**



SEE CAO

Damas

Web Services Interface





Unicorn © 2013 – Unicorn Systems a.s.

Jankovcova 1037/49, CZ – 170 00 Prague 7

| Project: |  | SEE CAO |
| --- | --- | --- |
| Project – Subject: |  | Damas |
| Document Title: |  | Web Services Interface |
|  |  |  |
| Author: |  | Dušan Nulíček |
|  |  |  |
| Contact: |  | E-mail: info@unicornsystems.eu |
|  |  | Tel.: (+420) 221 400 111 |

# List of Contents

1.  List of Contents 3

2.  Revision History 5

3.  Introduction 6

3.1  Document Purpose 6

3.2  Document Organization 6

4.  Web Services Description 7

4.1  What Are Web Services? 7

4.2  Benefits 7

4.3  Standards 7

5.  Web Services Interface 9

5.1  Transfer Technology 9

5.2  Data Format 9

5.3  Interface of Damas Web Services 10

5.4  Client Application Development Guideline 10

5.4.1  Development prerequisites 11

5.4.2  Production prerequisites 11

5.4.3  Best practices of the client implementation 12

5.5  SOAP 12

5.6  SOAP Message 12

5.6.1  Input Parameters 13

5.6.2  Output Parameters 14

5.6.3  Error Handling 15

5.7  WSDL 17

5.7.1  Synchronous Request 17

5.7.2  Asynchronous Request 19

5.7.3  Asynchronous Request State 21

5.7.4  Current Date and Time 22

6.  Web Service Security 24

6.1  General Description 24

6.1.1  Security 24

6.1.2  PKI Infrastructure 24

6.2  Damas Security Model 24

6.3  SOAP Request Preparation 25

6.3.1  SOAP Request Description 25

6.4  SOAP Response Parsing 26

6.4.1  SOAP Response Description 27

6.5  Error Handling 27

7.  Data Flows 28

7.1  List of Data Flows 28

7.2  Data Flows for Sending Data 29

7.2.1  Submit/Modify Auction Bids 29

7.2.2  Submit/Modify Capacity Transfers 31

7.2.3  Submit/Modify Capacity Resales 33

7.3  Data Flows for Data Download 35

7.3.1  Download Offered Capacity 35

7.3.2  Download Capacity Rights 36

7.3.3  Download Detailed Auction Results 37

7.3.4  Download Current Date and Time 38

7.3.5  Download Long-term Offered Capacity and Planned Outage Periods 39

7.3.6  Download Auction Specification (CASD) 39

8.  XSD Schemas 41

8.1  List of the XSD Schemas 41

8.1.1  ENTSO-E XSD Schemas 41

8.1.2  Damas specific XSD schemas 41

8.2  Description of ENTSO-E XSD Schemas 42

8.2.1  Capacity Document 42

8.2.2  Bid Document 45

8.2.3  Allocation Result Document 54

8.2.4  Rights Document 60

8.2.5  Acknowledgement Document 72

8.3  Description of Damas specific XSD Schemas 75

8.3.1  Long-term OC Information Document 75

8.4  Code Lists 76

8.4.1  Role 76

8.4.2  Capacity Contract Type 76

8.4.3  Measurement Unit 76

8.4.4  Message Type 77

8.4.5  Process Type 77

8.4.6  Classification Type 78

8.4.7  Coding Scheme 78

8.4.8  Business Type 78

8.4.9  Object Aggregation 78

8.4.10  Currency 79

8.4.11  Reason Codes 79

8.4.12  Curtailment Types 81

8.4.13  Invoicing Data Types 81

8.4.14  Invoice States 81

9.  Appendix A – Examples of Xml Messages 82

10.  Appendix B – Examples of SOAP Request Messages 83

11.  Appendix C – Configuration of SOAMOA 84

12.  Open Issues 85

12.1  Closed Issues 85

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| 00.01 | 15.11.2013 | Dusan Nulicek | Initial version |
| 00.02 | 25.11.2013 | Dusan Nulicek | Minor changes |
| 01.00 | 25.11.2013 | Dusan Nulicek | Document ready for review |
| 01.01 | 05.12.2013 | Dusan Nulicek | Support for indivisible bids |
| 02.00 | 05.12.2013 | Martin Hradil | Document published for client review. |
| 02.01 | 04.07.2014 | Zdenek Janda | Document finalization |
| 02.02 | 02.02.2017 | Klára Maiová | 3rd enhancement added |
| 02.03 | 15.06.2017 | Jindra Marvalová | WS Downloading Auction Specification (CASD) |
| 03.00 | 29.06.2017 | Ivan Malý, Klára Maiová | Document finalization |
| 03.01 | 25.09.2017 | David Pelikán | Minor changes |

# Introduction

## Document Purpose

This document is specification for general approach that must be taken when accessing Damas web service. Web services are built on industry standard technologies. They are available on the Internet and ensure the same level of the privacy and security as Damas web site.

## Document Organization

Chapter 4 provides brief introduction to web services with emphasis on web service properties and existing standards. Chapter 5 focuses on explaining of the implemented Damas web service interface and its use for automated communication with other systems. In addition, detailed technical information about the implemented SOAP format and WSDL is presented in this chapter. Main subject of the Chapter 6 is security of the web services and data transfers. Comprehensive description of all existing data flows for downloading and uploading data from/to Damas is included in Chapter 7. Detailed technical description of the web services is given in Chapter 8 that contains specification and explanation for all used XSD schemas.

# Web Services Description

## What Are Web Services?

Web services are cornerstone of the movement towards distributed computing on the Internet. Open standards, focus on communication and collaboration of people and applications have created an environment where are web services a platform for application integration. Applications are constructed using multiple web services from various sources that work together regardless of where they are located or how they were implemented.

## Benefits

Web services selected as platform to enable integration of Damas with other systems represent up-to-date and high-performance technology and have following major advantages:

* Full platform independence
* Easy implementation and consumption
* Strong support for development of client applications
* Web services simplify process of the data preparation and processing as a human end user is replaced by end an application. Obviously, human factor is not fully eliminated from process but end user can move burden of his/her requirements to application. End user submits his/her requirements to this application and requests all data necessary to make decisions. Users do not have to care how application handles or acquires data. Application handles entire communication with Damas.

Key advantages for user include:

* Elimination of the routine and burdensome communication with Damas - It is not necessary to login to system, clicking and selecting file, waiting for confirmation that file was received, etc.
* Integration of all user activities into single system - If company has software designed to facilitate trading with electricity, then it’s likely that data related to Damas will be processed by such system. In such case, data do not have to be exported and transferred to Damas in complicated way, but after creating simple communication component, they can be forwarded to proper location by simple click.

Of course, described method can be combined with existing conventional approach. User can always decide how to communicate with Damas to achieve his/her goal.

## Standards

Web services are applications whose logic and functions are accessible using the standard Internet protocols and data formats, such as Hypertext Transfer Protocol (HTTP) and Extensible Markup Language (XML).

The XML web services provide useful functionality to web users through the Simple Object Access Protocol (SOAP), which is essential standard for information exchange in decentralized distributed environment. It is XML-based protocol that consists of the envelope that defines framework for describing what is in message and how to process it and this protocol also consists of convention for representing remote procedure calls and responses.

Web services provide way how to describe their interfaces in sufficient detail to allow user to build client application to communicate with them. This description is provided in XML documents called Web Services Description Language (WSDL) documents. WSDL is XML format for describing network services as a set of the endpoints operating on messages containing either document-oriented or procedure-oriented information.

# Web Services Interface

This chapter provides overview and explanation of the major properties of Damas web services implementation. In first part of this chapter, principles of the transfer technology are introduced and structure of the data formats is briefly described. Second part contains instructions for calling web services and short guideline on using description of Damas web services interface to develop client application. Finally, technical information and detailed description of the web service implementation are provided at the end of this chapter. Firstly, structure of the SOAP message used in Damas is introduced and explained. After that, the WSDL description of the request and response SOAP message formats for all web services is presented.

## Transfer Technology

Web services in Damas can be used for automated data exchange or for machine-controlled data exchange. Use of this technology significantly simplifies communication among Damas and Market Participants.

Main transfer unit is a text file containing the SOAP XML message. Format of the SOAP message in Damas was designed according to the SOAP 1.1 and SOAP 1.2 specification recommended by W3C ([http://www.w3.org/TR/2000/NOTE-SOAP-20000508/](http://www.w3.org/)).

Supported communication protocol is HTTPS. Common authentication process, containing login name and password, is defined. Login and password must be sent with each SOAP message so that message could be processed. All actions performed using web services are executed in Damas with permissions of the user whose credentials are provided in the SOAP message.

Credentials must be provided in form of the Username token in accordance with Web Services Security specification. For details of the Web Services Security, see [http://www.oasis-open.org/committees/tc\_home.php ?wg\_abbrev=wss](http://www.oasis-open.org/committees/tc_home.php%20?wg_abbrev=wss).

## Data Format

Every web service message used in Damas consists of two parts:

* Header of the web service message
* Body of the web service message.

For all data flows designed for sending data to Damas, the XML file containing business data to be transferred is included in body of the web service message. Structure of the XML file is defined by XSD charts, which make it possible to validate semantics of the XML message. The XML files used in Damas are implemented according to the ESS standard v3r1 and ECAN v4r0 with Code List v6r4 (<https://www.entsoe.eu/publications/electronic-data-interchange-edi-library/>). All XSD definitions specifying formats of the XML files to be exchanged are listed in attachment of this document.

For successful data exchange, it is necessary to synchronize mechanism of the entity (auction participants and their partners) identification in order to match scheduling charts. Damas uses EIC codes standardized by ENTSO-E to identify entities and their partners abroad.

## Interface of Damas Web Services

Damas web services are accessible at following addresses:

|  |  |  |  |
| --- | --- | --- | --- |
| Environment | Address | Protocol | Port |
| Testing | https://test-auctions.seecao.com/DamasService.svc | https | 443 |
| https://test-auctions.seecao.com/DamasService2.svc | https | 443 |
| Production | https://auctions.seecao.com/DamasService.svc | https | 443 |
| https://auctions.seecao.com/DamasService2.svc | https | 443 |

Following web service interfaces are implemented in Damas to provide communication with neighbouring systems:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | SOAP request | SOAP response | Description |
| [Synchronous Request](#_Synchronous_request) | RunSynchrous | RunSynchrous Response | It provides synchronous exchange of the commercial data with Damas. |
| [Asynchronous Request](#_Asynchronous_Request_1) | RunAsynchrous | RunAsynchrous Response | Itprovides asynchronous exchange of the commercial data with Damas. |
| [Asynchronous Request Stat](#_Asynchronous_Request_State_1)us | **CheckRQResult** | **CheckRQResult Response** | It returns status of the asynchronous request that is being processed by Damas. |
| [Current Date and Time](#_Current_Date_and_Time) | **GetActualDateTime** | **GetActualDateTime Response** | It returns current system date and time that is important for automatic operations carried out by the system. |

Damas web services can be used in either synchronous or asynchronous mode:

* **Synchronous call of the web service -** Data are passed to web service via the RunSynchrous method. By performing this step synchronous request is established within Damas and processed, and result is returned back to client application. Output parameter of this web service is XML with structure varying for individual data streams.
* Asynchronous call of the web service - Data in this case are passed to web service via the RunAsynchrous method. Request is established within Damas as asynchronous. In this case, output of the web service does not include processing of the established request but rather it contains only ID of the request. This ID is used to request result of such request later on.

## Client Application Development Guideline

This is brief manual of the client application implementation based on information available in description of Damas web service interfaces.

In first step, way how client application communicates with web service must be decided. Generally, there are two options:

1. to use the SOAP standard by means of the HTTP/HTTPS protocol

To create request in XML in compliance with the SOAP standard and to build client application capable of sending this XML as HTTPS request to web server. In addition, it is necessary to implement functionality capable of web service reply processing. Description of the web service interface also includes description of the SOAP requests and replies.

1. to create proxy class based on the WSDL interface description

Description of Damas web service interface also includes description of the WSDL interface. The WSDL is XML standard that is designed to describe arbitrary web service. Current development platforms can generate source code based on the WSDL document. Result is usually class, which allows handling of the web service as object. There is no need to implement actual communication protocol as development environment does this for you. Examples of such platforms supporting this kind of functionality include Visual Studio .NET and J2EE.

