and *time range (applicable time interval OR timestamp) are the only mandatory criteria. Currently supported data flows codes and document types are listed at the table below. Please note, that dataflow and document type codes are maintained by eCAT and they can be subject to change.

The operation “list” only allows access to documents listed here below:

OUTPUT	Definition	Document type code
A_TR	approved transfer	ecan-rights-document-v4_0
AR	allocation results	ecan-allocation-results-document-v4_0
D_CUA	daily capacity usage authorization	ecan-rights-document-v4_0
I_CUA	intraday capacity usage authorization	ecan-rights-document-v4_0
LT_CUA	long term capacity usage authorization	ecan-rights-document-v4_0
MAX_AAC	maximum already allocated capacity	ecan-rights-document-v4_0
MIN_AAC	minimum already allocated capacity	ecan-rights-document-v4_0

INPUT	Definition	Document type code
BID	auction bid	ecan-bid-v4_0
RETURN	resale request	ecan-rights-document-v4_0
TR	transfer request	ecan-rights-document-v4_0

The operation “list” does not return payload messages directly, instead only message IDs are returned (please note that it is a technical message ID generated by the system). The result set is limited to 100 messages.

4.2.2. Request example for “list”

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:typ="http://ca.jao.eu/types">
   <soapenv:Header/>
   <soapenv:Body>
   <!-- Please go to File - Preferences - SSL Settings and set KeyStore path
to your *.pfx certificate and fill your KeyStore Password -->
      <typ:listRequest>
         <typ:dataFlowCode>LT_CUA</typ:dataFlowCode>
         <typ:applicableTimeFromUtc>2017-11-
03T23:00:00Z</typ:applicableTimeFromUtc>
         <typ:applicableTimeToUtc>2017-11-
04T00:00:00Z</typ:applicableTimeToUtc>
      </typ:listRequest>
   </soapenv:Body>
</soapenv:Envelope>
```

4.2.3. Response example for list when no matching criteria found

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <listResponse xmlns="http://ca.jao.eu/types"/>
   </soap:Body>
</soap:Envelope>
```

4.2.4. Operation “get”

The operation “get” is used by WS client to retrieve payload of any document using document ID. Document ID can be obtained from the operation “list” which supports criteria search. Document payload is returned as binary data (encoded as base64 string). When document ID is not found, SOAP fault message is returned. See response examples.

4.2.5. Request example for “get”

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:typ="http://ca.jao.eu/types">
   <soapenv:Header/>
   <soapenv:Body>
   <!-- Please go to File - Preferences - SSL Settings and set KeyStore path
to your *.pfx certificate and fill your KeyStore Password -->
      <typ:getRequest>
         <typ:messageId>59ec5zz6fe4b0bd788de99bca</typ:messageId>
               </typ:getRequest>
   </soapenv:Body>
</soapenv:Envelope>
```

4.2.6. Response example for “get” SOAP error

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <soap:Fault>
         <faultcode>soap:Server</faultcode>
         <faultstring>Unable to find any document by ID =
59ec5f6fe4b0bd788de51ddca</faultstring>
      </soap:Fault>
   </soap:Body>
</soap:Envelope>
```

4.3. Open auctions

The operation “listOpenauctions” allows users to request a list of auctions that are currently opened.

This WS does not require any parameter. If no auction is running, the WS will explicitly return that no auction is open.

4.3.1. Request example for Request example for “listOpenauctions”

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:eter="http://eterra.ca.market.webservice.service.TraderServiceWS">
   <soap:Header>
      <wsse:Security soap:mustUnderstand="true"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
         <wsse:UsernameToken wsu:Id="UsernameToken-2">
            <wsse:Username>YourUsername</wsse:Username>
            <wsse:Password Type="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-username-token-profile-
1.0#PasswordText">YourPassword</wsse:Password>
         </wsse:UsernameToken>
      </wsse:Security>
   </soap:Header>
   <soap:Body>
      <eter:listOpenAuctions/>
   </soap:Body>
</soap:Envelope>
```

4.4. Offered capacity

The operation “getOfferedCapacityForauction” allows users to request the offered capacity of a particular auction.

The WS will use the target auction ID as an entry parameter and will return:

- If the auction ID doesn’t correspond to an auction within the system → an error message stating the auction is unknown
- If the bid submission gate is not open → an error message stating the auction is not open
- If the bid submission gate is open → the offered capacity of the target auction (in auction specification format).
- If the bid submission gate is closed → the offered capacity of the target auction (in auction specification format) and a message stating the auction is closed.

4.4.1. Request example for “getOfferedCapacityForauction”

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:eter="http://eterra.ca.market.webservice.service.TraderServiceWS">
   <soap:Header>
      <wsse:Security soap:mustUnderstand="true"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
         <wsse:UsernameToken wsu:Id="UsernameToken-2">
            <wsse:Username>YourUsername</wsse:Username>
            <wsse:Password Type="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-username-token-profile-
1.0#PasswordText">YourPassword</wsse:Password>
         </wsse:UsernameToken>
      </wsse:Security>
   </soap:Header>
   <soap:Body>
      <eter:getOfferedCapacityForAuction>
         <!--Optional:-->
         <arg0>AT-CH-D-DAILYPRODU-180203-01</arg0>
      </eter:getOfferedCapacityForAuction>
   </soap:Body>
</soap:Envelope>
```

5. WSDL

The web services are described in technical detail in the following WSDL.

5.1. For the operations “put”, “list”, “get”

The WSDL is available online here.

5.2. For the operations “listOpenauction” and “getOfferedCapacityForauction”

The WSDL is available online here.

END OF DOCUMENT