Communication with Damas web service interface takes place via secured SSL channel; client authenticates himself using valid login and password identical to those used for accessing Damas web platform. If subject implementing WS client applies method for automatic source code generation based on the WSDL document, then this code probably must be extended to include such functions.

### Development prerequisites

* Testing Environment is intended to be used for integration tests
* Testing Environment will support, in case of the WS, both ways of the authentication:
  1. Username and password over HTTP
  2. Username, password and client certificate (X509) over HTTPS

Access using HTTP will be needed at start of the implementation of the client before last step, i.e. logon using HTTPS and signing of the messages using client certificate will be additionally implemented as well.

* User accounts for Testing Environment will be issued by the SEE CAO as part of the standard procedure of the creating new Damas user. Result of the procedure will be one account for basic logon using HTTP with username and password and one account for target authentication using HTTPS, username, password and client certificate.
* The SEE CAO will perform support providing users with requested data (i.e. to simulate various system states to allow users to build robust solution).

### Production prerequisites

* Production Environment requires authentication using username, password and client certificate (X509) over HTTPS
* User accounts for the Production Environment will be issued by the SEE CAO as part of standard procedure of the creating new Damas user. Every created user account will have granted rights for WS functionalities.

### Best practices of the client implementation

As first step it is the best way implementation of the synchronous call for action Current Date and Time – this functionality serves as simple check of the general functionality of the WS. It is also the easiest implementation of the interface to Damas Web services. It is not connected to any business, but helps solving initial technical issues, authentication problems, etc.

Recommended list of the development steps:

1. Get example request SOAP message from [APPENDIX A – EXAMPLES OF UPLOADING XMLS](#APPENDIX_A). Replace sample username and password according to chapter Security
2. Modify value in tag <wsu:Expires> to be in future, otherwise message would be rejected by server
3. Before any implementation try to send message using some tool like SoapUI (free tool used to send SOAP messages to WS, for details please see Appendix C – Configuration of SoapUI ). This will ensure you have the valid SOAP message.

After you have completed successfully initial step ensuring you have valid request message you can continue with implementation of the web service client providing the same type of the message as created manually before.

If this all has been done successfully, authentication may be extended with client certificate – see chapter Web Service Security. Then the HTTPS access using SSL is required.

Next step is implementation of the additional necessary actions (i.e. sending bids or nominations, etc.) which are similar to basic one.

In case of the need, you can implement also asynchronous call using the RunAsychrous actions and the CheckRQResult actions.

## SOAP

Structure of the SOAP message is implemented according to the SOAP 1.1 specification recommended by W3C (<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>).

## SOAP Message

The SOAP message implemented in Damas consists of the SOAP header and body.

UTF-8 encoding is required for all SOAP messages passed into Damas. All outgoing messages are UTF-8 encoded as well.

The SOAP header contains information that is essential for user authorization, such as the user's login name and password.

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

The WSS Security Header contains security tokens necessary to authenticate sender and check message integrity. These tokens are user credentials. For details of the WSS Security see Chapter [Web Services Security.](#_Web_Service_Security)

Body of the SOAP message includes element, which contains input/output parameter class. Element name is derived from name of the web service that is used.

<soap:Body>

<**WebServiceName** xmlns="http://auctions.seecao.com/wse">

**Input/Output Parameters**

</**WebServiceName**>

</soap:Body>

For details of the WSS Security, Header, see chapter Web Services Security.

### Input Parameters

The parameter class defined for input parameters:

<Input>

<FID>**FID**</FID>

<Parameters>

<**XXXParam** Name="**param\_name1**">**param\_val1**</**XXXParam**>

<**XXXParam** Name="**param\_name2**">**param\_val2**</**XXXParam**>

…

<**XXXParam** Name="**param\_nameN**">**param\_valN**</**XXXParam**>

</Parameters>

</Input>

The highlighted parameter shall be replaced by values according to following rules:

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | Note |
| **FID** | String | Identification of the dataflow. See chapter Data flows. | Unique for each data flow. |
| **XXXParam** |  | Element name of the parameter represents its data type. For overview of the supported data types see table below. | Depends on the data flow. |
| **param\_nameX** | String | Name of data flow input parameter | Depends on the data flow. |
| **param\_valX** | String, Number, Date | Value of data flow input parameter | Depends on the data flow. |

List of the input parameter data types:

|  |  |  |
| --- | --- | --- |
| Data type element name | Corresponding XSD type | Example |
| BooleanParam | xs:Boolean | True |
| DateParam | xs:date | 2002-09-24 |
| DateTimeParam | xs:dateTime | 2002-09-24T09:30:10Z |
| DecimalParam | xs:decimal | 999.50 |
| IntParam | xs:int | 999 |
| StringParam | xs:string | TEXT |
| XmlParam | Any XML node tree (corresponds to <xs:any> XSD element). | Any XML node |

Elements with data flow input parameters (XXXParam) must be alphabetically ordered by their type names (that is <BooleanParam> elements come first, <DateParam> elements come second etc.).

### Output Parameters

The parameter class defined for output parameters:

<Output>

<RQID>**RQID**</RQID>

<Result>**resultXML**</Result>

<RQState>

<Code>**RQState\_Code**</Code>

<Description>**RQState\_Description**</Description>

</RQState>

</Output>

The highlighted parameter shall be replaced by values according to following rules:

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | Note |
| **RQID** | Number | Unique identification of the asynchronous request in Damas |  |
| resultXML | String | Contains result of the request | Depends on the data flow; see data flows description |
| RQState\_Code, | String | Code of the state of the request. (For list of the possible codes see Chapter SOAP CheckRQResultResponse). | For synchronous requests the RQState\_Code value is always“COMPLETED” |
| RQState\_Description | String | Description of the state the request. |  |

### Error Handling

Errors returned by Damas web services interface are divided into two basic groups:

* Business errors – These errors originate in business control algorithms and it express that imported business data violates business rules. This applies only to input data flows (see Chapter [Data Flows for Sending Data](#_Data_Flows_for_Sending Data)). These errors are returned in form of the Acknowledgement as standard output of the data flow (see Chapter [Output Parameters](#_Output_Parameters)) and therefore are not subject of this chapter.
* System errors – These errors represent non-business faults. This includes user authentication errors, bad format of the SOAP xml, input parameters etc. These errors should be handled by client applications. System errors are listed below.

Errors are distributed to the client by using <soap:Fault> element, as defined in SOAP/1.1 specification (see <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/#_Toc478383507> ).

Detailed information about error is carried in the <Error> element (see example of the SOAP fault below):

<soap:Fault>

<faultcode>**faultcode**</faultcode>

<faultstring>**faultstring**</faultstring>

<detail>

<e:Error xmlns:e="http://auctions.seecao.com/xsd/errors.xsd">

<ErrID>**errID**</ErrID>

<ErrDescr>**errDescr**</ErrDescr>

<ErrXML>**errXML**</ErrXML>

</e:Error>

</detail>

</soap:Fault>

The highlighted parameter shall be replaced by values according to following rules:

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | Note |
| **faultcode** | String | Code of the error as specified in SOAP/1.1. |  |
| **faultstring** | String | Description of the error as specified in SOAP/1.1. |  |
| **ErrID** | Number | Identification number of the error. | See List of Standard Errors for more information. |
| **ErrDescr** | String | Short description of the error. | See List of Standard Errors for more information. |
| **ErrXML** | XML | Additional debug information are not intended to be processed by client applications. |  |

The <e:Error> element doesn’t have to be present in the Fault message. It is present only for errors with faultcode of the “soap:Client” value or “soap:Server” value (see link <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/#_Toc478383507> for details on faultcode).

Errors resulting from sender's identity and message integrity checks are returned to client application according to the WSS standard (see <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>, Chapter Error Handling).

List of the Standard Errors:

|  |  |  |
| --- | --- | --- |
| Error ID | Error Description | Fault Code |
| -130 | User is not authorized for requested data stream. | soap:Client |
| -501 | Date is invalid. | soap:Client |
| -502 | Unknown entity code. It concerns EIC codes describing parties | soap:Client |
| -506 | Unknown Control Area code. | soap:Client |
| -507 | Not existing auction with specified code | soap:Client |
| -508 | Unknown or invalid capacity type code. | soap:Client |
| -510 | Data flow with requested FID does not exist. | soap:Client |
| -512 | Invalid XML format. Errors occurred during XSD validation of the submitted data XML. | soap:Client |
| -513 | Invalid data flow input parameters | soap:Client |
| -514 | Internal server error | soap:Server |
| -515 | Requested data has not been published yet. | soap:Client |
| -516 | Requested date range is not valid | soap:Client |
| -517 | Asynchronous request does not exist | soap:Client |
| -518 | Requested operation is not permitted for this data flow | soap:Client |
| -520 | User is not authorized to access data of another entity. | soap:Client |
| -521 | OutArea (InArea) must be existing code. | soap:Client |
| -522 | Specified area do not specify border direction or the border direction is invalid | soap:Client |
| -523 | Not existing or invalid nomination capacity type code | soap:Client |

## WSDL

This part of the document contains description of all web services provided by Damas as interface for automatic communication with other system.

### Synchronous Request

This web service ensures synchronous exchange of the commercial data with Damas.

#### SOAP RunSynchrous

The SOAP request format for establishing the synchronous request in Damas

POST /wse/DamasService.asmx HTTP/1.1

Host: auctions.seecao.com

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

SOAPAction:“http://auctions.seecao.com/wse/RunSynchrous"

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<RunSynchrous xmlns="http://auctions.seecao.com/wse">

<Input>

<FID>**FID**</FID>

<Parameters>

<**XXXParam** Name="**param\_name1**">**param\_val1**</**XXXParam**>

<**XXXParam** Name="**param\_name2**">**param\_val2**</**XXXParam**>

…

<**XXXParam** Name="**param\_nameN**">**param\_valN**</**XXXParam**>

</Parameters>

</Input>

</RunSynchrous>

</soap:Body>

</soap:Envelope>

#### SOAP RunSynchrous Response

The SOAP response format with result of the synchronous request returned from Damas:

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<RunSynchrousResponse xmlns="http://auctions.seecao.com/wse">

<Output>

<RQID>-1</RQID>

<Result>**resultXML**</Result>

<RQState>

<Code>

COMPLETED

</Code>

<Description>

The request is completed.

</Description>

</RQState>

</Output>

</RunSynchrousResponse>

</soap:Body>

</soap:Envelope>

For details of the WSS Security, Header, see Chapter [Web service security](#_Web_Service_Security). Note that <RQID> element in this case contains -1 (id of the request is not returned for synchronous requests).

### Asynchronous Request

This web service **en**sures the asynchronous exchange of the commercial data with Damas. The RunSynchrous and RunAsynchrous methods use almost identical formats of the SOAP request and response. Asynchronous call is used for uploading larger XML (more than 2 timeseries).

#### SOAP RunAsynchrous

The SOAP request format for establishing the asynchronous request in Damas

POST /wse/DamasService.asmx HTTP/1.1

Host: auctions.seecao.com

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

SOAPAction:“http://auctions.seecao.com/wse/RunAsynchrous"

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<RunAsynchrous xmlns="http://auctions.seecao.com/wse">

<Input>

<FID>**FID**</FID>

<Parameters>

<**XXXParam** Name="**param\_name1**">**param\_val1**</**XXXParam**>

<**XXXParam** Name="**param\_name2**">**param\_val2**</**XXXParam**>

…

<**XXXParam** Name="**param\_nameN**">**param\_valN**</**XXXParam**>

</Parameters>

</Input>

</RunAsynchrous>

</soap:Body>

</soap:Envelope>

#### SOAP RunAsynchrous Response

The SOAP response format with result of the asynchronous request returned from Damas

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<RunAsynchrousResponse xmlns="http://auctions.seecao.com/wse">

<Output>

<RQID>**RQID**</RQID>

<Result/ >

<RQState>

<Code>

REGISTERED

</Code>

<Description>

The request is registered for execution.

</Description>

</RQState>

</Output>

</RunAsynchrousResponse>

</soap:Body>

</soap:Envelope>

For details of the WSS Security, Header, see Chapter [Web service security](#_Web_Service_Security). Note that <Result> element is in this case empty (or missing); result is not available at this moment, only ID of the asynchronous request is returned (RQID). You can check request result later by calling CheckRQResult method (see Chapter [Asynchronous Request State](#_Asynchronous_Request_State_1) for details).

### Asynchronous Request State

This web service returns status of the asynchronous request that is being processed in Damas. The asynchronous request is identified by request ID which can be obtained by calling RunAsynchrous method (see Chapter [Asynchronous Request](#_Asynchronous_Request_1)).

#### SOAP CheckRQResult

The SOAP request format for downloading a status of the asynchronous request from Damas

POST /wse/DamasService.asmx HTTP/1.1

Host: auctions.seecao.com

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

SOAPAction:“http://auctions.seecao.com/wse/CheckRQResult"

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<CheckRQResult xmlns="http://auctions.seecao.com/wse">

<RQID>**RQID**</RQID>

</CheckRQResult>

</soap:Body>

</soap:Envelope>

For details of the WSS Security, Header, see Chapter [Web service security](#_Web_Service_Security). Note that highlighted item RQID must be replaced with ID of the existing asynchronous request.

#### SOAP CheckRQResultResponse

The SOAP response format with status of the asynchronous request returned from Damas

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<CheckRQResultResponse xmlns="http://auctions.seecao.com/wse">

<Output>

<RQID>**RQID**</RQID>

<Result>**resultXML**</Result>

<RQState>

<Code>**RQState\_Code**</Code>

<Description>**RQState\_Description**</Description>

</RQState>

</Output>

</CheckRQResultResponse>

</soap:Body>

</soap:Envelope>

For details of the WSS Security, Header, see Chapter [Web service security](#_Web_Service_Security). Element <RQState> contains information about state of the asynchronous request. Following table contains overview of the possible request states:

|  |  |  |
| --- | --- | --- |
| Code | Description | Note |
| REGISTERED | Asynchronous request is registered for execution. | <Result> element is empty; you should check request state later. |
| COMPLETED | Asynchronous request is completed. | <Result> element is filled with result of asynchronous request. |
| RUNNING | Asynchronous request is not completed. | <Result> element is empty; you should check request state later. |
| ERROR | Error occurred while running asynchronous request. | Internal server error occurred; in this case you should contact the system administrator. |

### Current Date and Time

This web service returns current system date and time that is important for automatic operations carried out by system. This service is also accessible via the RunSynchrous web service as data flow with ID“GETDATETIME”.

#### SOAP GetActualDateTime

The SOAP request format for downloading current date and time from Damas

POST /wse/DamasService.asmx HTTP/1.1

Host: auctions.seecao.com

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

SOAPAction:“http://auctions.seecao.com/wse/GetActualDateTime"

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<GetActualDateTime xmlns="http://auctions.seecao.com/wse" />

</soap:Body>

</soap:Envelope>

#### SOAP GetActualDateTime Response

The SOAP response format with current date and time returned from Damas

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: **length**

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing" >

<soap:Header>

<!-- WSS Security Header -->

</soap:Header>

<soap:Body>

<GetActualDateTimeResponse xmlns="http://auctions.seecao.com/wse">

<Output>

<RQID>-1</RQID>

<Result>**resultXML**</Result>

<RQState>

<Code>

COMPLETED

</Code>

<Description>

The request is completed.

</Description>

</RQState>

</Output>

</GetActualDateTimeResponse>

</soap:Body>

</soap:Envelope>

# Web Service Security

## General Description

### Security

Secure communication is integral part of the securing your distributed application to protect sensitive data, including credentials, passed to and from your application. Applications sharing sensitive business data among several different organizations across the Internet bring up challenges associated with robust mechanism for keeping confidentiality of the exchanged information based on world-wide encryption standards.

### PKI Infrastructure

Encryption translates data from intelligible format to unintelligible one. Decryption is process in reverse. Confidentiality it is achieved if encrypted data can be decrypted only by intended reader. To achieve this goal in scenario covering many organizations, the most proper way is using asymmetric encryption. Asymmetric encryption uses public and private key pairs. Data becomes confidential when sender encrypts it with the public key of the intended recipient and receiver decrypts it with the private key. Rules and methods concerning working with the public and private keys defines the public key infrastructure (PKI)

In the public key infrastructure, subject (that is entity) whose identity is of the significant value is assigned pair of the cryptographic keys. The public key is published to general public. The private key is secret, known only to assignee. Relationship between these two keys is such that data encrypted by one key can be decrypted only by corresponding key. It is almost impossible to obtain the private key from the public key. Certificates are used to ensure validity of the public keys. Key management is simple as long as the public key is certifiable by the reputable certificate authority (CA).

## Damas Security Model

Damas security model is built upon system of the user accounts. For each user who wants to use system services, user account must be created first, with following security elements assigned:

1. Username and password;
2. X509 certificate required for establishing the SSL communication.

Certificate for establishing the SSL communication can (but does not have to) be the same as certificate used to sign outgoing messages. These certificates are issued for particular person.

Damas web service interface security is implemented in accordance with the Web Services Security standard (see [http://www.oasis-open.org/committees/tc\_home  
.php?wg\_abbrev=wss](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss)).

Based on this standard, following security issues are addressed:

1. Transferring credentials (username and password) in the SOAP requirements;
2. Encrypted communication is ensured by the HTTPS (HTTP over SSL) protocol. Because of this fact, the SOAP requirements/responses are not encrypted further, using procedures described in the WSS specification.

## SOAP Request Preparation

In addition to web service input parameters, the SOAP request also includes authentication data of Damas user account.

### SOAP Request Description

The SOAP request format with user authentication information:

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"

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" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing">

<soap:Header>

<wsa:Action>**Damas\_soap\_method**</wsa:Action>

<wsa:To>**http\_addressing\_url**</wsa:To>

<wsse:Security soap:mustUnderstand="1">

<wsse:UsernameToken xmlns:wsu="http://docs.oasis-

open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"

wsu:Id="**username\_token\_id**">

<wsse:Username>**username**</wsse:Username>

<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-

200401-wss-username-token-profile-

1.0#PasswordText">**password\_hash**</wsse:Password>

<wsse:Nonce>**nonce\_value**</wsse:Nonce>

<wsu:Created>**utc\_datetime**</wsu:Created>

</wsse:UsernameToken>

</wsse:Security>

</soap:Header>

<soap:Body wsu:Id="**soap\_body\_id**">

<RunSynchrous xmlns="http://auctions.seecao.com/wse">

<!-- Input parameters comes here -->

</RunSynchrous>

</soap:Body>

</soap:Envelope>

#### Description of <wsa:To> element

This element contains addressing point in Damas infrastructure.

|  |  |
| --- | --- |
| XML Element | Description |
| To | Testing environment:  https://test-auctions.seecao.com/wse/DamasService.asmx  Production environment  https://auctions.seecao.com/wse/DamasService.asmx |

#### Description of <wsa:Action> element

This element defines the SOAP action name. It contains name of the SOAP method – depends whether the synchronous or asynchronous mode is used, if user requests the asynchronous request result or if user requests for actual date and time.

|  |  |
| --- | --- |
| XML Element | Description |
| Action | http://auctions.seecao.com/wse/GetActualDateTime  http://auctions.seecao.com/wse/RunSynchrous  http://auctions.seecao.com/wse/RunAsynchrous  http://auctions.seecao.com/wse/CheckRQResult |

#### Description of <wsse:Security> element

According to the WSS all security tokens are included in the wsse:Security element. This element is part of the SOAP header and consists of user authentication information (username and password).

#### Description of <wsse:UsernameToken> element

This element contains username and password assigned to the relevant Damas user account.

|  |  |
| --- | --- |
| XML Element | Description |
| Username | Login name for Damas user account |
| Password | Password for Damas user account. Password is not transferred directly, but rather its MD5 hash is transferred encoded in BASE64 format. |
| Password/@Type | Type of used UsernameToken; must be always“<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText>”. |
| Nonce | This element must be filled with a random value. This value is used as countermeasure against replay attacks. Server maintains a cache of used nonces. When client tries to use the same nonce more than once, the server will raise SOAP fault. Nonce value must be BASE64 encoded. |
| Created | Date and time of Nonce creation. Nonces older than 10 minutes are automatically rejected. Date and time must be in UTC time. (Example: 2005-05-30T09:30:10Z). |

## SOAP Response Parsing

The server SOAP response is not signed using digital certificate. Unlike in the SOAP request, no authentication data of Damas user account are transferred in this case.

### SOAP Response Description

The SOAP response format:

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"

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">

<soap:Header>

<wsse:Security soap:mustUnderstand="1">

</wsse:Security>

</soap:Header>

<soap:Body wsu:Id="**soap\_body\_id**">

<RunSynchrousResponse xmlns="http://auctions.seecao.com/wse">

<!-- Output parameters come here -->

</RunSynchrousResponse>

</soap:Body>

</soap:Envelope>

Provided example is similar to request example from Chapter SOAP Request Description. For detailed description of each element, see Chapter [SOAP Request Description.](#_SOAP_Request_Description)

The SOAP response differs from the SOAP request in following points:

Credentials are not sent back to client – element <wsse:UsernameToken> or the whole <soap:Header> element is missing.

## Error Handling

Errors resulting from sender's identity and message integrity checks are returned to client application according to the WSS standard (see <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>, Chapter Error Handling).

# Data Flows

This chapter provides description of all data flows for downloading and uploading data from/to Damas. The summary overview of the input and output data flows is presented in first part. Second part of the chapter deals with detailed definition of the data flows. Each data flow is described in detail, with input and output parameters explained.

## List of Data Flows

Following table shows various data flows that are available for uploading data into Damas. User types allowed to use data flows are indicated in “User” column in overview below.

|  |  |  |  |
| --- | --- | --- | --- |
| Data Flows | FID | Description | User |
| Submit / Modify Auction Bids | DMSWS\_BID\_IN | Enables uploading of the long-term and daily bids. Bids are being uploaded per auction, and per Market Participant.  XML – Bid Document | Trader, Border Administrator |
| Submit / Modify Capacity Transfers | DMSWS\_TRA\_IN | Enables uploading of the long-term and daily capacity transfers.  XML – Rights Document | Trader, Border Administrator |
| Submit / Modify Capacity Resales | DMSWS\_RES\_IN | Enables uploading of the explicit resales from long-term auction to subsequent long-term auction.  XML – Rights Document | Trader, Border Administrator |

Following tables show various data flows available for downloading data from Damas. User types allowed to use data flows are indicated in “User” column in overview below.

|  |  |  |  |
| --- | --- | --- | --- |
| Data Flows | FID | Description | User |
| Download Actual Date and Time | GETDATETIME | Downloading current date and time in Damas.  No particular XML – system data flow. | Trader Border Administrator TSO |
| Download Offered Capacity | DMSWS\_ATC\_OUT | Enables downloading offered capacity values for Daily auction.  Input parameters:  Auction ID  XML – Capacity Document | Trader Border Administrator  TSO |
| Download Detailed Auction Results | DMSWS\_DAR\_OUT | Enables downloading detailed auction results of the individual auctions (long-term, daily) per auction and an Market Participant.  Input parameters:  Auction ID  Market Participant  XML – Allocation Result Document | Trader Border Administrator  TSO |
| Download Capacity Rights | DMSWS\_ENT\_OUT | Enables downloading Capacity Rights per Delivery Day, Border direction, and a Market Participant.  Delivery Day  OutArea  InArea  Market Participant  XML – Rights Document | Trader Border Administrator  TSO |
| Download Long-term OC and Planned Outage Periods | DMSWS\_POP\_OUT | Allows user to download long-term OC and planned outage periods for a long-term auction.  Input parameters:  Auction ID | Trader Border Administrator  TSO |

## Data Flows for Sending Data

Delivery of the values to Damas is automatically confirmed and sender is immediately informed about processing result. Processing results are delivered as single response message which includes acknowledgement to request and results from processing of the request.

### Submit/Modify Auction Bids

It enables submitting and modifying bids for Long-term and Daily auctions into Damas.

#### Description

This data flow enables uploading the XML Bid Document with Long-term and Daily auctions. The resolution P1D is used for long-term auctions. For Daily auctions is used PT60M. The XML file contains one or more bid time series with appropriate number of the values for every auction.

Long-term, resp. Daily, bids can be submitted/edited/deleted until the to long-term and Daily auction before gate closure for bids submitting on given Border and given Auction.

Bids registered in the system can be deleted (including the default bids and bids entered via system screen) by uploading an XML file with a new/updated set of bids omitting the bid desired to be deleted.

*NOTE:* For *all information about cancellation of bids, please see Long-term, resp. Daily, Auction Bids chapter.*

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_BID\_IN |  |
| XML | XmlParam | XML Bid Document with capacity timeseries. | See [Bid Document](#_Bid_Document) definition. It is possible to send long-term, daily bids for one auction. |

#### Output Parameters

User receives acknowledgement message as response confirming data delivery and describing processing results.

#### Business Validations

Following business validations are performed when uploading auction bids:

1. Logged user must be authorized to send data (i.e. sending user must have operator or manager role or in case of the Trader, user’s party must correspond to the SenderIdentification Element)
2. In case of the updating existing bids, new version of the bid document must contain all bids from previous version (bids are identified by Bid ID)
3. DocumentType must be A24 (Bids)
4. Document must contain bids only to one auction
5. Document must contain bids only for one participant
6. Provided document version must be higher than the last accepted document’s version
7. Provided time interval must be in correct format (Delivery Days to be considered)
8. Document must contain at least one timeseries
9. Only one period is allowed in time series
10. Auction participant must exist in system
11. Auction participant must be allowed to send bids to specified auction
12. In and Out Area elements must specify existing Border direction
13. Bids to only one Border direction are allowed in document
14. Specified auction must exist
15. Auction must be operated by Damas
16. Auction must be held for given Border direction
17. Auction must be in state for submitting bids
18. There must be at least one non-zero capacity specified for new diagrams
19. Maximum number of the bids for auction participant must not be exceeded
20. Measurement unit quantity element’s value must be MAW
21. Measurement unit price element’s value must be MWH
22. Currency element’s value must be EUR (configurable)
23. Number of the days/hours must correspond with time interval specified
24. Resolution must be PT60M (for daily bids) or P1D for long-term bids
25. Bids must be specified as divisible (i.e. value of the Divisible element must be A01)
26. The bids must not be specified as block (i.e. value of the BlockBid element must be A02)
27. Time series identification must be unique in document
28. Auction participant’s credit limit must not be exceeded. In case the configuration is set to “Warning”. Bids are not rejected, but accepted with A10 reason code (Credit limit exceeded).
29. Bid amount must be non-negative integer, higher than 1MW
30. Bid price must be non-negative decimal number with 2 decimal positions and is different for each bid
31. Zero bid price must be used when bid amount is zero
32. Non-zero bid amount must be greater than auction’s minimum bid amount allowed
33. Bid amount must be lower than auction’s maximum bid amount allowed
34. Bid amount capacity must not exceed auction’s Offered Capacity

**Validations performed only for Daily bids:**

1. Bids are allowed only for one Delivery Day

**Validations performed only for Long-term bids:**

1. Bids price and capacity must be the same for all Delivery Days

### Submit/Modify Capacity Transfers

It enables submitting/modifying the Capacity Transfers for the Long-term and Daily auction into Damas.

#### Description

This data flow enables uploading of the XML Rights Document with Long-term or Daily capacity Transfers for one or more Delivery Days and Border direction to Damas. The resolution PT60M is used. The XML file contains one or more capacity timeseries with 24 (23, 25) values for every Delivery Day, and Border direction. In case there is clock change day inside the Transfer Time Interval, interval must be split into multiple documents to define clock change day independently.

Capacity Transfers can be submitted relative to the Long-term and Daily auction for Delivery Day before gate closure on respective Border.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_TRA\_IN |  |
| XML | XmlParam | XML Rights Document with capacity (i.e. capacity to be transferred) timeseries. | See Rights Document definition. It is possible to send capacity transfers for one or more Delivery Days, one Border direction and one auction. |

#### Output Parameters

User receives acknowledgement message as response confirming data delivery and describing processing results.

#### Business Validations

Following business validations are performed when uploading Transfers:

1. Document Type element must have the value A21 (transfers)
2. Provided document version must be higher than the last accepted document’s version
3. Only one timeseries is allowed for whole document. For the Long-Term Transfers time interval may be longer than one Delivery Day.
4. If document identification already exists in system, it must contain the same time series as provided (i.e. the existing transfer update)
5. The Rights Holder and transferee must exist in system
6. In and Out Area elements must specify existing Border direction
7. Business Type must be A32 – capacity transfer notification
8. For daily transfers time interval must specify only one Delivery Day
9. Time interval must be in correct format
10. Number of the values must correspond with time interval specified
11. User must be allowed to send data (i.e. having correct system role or correspond to the Sender Identification element)
12. Transfer must be performed between two different subjects (i.e. the Rights Holder must be different from the Transferee)
13. System must be in state for entering transfers
14. The Rights Holder must be an auction participant
15. At least one of the values must be non-zero
16. Capacity curtailment must not be in progress
17. Transfer with the same time series identification (transfer ID) must not exist for the same Rights Holder unless existing transfer ID belongs to the same document (i.e. transfer updating)
18. Existing Transfers in unconfirmed state may be updated.
19. First part of the Contract Identification element’s value must correspond to the Rights Holder (the second part specifies the source auction)
20. Specified auction must exist in system
21. Transferred capacity must be non-negative integer
22. The Rights Holder must have enough capacity to be transferred

### Submit/Modify Capacity Resales

It enables submitting/modifying explicit Capacity Resales of the Long-term capacity to subsequent Long-term auction into Damas.

#### Description

This data flow allows sending of the XML Rights Document with explicit Capacity Resales of the Long-term capacity to subsequent Long-term auction to Damas. The resolution P1D is used. The XML file contains one capacity timeseries with appropriate number of the values for given auction.

Capacity Resales can be submitted relative to the Long-term auction before gate closure on given Border.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_RES\_IN |  |
| XML | XmlParam | XML Rights Document with capacity (i.e. capacity to be resold) timeseries. | See Rights Document definition. It is possible to send explicit Capacity Resale for one Border direction and one given source and target auction. |

#### Output Parameters

User receives acknowledgement message as response confirming data delivery and describing processing results.

#### Business Validations

Following business validations are performed when uploading Resales:

1. Document Type element must have the value A19 (resales)
2. Provided document version must be higher than the last accepted document’s version
3. Only one timeserie is allowed for whole document
4. If document identification already exists in system, it must contain the same time series as provided (i.e. existing transfer update)
5. Time intervals in document must be in correct format
6. Document’s time interval must correspond with time series’ time interval
7. User must be allowed to send data (i.e. having correct system role or correspond to the Sender Identification element)
8. Business Type must be A32 – capacity transfer notification
9. The Rights Holder must exist in system
10. Out and In area must exist in system and must specify existing Border direction
11. Specified Capacity Contract Type must exist in system
12. Target auction (element Auction Identification) must exist
13. Source auction (its ID is parsed from the Contract Identification element) must exist
14. Subject EIC code parsed from the Contract Identification element must correspond with the Rights Holder
15. Specified contract type must correspond with the source auction type
16. Specified resolution must be P1D (one day)
17. Number of the values must correspond with time interval
18. All values must be the same for the whole time interval
19. Resale with the same time series identification (resale ID) must not exist for the same Rights Holder unless existing resale ID belongs to the same document (i.e. resale updating)
20. Update of the resale is allowed only for unconfirmed resale (i.e. not cancelled or confirmed)
21. Source auction must have its results already published
22. Trader may submit resale until target auction is in state Capacity Resale Opened. Border Administrator may do it before the Long-term Auction and OC has been published.
23. Target auction interval must be contained in source auction interval
24. Resales entering must be allowed for source auction in the Auction Configuration Register
25. Capacity curtailment must not be in progress
26. Source and target auction products must be compatible
27. All values must be non-negative integers
28. The Rights Holder must have enough capacity to be resold

## Data Flows for Data Download

### Download Offered Capacity

It enables downloading offered capacity values for Daily auctions from Damas.

#### Description

This data flow enables downloading of the XML Capacity Document containing hourly offered capacity values. The XML file contains offered capacity for selected auction. The resolution PT60M is used for Daily auctions. The XML file contains one schedule with appropriate number of the values for given auction.

The web service is accessible to Border Administrators and Traders. These users are entitled to download the same data.

It is possible to download data after publishing of the offered capacity for given auction.

Offered capacity of the Long-term auctions is accessible using the [Download Long-term Offered Capacity and Planned Outage Periods](#POP).

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_ATC\_OUT | Mandatory parameter |
| AuctionID | StringParam | Auction identification of the individual auction. | Mandatory parameter |

Example of the input parameters:

<Input>

<FID>**DMSWS\_ATC\_OUT**</FID>

<Parameters>

<StringParam Name="AuctionID">**MEBA-DH-01032011-00017**</StringParam>  
 </Parameters>

</Input>

#### Input Validations

Following validations are being performed for input parameters:

1. Parameter Auction ID must identify Daily auction registered in Damas.
2. Offered capacity for given auction has to be published in system.

#### Output Parameters

Offered capacity is received in the XML [Capacity Document](#_Result_Message) (see Appendix B).

### Download Capacity Rights

It enables downloading Capacity Rights for the Delivery Day, the Border directions and relevant capacity types (long-term, daily) from Damas.

#### Description

It enables downloading the XML Rights Document with hourly values of the Long-term and daily capacity rights for the given Delivery Day, Border direction and the Market Participant. The resolution PT60M is used. The XML file can contain multiple time series of 24 (23, 25) values for each Market Participant.

The web service is accessible to Trader users. Border Administrator is allowed to download Capacity Rights for all Market Participants. The Trader is entitled to download only own Capacity Rights.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_ENT\_OUT | Mandatory parameter |
| Date | DateParam | Delivery Day in format YYYY-MM-DD | Mandatory parameter |
| OutArea | StringParam | First Control Area defining the in combination with InArea. | Mandatory parameter  See List of the Control Areas. |
| InArea | StringParam | Second Control Area defining the Border in combination with OutArea. | Mandatory parameter  See List of the Control Areas. |
| Nominator | StringParam | EIC of the Market Participant. | Mandatory parameter  Capacity Rights of all Market Participant will be downloaded when the Nominator (ITR) parameter is empty (this feature is available only for Border Administrator). |
| Trader | StringParam | EIC of the Market Participant (the same code as in Nominator field). This concept is due to compatibility of the data flow with other Damas installations. | Mandatory parameter  Rights of all Traders will be downloaded when the Subject parameter is empty (this feature is available only for Border Administrator). |

*Note: Trader and Nominator are entered due to backward compatibility with other Damas installations.*

Example of the input parameters:

<Input>

<FID>**DMSWS\_ENT\_OUT**</FID>

<Parameters>

<DateParam Name="**Date**">**2011-08-01**</DateParam>

<StringParam Name="**OutArea**">**10YCS-CG-TSO---S**StringParam>  
 <StringParam Name="**InArea**">**10YBA-JPCC-----D**</StringParam>

<StringParam Name="**Nominator**">**10X--TRADER02---**</StringParam>

<StringParam Name="**Trader**">**10X--TRADER02---**</StringParam>

</Parameters>

</Input>

#### Input Validations

The following validations are being performed for input parameters:

1. The parameter Date must be in format YYYY-MM-DD
2. OutArea must contain valid EIC code from list of the Control Areas
3. InArea must contain valid EIC code from list of the Control Areas
4. Combination of the OutArea and InArea must be valid Damas Border Direction
5. Nominator must contain EIC code of the Market Participant registered in Damas
6. Trader must contain EIC code of the Market Participant registered in Damas

#### Output Parameters

Rights are received in the XML Rights Document (see Appendix B).

### Download Detailed Auction Results

It enables downloading results of the individual daily and Long-term auction per Trader from Damas.

#### Description

Download the XML Allocation Result Document with hourly values of the detailed auction results from given auction. The resolution P1D is used for Long-term auctions. For Daily auctions is used PT60M. The XML file can contain multiple timeseries of the appropriate number of the values for given auction for each Market Participant. Planned outage periods and Long-term auction products are not reflected for Long-term auctions.

The web service is accessible to the Border Administrator and Trader. The Trader is entitled to download auction results from his/her participation in auction. If the Trader has not participated in auction, the resulting XML file will contain no timeseries.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_DAR\_OUT | Mandatory parameter |
| AuctionID | StringParam | AuctionID identification of the individual auction. | Mandatory parameter |
| Trader | StringParam | EIC of the Trader. | Mandatory parameter |

Example of the input parameters:

<Input>

<FID>**DMSWS\_DAR\_OUT**</FID>

<Parameters>

<StringParam Name="**AuctionID**">**MEBA-A-01012011-00001**</StringParam>  
 <StringParam Name="**Trader**">**10X--TRADER01---**</StringParam>

</Parameters>

</Input>

#### Input Validations

Following validations are being performed for input parameters:

1. Parameter AuctionID must contain auction registered in Damas
2. Trader must contain EIC code of the Trader registered in Damas

#### Output Parameters

Detailed auction results are received in the XML Allocation Result Document (see Appendix B).

### Download Current Date and Time

Downloading the current date and time from Damas

#### Description

Downloading current date and time from Damas; this web service is accessible to all users.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | GETDATETIME | Mandatory parameter |

Example of the input parameters:

<Input>

<FID>**GETDATETIME**</FID>

<Parameters/>

</Input>

#### Output Parameters

Current date and time in Damas is received. Time is in form of the Coordinated Universal Time (UTC).

### Download Long-term Offered Capacity and Planned Outage Periods

#### Description

Download information on the Offered Capacity and defined Planned Outage Periods (POP) for given Long-term auction.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_POP\_OUT | Mandatory parameter |
| AuctionID | StringParam | Unique identification of the required Long-term auction.  example: MEBA-M-01032011-00017 | Mandatory parameter |

Example of the input parameters:

<Input>

<FID>**DMSWS\_POP\_OUT**</FID>

<Parameters>

<StringParam Name="AuctionID">**MEBA-M-01032011-00017**</StringParam>  
 </Parameters>

</Input>

#### Input Validations

Following validations are being performed for input parameters:

1. Parameter AuctionID must contain Auction registered in Damas

#### Output Parameters

The POP is delivered in the XML POP Information (see Appendix B)

### Download Auction Specification (CASD)

#### Description

Downloading Auction Specification document (CASD) according to given parameters. It returns data for the auctions of given ContractType that are whole in given period.

#### Input Parameters

List of the input parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | Note |
| FID |  | DMSWS\_CASD\_OUT | Mandatory parameter |
| OutArea | StringParam | First Control Area defining the in combination with InArea. | Mandatory parameter  See List of the Control Areas. |
| InArea | StringParam | Second Control Area defining the Border in combination with OutArea. | Mandatory parameter  See List of the Control Areas. |
| ContractType | StringParam | The parameter defines the type of requested Auctions. | Mandatory parameter  A01 - Daily Auctions  A03 - Monthly Auctions  A04 - Yearly Auctions |
| DateFrom | DateParam | Start Day of the period in format YYYY-MM-DD | Mandatory parameter |
| DateTo | DateParam | End Day of the period in format YYYY-MM-DD | Mandatory parameter |

Example of the input parameters:

<Input>

<FID> **DMSWS\_CASD\_OUT** </FID>

<Parameters>

<DateParam Name="DateFrom">2017-05-01</DateParam>

<DateParam Name="DateTo">2017-05-31</DateParam>

<StringParam Name="InArea">10YAL-KESH-----5</StringParam>

<StringParam Name="OutArea">10YBA-JPCC-----D</StringParam>

<StringParam Name="ContractType">A01</StringParam>  
 </Parameters>

</Input>

#### Input Validations

Following validations are being performed for input parameters:

1. OutArea must contain valid EIC code from list of the Control Areas
2. InArea must contain valid EIC code from list of the Control Areas
3. Combination of the OutArea and InArea must be valid Damas Border Direction
4. The parameter DateFrom must be in format YYYY-MM-DD
5. The parameter DateTo must be in format YYYY-MM-DD
6. ContractType must be valid capacity type.

#### Output Parameters

The POP is delivered in the XML POP Information (see Appendix B)

# XSD Schemas

This part of the document provides detailed technical description of all XSD schemas used in Damas. Each XSD description contains model of the XSD schema structure, detailed description of the schema and explanation of the meaning of all XSD elements. Examples of the XML files are attached to the document.

## List of the XSD Schemas

### ENTSO-E XSD Schemas

|  |  |  |
| --- | --- | --- |
| **Web Service ID** | **Description** | **Name of XSD Schema** |
| DMSWS\_ATC\_OUT | Capacity Document is used for downloading offered capacity values. | Capacity Document |
| DMSWS\_BIDS\_IN | Bid Document is used for sending bids into auctions. | Bid Document |
| DMSWS\_DAR\_OUT | Allocation Result Document is used for downloading Auction Results. | Allocation Result Document |
| DMSWS\_ENT\_OUT | Rights Document is used for downloading Capacity Rights. | Rights Document |
| DMSWS\_TRA\_IN | Rights Document is used for uploading Capacity Transfers. | Rights Document |
| DMSWS\_RES\_IN | Rights Document is used for uploading Capacity Resales. | Rights Document |
| Result of Input WS | Acknowledgement document is used for acknowledging receptions of the bids, transfers, resales and nominations. | Acknowledgement Document |

### Damas specific XSD schemas

|  |  |  |
| --- | --- | --- |
| **Web Service ID** | **Description** | **Name of XSD Schema** |
| DMSWS\_POP\_OUT | Long-Term Information Document is used for downloading Long-term OC and Outage Periods. | Long-Term Information Document |
| DMSWS\_INV\_OUT | Invoicing Document is used for downloading Data. | Invoicing Document |
| DMSWS\_CASD\_OUT | CASD WS Document is used for downloading Data. | CASD WS Document |

## Description of ENTSO-E XSD Schemas

All XSD Schemas described below are based on the ENTSO-E standards.

### Capacity Document

#### Capacity Document Description

The Capacity Document is used for transferring offered capacity values.

The structure of the XML file is divided into two parts. Header element of the Capacity Document contains primary information about message, such as document identification, document type and current time, and it is used for storage of the EIC codes of sender and recipient of document. The OutArea and InArea elements are used for definition of the Border direction. The Period element defines time period where offered capacity is valid. Period type is distinguished by element Period Type as whole period of the auction. Offered Capacity values are stored in element Qty.

Period element defines the Delivery Day or period for which values are entered and its time resolution. The XML file with resolution PT60M contains schedules with 24 values (23 when switching from winter time to summer time, 25 when switching from summer time to winter time). In case of 25, addition hour is inserted in right position inside of the diagram and remaining hourly values are shifted up.

#### Specification of Capacity Document Elements

List of the XML elements included in the Capacity Document element

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Document Identification | Unique identification of the document for which time series data is being supplied. | The naming convention is: <DocumentType>\_ <AuctionID> | Mandatory |
| Document Version | Version of the document being sent. | Version number is always 1. | Mandatory |
| Document Type | Coded type of the document being sent. | A13 – used for offered capacity.  A32 – used for proposed capacity. | Mandatory |
| Process Type | Nature of the process that document is directed at. | A07 (Capacity Allocation) – used for offered capacity.  A15 (Capacity Determination) | Mandatory |
| Sender Identification | Identification of the party who is sending document. | EIC code of the sender.  A01 coding scheme. | Mandatory |
| Sender Role | Identification of the role that is played by sender. | A18 (Grid operator).  A04 (System operator) used for data upload.  A07 (SEE CAO) | Mandatory |
| Receiver Identification | Identification of the party who is receiving document. | EIC code of the receiver. A01 coding scheme. | Mandatory |
| Receiver Role | Identification of the role played by receiver. | A29 (Trader) – Used for downloading offered capacity.  A04 (System Operator)  A07 (SEE CAO) | Mandatory |
| Creation Date And Time | Date and time of the creation of the document. UTC coding. Format:  YYYY-MM-DDTHH:MM:SSZ |  | Mandatory |
| Capacity Time Interval | Beginning and ending date and time of the period covered by document.  UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | For Daily auction Interval contains one Delivery Day. | Mandatory |
| Domain | Identification of the domain that is covered in the Capacity Document. Domain identifies the ICO concerned in time series. | EIC code of the Domain. A01 coding scheme. | Mandatory |
| Capacity Time Series | Time series containing schedule. |  | Optional |

List of the XML elements included in the Capacity Time Series element:

|  |  |  |  |
| --- | --- | --- | --- |
| **Element** | **Description** | **Values** | **Applicability** |
| Time Series Identification | Sender’s identification of the time series instance that uniquely identifies the Capacity time series. | String. | Mandatory |
| Business Type | Identifies the nature of the time series for which product is handled. | A31 (Offered Capacity)  A26 (Available Transfer Capacity) | Mandatory |
| Product | Identification of the energy product. | 8716867000016 (Active power). | Mandatory |
| In Area | Identification of the destination area of the Border direction. | EIC code of destination control area. A01 coding scheme. | Mandatory |
| Out Area | Identification of the source area of the Border direction. | EIC code of source control area. A01 coding scheme. | Mandatory |
| Measurement Unit | Unit of the measure which is applied to quantities in which time series is expressed. | MAW | Mandatory |
| Auction Identification | Unique identification of the set of the specifications that clearly identify auction to which capacity is addressed. | String. | Mandatory |
| Period |  |  |  |

List of the XML elements included in the Period element:

|  |  |  |  |
| --- | --- | --- | --- |
| **Element** | **Description** | **Values** | **Applicability** |
| Time Interval | Start and end date and time of the time interval of the period. UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | The same rules as for Schedule Time Interval. | Mandatory |
| Resolution | Resolution defining the number of periods that the time interval is divided. | PT60M – for Daily auctions | Mandatory |
| Interval | Time period interval. |  | Mandatory |

List of the XML elements included in the Interval element:

|  |  |  |  |
| --- | --- | --- | --- |
| **Element** | **Description** | **Values** | **Applicability** |
| Pos | Relative position of the period within time interval. | Non-signed integer value. | Mandatory |
| Qty | Quantity of the product scheduled for position within time interval. | Non-signed integer value. | Mandatory |

**Capacity Document Example**

Example of the Capacity Document with offered capacity for daily auction (downloaded by auction participant for Delivery Day 16.9.2010) and border direction BA-ME:

<CapacityDocument xmlns="http://auctions.seecao.com/xsd/CapacityDocument.xsd" DtdVersion="3" DtdRelease="1">

<DocumentIdentification v="A13\_BAME-DH-16092010-00001"/>

<DocumentVersion v="1"/>

<DocumentType v="A13"/>

<ProcessType v="A07"/>

<SenderIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<SenderRole v="A07"/>

<ReceiverIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<ReceiverRole v="A29"/>

<CreationDateTime v="2010-09-16T10:59:47Z"/>

<CapacityTimeInterval v="2010-09-16T05:00Z/2010-09-17T05:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<CapacityTimeSeries>

<TimeSeriesIdentification v="1"/>

<BusinessType v="A26"/>

<Product v="8716867000016"/>

<InArea v=" 10YCS-CG-TSO---S" codingScheme="A01"/>

<OutArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<MeasureUnit v="MAW"/>

<AuctionIdentification v="BAME-DH-16092010-00001"/>

<Period>

<TimeInterval v="2010-09-16T05:00Z/2010-09-17T05:00Z"/>

<Resolution v="PT60M"/>

<Interval>

<Pos v="1"/>

<Qty v="60"/>

</Interval>

...

<Interval>

<Pos v="24"/>

<Qty v="60"/>

</Interval>

</Period>

</CapacityTimeSeries>

</CapacityDocument>

### Bid Document

#### Bid Document Description

The Bid Document is used for submitting bids to Long-term (annual, monthly) and daily auctions. Data format is compliant with ECAN v4r0 and CodeList v6r4.

**Bid Document**

The Bid Document contains set of the all bids for one Trader and one auction. Bid is represented by time series. Primary information on the Bid Document, such as identification and version, sender's and recipient's identification and creation time, are included in header of the message.

Each Trader may send only one such document for auction and modifies bids by uploading new versions of this document. This document must always contain all already submitted bids in previous version. For cancellation of the bid/bids, it is necessary to send given bid/bids with zero values of the requested capacity.

If capacity is not requested in some hour, requested capacity and offered price will be equal to zero in this hour.

**Bid Time Series**

Each auction bid is stored in the BidTimeSeries element and is described by its unique identification, source and target area, auction identification and auction participant for which capacity is requested. The Bid Identification is unique identification of the bid for given auction identification. All bids are divisible. Linked bids and block bids are not used.

Auction period and resolution are specified in the Period element. There could be only one period class for time series. Time interval covered with period class must be equal to bid time interval. For submitting bids to auction, system elements Qty and Price must be populated with bid amount and bid price.

Daily resolution is used for Long-term auction. Hourly resolution is used for Daily auction; i.e. XML file with resolution PT60M containing 24 values (23 when switching from winter time to summer time, 25 when switching from summer time to winter time) is used for Daily auction.

**Rules for Versions**

Version is defined for Auction Participant and auction.

Rules for increasing version for Long-term auction bids:

* If set of the bids for Auction Participant is entered via *Auction Bid* web form and all business validations *Check of Long-term Auction Bid* are successful, version is increased for set of the bids for given auction and Auction Participant.
* If set of the bids for Auction Participant is uploaded as the XML Bid Document via *Long-term Bid Upload (XML)* web form or web services and all business validations *Check of Long-term Auction Bid* and validations of the XML Bid Document are successful, version of the document is stored as new version of the bid set for given auction and Auction Participant.

Rules for increasing version for Daily auction bids:

* If one bid is entered via web form *Daily Auction Bids* and all business validations *Check of Daily Auction Bid* are successful, version is increased for set of the bids for given auction and Auction Participant.
* If set of the bids for Auction Participant is uploaded as the XML Bid Document via *Daily Bids Upload (XML)* web form or web services and all business validations *Check of Daily auction Bid* and validations of the XML Bid Document are successful, version of the document is stored as new version of the bid set for given auction and Auction Participant.

#### Bid Elements Description

List of the XML elements included in the Bid Document element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Document Identification | Unique identification of the document for which the time series data is being supplied.  Document name must be the same for all document transmission. | String with 35 characters.  The recommended naming convention is: <DocumentType>\_<PartyID>\_<AuctionID>  <DocumentType> - A24.  <PartyID> - EIC of auction participants.  <AuctionID> - The naming convention of auction is described in *Long-term Auction Module*. Only last part with auction ID will be used (whole auction identification is too long).  If first bid is created via web form (before sending first XML document), document identification will be automatically created by system. Trader must use this document identification in uploaded XML files. Document identification will be displayed in message attachment with processing results. | Mandatory |
| Document Version | Version of the document being sent. Each transmission being identified by different version number that starts at 1 and increases sequentially. | Non-signed integer value. If at least one bid is modified or added, document version will be increased. The initial value is 1.  Version of the provided XML Bid Document for given Trader and auction must be higher than previous accepted version. Otherwise, document will be rejected. Document version will be displayed in message attachment with processing results. | Mandatory |
| Document Type | Coded type of the document being sent. | A24 – Bid document. | Mandatory |
| Sender Identification | Identification of the party who is sending document. | A01 coding scheme. EIC code of market participant. | Mandatory |
| Sender Role | Identification of the role that is played by sender. | A29 (Trader) – For submitting bids. | Mandatory |
| Receiver Identification | Identification of the party who is receiving document. | EIC code of the receiver. A01 coding scheme. | Mandatory |
| Receiver Role | Identification of the role played by receiver. | A18 (Transmission capacity allocator) – For submitting bids. | Mandatory |
| Creation Date Time | Date and time of the creation of the document. UTC coding. Format:  YYYY-MM-DDTHH:MM:SSZ | Actual date and time of the creation and providing of the document.  Date and time is updated with each document transmission. | Mandatory |
| Bid Time Interval | Beginning and ending date and time of the period covered by document. UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | For Long-term auction the BidTimeInterval is equal to auction period. For Daily auction the BidTimeInterval contains one Delivery Day. | Mandatory |
| Domain | Identification of the domain that is covered in the Capacity Document. | EIC code of Domain. A01 coding scheme | Mandatory |
| Subject Party | Party is Trader for whom bids are being submitted. | EIC code of the Trader. A01 coding scheme. | Mandatory |
| Subject Role | Role of the Subject Party. | A29 (Trader) – For submitting bids. | Mandatory |
| Bid Time Series | Time series containing schedule. |  | Optional |

List of the XML elements included in the Bid Time Series element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Bid Identification | Identification attributed by sender that uniquely identifies bid. This must be unique for over time and guarantee the non-duplication of the bid for sender in future bids. | String. Non-signed integer value recommended. Identification must be unique inside of the whole document.  If bid is created via web form, bid identification is generated and assigned by system and could not be changed. Market participant must use this identification.  If bid is uploaded in XML document, identification for new bids is assigned by bidder and stored in system. If value -1 is entered, bid identification will be assigned by system in the same way as via web form.  Set of the bids in the XML Bid Document must always correspond with bids already stored in Damas, only new bids could be added. Key parameters of the bids already stored in Damas (i.e. bid identification, auction participant, Border direction, auction ID) must remain unchanged in new version of the XML Bid Document. Only values of the requested capacity and offered price could be modified. | Mandatory |
| Auction Identification | Unique identification of the set of the specifications that clearly identify auction to which capacity is addressed. | The naming convention of the auction is described in *Long-term Auction Module*. | Mandatory |
| Business Type | Identifies the nature of the time series for which product is handled. | A03 (External trade Explicit Capacity). | Mandatory |
| In Area | Identification of the destination area of the Border direction. | EIC code of the destination area. A01 coding scheme. | Mandatory |
| Out Area | Identification of the source area of the Border direction. | EIC code of the source area. A01 coding scheme. | Mandatory |
| Measure Unit Quantity | Unit of the measure in which quantities in time series are expressed. | MAW | Mandatory |
| Currency | Currency in which monetary amount is expressed. | See Currency Code List. | Mandatory |
| Measure Unit Price | Unit of the measure in which price in time series is expressed per unit of the currency. | MWH (MWh per unit) is used for all auction types. | Mandatory |
| Divisible | The Indication whether or not each element of the bid may be partially accepted or not. | A01 (Yes) or A02 (No) | Mandatory |
| Linked Bids Identification | Unique identification associated with all linked bids. | Not used. Element should not be included in the Bid Document XML. | Dependent |
| Block Bid | Indication that values in period constitute block bid and that they cannot be changed. | A02 (No) – Used for all bids in daily and long-term auctions. Block bids are not supported. | Mandatory |
| Period |  | Only one period class will be used for one time series. |  |

List of the XML elements included in the Period element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Time Interval | Start and end date and time of the time interval of the period. UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | The time interval must be equal to auction period, i.e. it contains the same value as the BidTimeInterval element. | Mandatory |
| Resolution | Resolution defining number of the periods that time interval is divided. | PT60M – Hourly resolution will be supported for Daily auction only.  PT1D – Daily resolution will be supported for long-term auction only.  The resolution must be constant through whole time series. | Mandatory |
| Interval | Time period interval. |  | Mandatory |

List of the XML elements included in the Interval element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Pos | Relative position of the period within the bid interval. | Non-signed integer value starting from 1. | Mandatory |
| Qty | Quantity that is bid for interval in question. | Non-signed integer value. | Mandatory |
| Price Amount | Price expressed for each unit of the quantity. | Non-signed decimal value with two digits. The decimal mark that separates digits shall always be period (“.”). | Mandatory |

#### Bid Examples

The Bid Document XSD schema

<?xml version="1.0" encoding="UTF-8"?>

<BidDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/bid-document.xsd" DtdVersion="4" DtdRelease="0">

#### Long-term Auction Bids

The same XML structure will be used for submitting bids for annual and monthly auctions.

Example is the Bid Document for monthly auction for January 2010 (Trading Month) and border direction ME-BA:

<?xml version="1.0" encoding="UTF-8"?>

<BidDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/bid-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A24\_10X--TRADER01---\_00556"/>

<DocumentVersion v="1"/>

<DocumentType v="A24"/>

<SenderIdentification v="10X--TRADER01---" codingScheme="A01"/>

<SenderRole v="A29"/>

<ReceiverIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<ReceiverRole v="A07"/>

<CreationDateTime v="2009-12-01T11:02:28Z"/>

<BidTimeInterval v="2010-01-01T06:00Z/2011-02-01T06:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<SubjectParty v="10X--TRADER01---" codingScheme="A01"/>

<SubjectRole v="A29"/>

<BidTimeSeries>

<BidIdentification v="1099"/>

<AuctionIdentification v="MEBA-M-01012010-00556"/>

<BusinessType v="A03"/>

<InArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<OutArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<MeasureUnitQuantity v="MAW"/>

<Currency v="EUR"/>

<MeasureUnitPrice v="MWH"/>

<Divisible v="A01"/>

<BlockBid v="A02"/>

<Period>

<TimeInterval v="2010-01-01T06:00Z/2011-02-01T06:00Z"/>

<Resolution v="P1D"/>

<Interval>

<Pos v="1"/>

<Qty v="200"/>

<PriceAmount v="50.00"/>

</Interval>

…

<Interval>

<Pos v="31"/>

<Qty v="200"/>

<PriceAmount v="50.00"/>

</Interval>

</Period>

</BidTimeSeries>

</BidDocument>

#### Daily auction Bids

Example is the Bid Document for daily auction for 2.1.2010 and border direction ME-BA:

<?xml version="1.0" encoding="UTF-8"?>

<BidDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/bid-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A24\_10X--TRADER01---\_00666"/>

<DocumentVersion v="3"/>

<DocumentType v="A24"/>

<SenderIdentification v="10X--TRADER01---" codingScheme="A01"/>

<SenderRole v="A29"/>

<ReceiverIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<ReceiverRole v="A07"/>

<CreationDateTime v="2010-01-01T11:10:30Z"/>

<BidTimeInterval v="2011-01-02T06:00Z/2011-01-03T06:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<SubjectParty v="10X--TRADER01---" codingScheme="A01"/>

<SubjectRole v="A29"/>

<BidTimeSeries>

<BidIdentification v="2001"/>

<AuctionIdentification v="MEBA-DH-02012011-00666"/>

<BusinessType v="A03"/>

<InArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<OutArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<MeasureUnitQuantity v="MAW"/>

<Currency v="EUR"/>

<MeasureUnitPrice v="MWH"/>

<Divisible v="A01"/>

<BlockBid v="A02"/>

<Period>

<TimeInterval v="2011-01-02T06:00Z/2011-01-03T06:00Z"/>

<Resolution v="PT60M"/>

<Interval>

<Pos v="1"/>

<Qty v="10"/>

<PriceAmount v="2.00"/>

</Interval>

…

<Interval>

<Pos v="24"/>

<Qty v="10"/>

<PriceAmount v="2.00"/>

</Interval>

</Period>

</BidTimeSeries>

</BidDocument>

### Allocation Result Document

#### Document Description

The Allocation Result Document is used for receiving detailed auction results Long-term and Daily auctions. The results are available for each bid submitted to auction. The data format is compliant with ECAN v4r0 and CodeList v6r4. Only XSD schema is supported.

**Allocation Result Document**

Allocation Result Document contains a set of all bids for one Trader and one auction. Bid is represented by time series. All bids submitted to auction for this Trader will be always included in document. For each bid there will be available information about allocated capacity (it could be equal to zero) and auction price. Only values for whole auction period will be available without information about planned outage periods.

Primary information on allocation results document, such as identification, link to relevant bid document and its version, sender's and recipient's identification and creation time, are included in header of the message.

**Allocation Time Series**

Each auction bid is stored in the BidTimeSeries element and is described by its unique identification, source and target area, auction identification and auction participant for which capacity is requested / allocated. The same rules are valid for time series and period as in the Bid Document specification.

Daily resolution is used for Long-term auction. Hourly resolution is used for Daily auction; i.e. XML file with resolution PT60M containing 24 values (23 when switching from winter time to summer time, 25 when switching from summer time to winter time) is used for Daily auction.

The Reason element will not be used.

**Rules for Versions**

Version is defined for auction participant and auction. Version will be increased after publishing of the preliminary or final auction results of the given auction.

#### Allocation Result Elements Description

List of the XML elements included in the AllocationResultDocument element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Document Identification | Unique identification of the document for which time series data is being supplied.  Document name must be the same for all document transmission. | String with 35 characters.  The recommended naming convention is: <DocumentType>\_<PartyID>\_<AuctionID>  <DocumentType> - A25.  <PartyID> - EIC of the auction participants.  <AuctionID> - The naming convention of the auction is described in *Long-term Auction Module*. Only last part with auction ID will be used (whole auction identification is too long). | Mandatory |
| Document Version | Version of the document being sent. Each transmission being identified by different version number that starts at 1 and increases sequentially. | Non-signed integer value. If at least one bid is modified or added, document version will be increased. The initial value is 1. | Mandatory |
| Document Type | Coded type of the document being sent. | A25 – Allocation results document. | Mandatory |
| Sender Identification | Identification of the party who is sending document. | A01 coding scheme. EIC code of the sender. | Mandatory |
| Sender Role | Identification of the role that is played by the sender. | A07 (Transmission capacity allocator) – downloading auction results. | Mandatory |
| Receiver Identification | Identification of the party who is receiving document. | A01 coding scheme. EIC code of the market participant. | Mandatory |
| Receiver Role | Identification of the role played by receiver. | A29 (Trader) – For downloading auction results. | Mandatory |
| Creation Date Time | Date and time of the creation of the document. UTC coding. Format:  YYYY-MM-DDTHH:MM:SSZ | The date and time is updated with each document transmission. | Mandatory |
| Bid Time Interval | Beginning and ending date and time of the period covered by document. UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | For Long-term auction the BidTimeInterval is equal to auction period. For Daily auction the BidTimeInterval contains one Delivery Day. | Mandatory |
| Domain | Identification of the domain that is covered in the document. | EIC code of Domain.  A01 coding scheme. | Mandatory |
| Subject Party | Party is the Trader for whom the bids are being submitted. | EIC code of Trader. A01 coding scheme. | Mandatory |
| Subject Role | Role of the Subject Party. | A29 (Trader) | Mandatory |
| Allocation Time Series | Time series containing schedule. |  | Optional |

List of the XML elements included in the Allocation Time Series element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Time Series Identification | Identification of the times series instance.  This must be unique number that is assigned by auction office for each time series in document. | String. Non-signed integer value recommended. The identification must be unique inside of the whole document.  The value will be equal to the BidIdentification element in respective Bid Document XML. | Mandatory |
| Bid Document Identification | Unique identification of the document for which bids referenced are contained. | The value will be equal to the DocumentIdentification element in respective Bid Document XML. | Mandatory |
| Bid Document Version | Version of the bid document having been sent. | The value will be equal to the DocumentVersion element in respective Bid Document XML. | Mandatory |
| Bid Identification | Identification of the time series that was used in original bid. | The value will be equal to BidIdentification element in respective Bid Document XML. | Mandatory |
| Auction Identification | Unique identification of the set of specifications that clearly identify auction to which capacity is addressed. | The value will be equal to the AuctionIdentification element in respective Bid Document XML. | Mandatory |
| Business Type | Identifies the nature of the time series for which product is handled. | A03 (External trade Explicit Capacity). | Mandatory |
| In Area | Identification of the destination area of the Border direction. | EIC code of destination area. A01 coding scheme. | Mandatory |
| Out Area | Identification of the source area of the Border direction. | EIC code of source area. A01 coding scheme. | Mandatory |
| Contract Type | Contract type defines conditions under which capacity was allocated and handled. | See Capacity Contract Type Code List. | Mandatory |
| Contract Identification | Contract identification of the time series instance.  There is only one contract identification assigned by the Transmission Capacity Allocator per auction identification, bid period and subject party. | The recommended naming convention is: <PartyID>\_<AuctionID>  <PartyID> - EIC of auction participants.  <AuctionID> - The naming convention of the auction is described in *Long-term auction Module*. Only last part with auction ID will be used (whole auction identification is too long). | Mandatory |
| Measure Unit Quantity | Unit of the measure in which quantities in time series are expressed. | MAW | Mandatory |
| Currency | Currency in which monetary amount is expressed. | See Currency Code List. Example: EUR | Mandatory |
| Measure Unit Price | Unit of the measure in which price in time series is expressed per unit of the currency. | MWH (MWh per unit) is used for all auction types. | Mandatory |
| Period |  | Only one period class will be used for one time series. |  |

List of the XML elements included in the Period element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Time Interval | Start and end date and time of the time interval of the period. UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | The time interval must be equal to auction period, i.e. it contains the same value as the BidTimeInterval element. | Mandatory |
| Resolution | Resolution defining number of the periods that time interval is divided. | PT60M – Hourly resolution is used for Daily auction.  P1D – Daily resolution is used for Long-term auction.  The resolution must be constant through whole time series. | Mandatory |
| Interval | Time period interval. |  | Mandatory |

List of the XML elements included in the Interval element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Pos | Relative position of the period within the bid interval. | Non-signed integer value starting from 1. | Mandatory |
| Qty | Quantity that has been allocated in auction evaluation process. | Non-signed integer value. | Mandatory |
| Price Amount | Price that has been assigned in auction evaluation process.  Marginal price for marginal price algorithm. | Non-signed decimal value with two digits. The decimal mark that separates digits shall always be period (“.”). | Mandatory |
| BidQty | That quantity that was in the original bid document. | Non-signed integer value. | Mandatory |
| Bid Price Amount | Original price offered in the original bid document. | Non-signed decimal value with two digits. The decimal mark that separates digits shall always be period (“.”). | Mandatory |

#### Allocation Result Examples

Allocation Result Document XSD schema

<?xml version="1.0" encoding="UTF-8"?>

<AllocationResultDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/allocation-result-document.xsd" DtdVersion="4" DtdRelease="0">

#### Long-term Auction Results

Example of the Allocation Results Document with results for monthly auction for January 2010, border direction ME-BA and planned outage period defined for day 10.1.2010. Results published 5.12.2009:

<?xml version="1.0" encoding="UTF-8"?>

<AllocationResultDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/allocation-result-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A25\_10X--TRADER01---\_00556"/>

<DocumentVersion v="1"/>

<DocumentType v="A25"/>

<SenderIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<SenderRole v="A07"/>

<ReceiverIdentification v="10X--TRADER01---" codingScheme="A01"/>

<ReceiverRole v="A29"/>

<CreationDateTime v="2009-12-04T12:02:28Z"/>

<BidTimeInterval v="2010-01-01T06:00Z/2010-02-01T06:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<SubjectParty v="10X--TRADER01---" codingScheme="A01"/>

<SubjectRole v="A29"/>

<AllocationTimeSeries>

<TimeSeriesIdentification v="1099"/>

<BidDocumentIdentification v="A24\_10X--TRADER01---\_00556"/>

<BidDocumentVersion v="1"/>

<BidIdentification v="1099"/>

<AuctionIdentification v="MEBA-M-01012010-00556"/>

<BusinessType v="A03"/>

<InArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<OutArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<ContractType v="A03"/>

<ContractIdentification v="10X--TRADER01---\_00556"/>

<MeasureUnitQuantity v="MAW"/>

<Currency v="EUR"/>

<MeasureUnitPrice v="MWH"/>

<Period>

<TimeInterval v="2010-01-01T06:00Z/2010-02-01T06:00Z"/>

<Resolution v="P1D"/>

<Interval>

<Pos v="1"/>

<Qty v="0"/>

<PriceAmount v="65.00"/>

<BidQty v="200"/>

<BidPriceAmount v="50.00"/>

</Interval>

…

<Interval>

<Pos v="31"/>

<Qty v="0"/>

<PriceAmount v="65.00"/>

<BidQty v="200"/>

<BidPriceAmount v="50.00"/>

</Interval>

</Period>

</AllocationTimeSeries>

</AllocationResultDocument>

#### Daily auction Results

Example is the Allocation Result Document with daily auction results for 2.1.2010 and border direction ME-BA. Results published 1.1.2010:

<?xml version="1.0" encoding="UTF-8"?>

<AllocationResultDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/allocation-result-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A25\_10X--TRADER01---\_00666"/>

<DocumentVersion v="1"/>

<DocumentType v="A25"/>

<SenderIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<SenderRole v="A07"/>

<ReceiverIdentification v="10X--TRADER01---" codingScheme="A01"/>

<ReceiverRole v="A29"/>

<CreationDateTime v="2010-01-01T09:10:30Z“/>

<BidTimeInterval v="2010-01-02T06:00Z/2010-01-03T06:00Z"/>

<Domain v="10YCS-CG-TSO---S" codingScheme="A01"/>

<SubjectParty v="10X--TRADER01---" codingScheme="A01"/>

<SubjectRole v="A29"/>

<BidTimeSeries>

<TimeSeriesIdentification v="2001"/>

<BidDocumentIdentification v="A24\_10X--TRADER01---\_00666"/>

<BidDocumentVersion v="3"/>

<BidIdentification v="2001"/>

<AuctionIdentification v="MEBA-DH-02012010-00666"/>

<BusinessType v="A03"/>

<InArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<OutArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<ContractType v="A01"/>

<ContractIdentification v="10X--TRADER01---\_00666"/>

<MeasureUnitQuantity v="MAW"/>

<Currency v="EUR"/>

<MeasureUnitPrice v="MWH"/>

<Period>

<TimeInterval v="2010-01-02T06:00Z/2010-01-03T06:00Z"/>

<Resolution v="PT60M"/>

<Interval>

<Pos v="1"/>

<Qty v="10"/>

<PriceAmount v="0.00"/>

<BidQty v="10"/>

<BidPriceAmount v="2.00"/>

</Interval>

…

<Interval>

<Pos v="24"/>

<Qty v="10"/>

<PriceAmount v="0.00"/>

<BidQty v="10"/>

<BidPriceAmount v="2.00"/>

</Interval>

</Period>

</AllocationTimeSeries>

</AllocationResultDocument>

### Rights Document

#### Document Description

The Rights Document is used for transferring of the Long-term (multi-annual, annual, monthly, etc.) and daily capacities and for reselling Long-term capacity allocated on given Border. It also used for providing information about Long-term and daily capacity rights. Data format is compliant with ECAN v4r0 and CodeList v6r4.

**Rights Document**

One XML Rights Document used for downloading capacity rights for one Delivery Day, Border direction, Market Participant and capacity type (all capacity types included - Long-term capacity rights and daily capacity rights). Resolution is always PT60M.

One XML Right Document used for uploading capacity transfers or resale can contain only one time series with capacity transfer or resale. Primary information on the rights document, such as identification and version, sender's and recipient's identification and creation time are included in header of the message.

Each Trader may send only one such document for given transfer or resale and modifies transfers and resales by uploading new versions of this document. For cancellation of the transfer or resale, it is necessary to send given transfer or resale with zero values of the transferred or resold capacity.

If capacity is not transferred or resold in some hour, value will be equal to zero in this hour.

**Rights Timeseries**

There will be maximally one time-series for combination of the Delivery Day, Trader, Border direction and capacity type (Long-term, daily) for capacity rights. Capacity transfers, resales and possible curtailed capacity will be only included in sum with allocated capacity; these data will not be provided separately. Data will be provided in hourly resolution

Each capacity diagram is stored in the RightsTimeSeries element and is described by its unique identification, source and target control area and the Rights Holder for which capacity is allocated. Conditions under which capacity was allocated are specified in the element CapacityContractType. Document could include time series with zero allocated capacity in all hours.

Transfer is stored in the RightsTimeSeries element and is described by its unique identification, source and target area, auction identification, the Right Holder, transferee party and capacity type for which capacity is transferred. The Time Serie Identification is unique identification of the transfer for given Rights Holder.

Transfer period and resolution are specified in the Period element. There could be only one period class for time series. Time interval covered with period class must be equal to applicable time interval and cannot be longer than 31 days. For submitting transfers, system elements Qty must be populated with transfer amount. The resolution PT60M is used for capacity transfers.

Resale is stored in the RightsTimeSeries element and is described by its unique identification, source and target area, previous auction identification, auction identification, the Right Holder and capacity type for which the capacity is resold. The Time Serie Identification is unique identification of the resale for given Rights Holder.

Resale period and resolution are specified in the Period element. There could be only one period class for time series. Time interval covered with period class must be equal to applicable time interval. For submitting resales, system elements Qty must be populated with resold amount. The resolution P1D is used for capacity resales.

**Rules for Versions – Capacity Rights**

Market Participant will download Capacity Rights for the following parameters:

* Day – Delivery Day
* Border direction – one Border direction.
* Capacity Type - Long-term, Daily

Each combination of these parameters will have separate version that will be increased after each modification. Rules for increasing versions:

* If Long-term capacity rights will be re-evaluated, version will be increased for Long-term capacity rights.
* If Daily capacity rights will be re-evaluated, version will be increased for daily capacity rights.

Data exchange will be limited to one Delivery Day.

**Rules for Versions – Capacity Transfers**

* If capacity transfer is entered via web form *Capacity Transfer Form* and all business validations *Capacity Transfer Validation* are successful, version is increased for given transferor, Delivery Day(s) and capacity transfer.
* If capacity transfer for transferor is uploaded as the XML Rights Document via *Transfer and Resale Upload (XML)* web form or web services and all business validations *Capacity Transfer Validation* and validations of the XML Rights Document are successful, version of the document is stored as new version of the capacity transfer for given transferor, Delivery Day(s) and capacity transfer.

**Rules for Versions – Capacity Resales**

* If capacity resale is entered via web form *Capacity Resale Form* and all business validations *Capacity Resale Validation* are successful, version is increased for given reseller, source and target auction and capacity resale.
* If capacity resale for reseller is uploaded as the XML Rights Document via *Transfer and Resale Upload (XML)* web form or web services and all business validations *Capacity Resale Validation* and validations of the XML Rights Document are successful, version of the document is stored as new version of the capacity resale for given reseller, source and target auction.

#### Rights Document Elements Description

List of the XML elements included in the Right Document element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Document Identification | Unique identification of the document for which time series data is being supplied.  Document name must be the same for all document transmission. | String with at most 35 characters.  Document identification used for the capacity rights: <BusinessType>\_<ContractType>\_<BusinessDay>\_<ReceiverEIC>  Elements used in the Document Identifications of the various documents:  <BusinessType> - code of the business type.  <ContractType> - code of the capacity contract type.  <BusinessDay> - Delivery Day in format YYYYMMDD.  <Sender> - EIC code of message sender. The value of element Sender Identification.  <Receiver> - EIC code of the receiver of the XML document  <SenderTimeSerieIdentification> - unique identification (for capacity Trader) of submitted transfer or resale  If the first capacity diagram is created via web form (before sending first XML document), the document identification will be automatically created by system. The Trader must use this document identification in uploaded XML files. Document identification will be displayed in message attachment with processing results.  For Capacity Transfers and Resales data flows the document identification is to be specified by sending party. No special form is required by Damas. But the similar form as used for capacity rights is recommended. | Mandatory |
| Document Version | Version of the document being sent. Each transmission being identified by different version number that starts at 1 and increases sequentially. | Non-signed integer value. The initial value is 1.  Document version will be displayed in message attachment with processing results. | Mandatory |
| Document Type | Coded type of the document being sent. | A19 (Capacity for Resale) – Used for capacity resales.  A21 (Capacity transfer notification) – Used for capacity transfers.  A23 (Allocations) – Used for distribution of capacity rights | Mandatory |
| Sender Identification | Identification of the party who is sending the document. | A01 coding scheme. EIC code of Trader) – for submitting capacity transfers and resales. | Mandatory |
| Sender Role | Identification of the role that is played by sender. | A07 (Transmission capacity allocator) – For downloading capacity rights.  A29 (Trader) – for submitting capacity transfers and resales. | Mandatory |
| Receiver Identification | Identification of the party who is receiving document. | EIC code of the receiver. A01 coding scheme. | Mandatory |
| Receiver Role | Identification of the role played by receiver. | A07 (Transmission capacity allocator) – – for submitting capacity transfers and resales.  A29 (Trader) – For downloading capacity rights. | Mandatory |
| Creation Date Time | Date and time of the creation of the document. UTC coding. Format:  YYYY-MM-DDTHH:MM:SSZ | Actual date and time of the creation and providing of the document.  The date and time is updated with each document transmission.  The actual date and time will be used for data downloaded from system. | Mandatory |
| Applicable Time Interval | Beginning and ending date and time of the period covered by document.  UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | It contains always only one Delivery Day for capacity rights, interval of the maximally 31 days for capacity transfers and time interval of the target auction for capacity resales. | Mandatory |
| Domain | Identification of the domain that is covered in the Rights Document. | EIC code of Domain.  A01 coding scheme. | Mandatory |
| Document Status | Status for the rights document. | A02 – Final. Only final value will be used in this element. | Mandatory |
| Rights Time Series | Time series containing capacity diagram. |  | Optional |

List of the XML elements included in the Rights Time Series element:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Element | | Description | Values | Applicability |
| Time Series Identification | | Sender’s identification of the time series instance that uniquely identifies the capacity time series.  The identification must be unique inside of the whole document. | String. Non-signed integer value recommended. The indentification must be unique inside of the whole document. The identification must be also unique for given reseller or transferor.  If transfer or resale is created via web form, transfer (resale) identification is generated and assigned by system and could not be changed. Trader must use this time series identification.  If transfer or resale is uploaded in the XML document, the identification for new transfer or resale is assigned by transferor (reseller) and stored in system. If the entered value is “-1”, transfer (resale) identification will be assigned by system in the same way as via web form.  Transfer or resale in the XML Rights Document must always correspond with transfer (resale) already stored in Damas. The key parameters of the transfer (resale) already stored in Damas (i.e. transfer identification, capacity Trader, Border direction, auction ID) must remain unchanged in new version of the XML Rights Document. Only values of the transferred capacity could be modified.  The naming convention for capacity rights is: <OutArea><InArea>\_<ContractType>\_<PartyID>  <OutArea> - Country code of the source control area on Border direction.  <InArea> - Country code of the destination control area on Border direction.  <ContractType> - The value of the element Contract Type.  <PartyID> - EIC code of the Trader that is owner of the capacity rights. | Mandatory |
| Business Type | | Identifies the nature of the time series for which product is handled. | A32 (Capacity transfer notification) – used for capacity transfers and resales.  A33 (Authorised AAC) – used for capacity rights. | Mandatory |
| In Area | | Identification of the destination area of the Border direction. | EIC code of the destination area. A01 coding scheme. | Mandatory |
| Out Area | | Identification of the source area of the Border direction. | EIC code of source area. A01 coding scheme. | Mandatory |
| Rights Holder | Identification of the party who is owner of the transmission rights in question. | | The capacity owner.  EIC code of the Interconnection Trade Responsible. A01 coding scheme. | Dependent. Used for capacity rights and transfers. |
| Transferee Party | Identification of the party to whom the rights are being transferred or the Nominator designated by transferor to use rights. | | EIC code of the transferee of the allocated capacity. A01 coding scheme. | Dependent. Used for capacity transfers. |
| Contract Identification | | Contract identification of the time series instance. The identification that uniquely identifies the allocation. | It must be string with maximum 35 characters.  For capacity rights:  <PartyID>  For capacity transfers and resales:  <PartyID>\_<AuctionID>  <PartyID> - EIC code of Trader that is owner of capacity.  <AuctionID> - The identification of the auction where capacity was originally acquired. The naming convention of the auction is described in *Long-term Auction Module*. Only last part with auction ID will be used (whole auction identification is too long). | Mandatory |
| Contract Type | | Contract type defines conditions under which rights were allocated and handled. | See Capacity Contract Type Code List. | Mandatory |
| Previous Contract Identification | | Identification of the previous contract used to identify transfer rights. | Not used for capacity rights, resales and transfers. Not included in the RightsDocument XML. | Dependent |
| Measure Unit Quantity | | Unit of the measure that is applied to quantities in which time series is expressed. | MAW | Mandatory |
| Auction Identification | | Unique identification of the set of the specifications that clearly identify auction to which capacity is addressed. | Not used for capacity rights.  The identification of the auction where capacity was originally acquired for capacity Traders.  The identification of the target auction for capacity resales.  The naming convention of the auction is described in *Long-term Auction Module*. | Dependent |
| Period | |  | Only one period class will be used for one time series. |  |

List of the XML elements included in the Period element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Time Interval | Start and end date and time of the time interval of the period. UTC coding. Format:  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | The time interval must be equal to auction period, i.e. it contains the same value as ApplicableTimeInterval element. | Mandatory |
| Resolution | Resolution defining number of the periods that time interval is divided. | PT60M – Used for capacity rights and transfers.  P1D – Used for capacity resales. | Mandatory |
| Interval | Time period interval. |  | Mandatory |

List of the XML elements included in the Interval element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Pos | Relative position of the period within bid interval. | Non-signed integer value. | Mandatory |
| Qty | Quantity that has been assigned to nominations party (state in element Transferor) for interval in question. | Non-negative integer value. | Mandatory |

#### Rights Document Examples

The Rights Document XSD schema

<?xml version="1.0" encoding="UTF-8"?>

<RightsDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/rights-document.xsd" DtdVersion="4" DtdRelease="0">

#### Long-term Capacity Rights

Example of the Rights Document with Long-term capacity rights for Delivery Day 1.1.2010, Border direction ME-BA:

<?xml version="1.0" encoding="UTF-8"?>

<RightsDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/rights-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A23\_Z06\_20100101\_10X--TRADER03---"/>

<DocumentVersion v="1"/>

<DocumentType v="A23"/>

<SenderIdentification codingScheme="A01" v="10XCS-SEECAO---O"/>

<SenderRole v="A07"/>

<ReceiverIdentification codingScheme="A01" v="10X--TRADER03---"/>

<ReceiverRole v="A29"/>

<CreationDateTime v="2009-09-09T13:21:56Z"/>

<ApplicableTimeInterval v="2010-01-01T06:00Z/2010-01-02T06:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<DocumentStatus v="A02"/>

<RightsTimeSeries>

<TimeSeriesIdentification v="1"/>

<BusinessType v="A33"/>

<InArea v="10YBA-JPCC-----D" codingScheme="A01" />

<OutArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<RightsHolder v="10X--TRADER03---" codingScheme="A01"/>

<ContractIdentification v="10X--TRADER02---"/>

<ContractType v="Z06"/>

<MeasureUnitQuantity v="MAW"/>

<Period>

<TimeInterval v="2010-01-01T06:00Z/2010-01-02T06:00Z“/>

<Resolution v="PT60M"/>

<Interval>

<Pos v="1"/>

<Qty v="135"/>

</Interval>

…

<Interval>

<Pos v="24"/>

<Qty v="135"/>

</Interval>

</Period>

</RightsTimeSeries>

</RightsDocument>

#### Capacity Transfer

Example of the Rights Document with capacity transfers for Delivery Day 01.07.2010 and auction BAME-A-01012010-01355.

<?xml version="1.0" encoding="UTF-8"?>

<RightsDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/rights-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A21\_A04\_10X--TRADER03---\_1002"/>

<DocumentVersion v="1"/>

<DocumentType v="A21"/>

<SenderIdentification v="10X--TRADER03---" codingScheme="A01"/>

<SenderRole v="A29"/>

<ReceiverIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<ReceiverRole v="A07"/>

<CreationDateTime v="2010-03-20T19:30:47Z"/>

<ApplicableTimeInterval v="2010-07-01T05:00Z/2010-07-02T05:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<DocumentStatus v="A02"/>

<RightsTimeSeries>

<TimeSeriesIdentification v="1002"/>

<BusinessType v="A32"/>

<InArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<OutArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<RightsHolder v="10X--TRADER03---" codingScheme="A01"/>

<TransfereeParty v="10X--TRADER02---" codingScheme="A01"/>

<ContractIdentification v="10X--TRADER03---\_01355"/>

<ContractType v="A04"/>

<MeasureUnitQuantity v="MAW"/>

<AuctionIdentification v="BAME-A-01012010-01355"/>

<Period>

<TimeInterval v="2010-07-01T05:00Z/2010-07-02T05:00Z"/>

<Resolution v="PT60M"/>

<Interval>

<Pos v="1"/>

<Qty v="10"/>

</Interval>

...

<Interval>

<Pos v="24"/>

<Qty v="10"/>

</Interval>

</Period>

</RightsTimeSeries>

</RightsDocument>

#### Capacity Resale

Example of the Rights Document with capacity resale for source auction BAME-A-01012010-00124, target auction BAME-M-01042010-00554.

<?xml version="1.0" encoding="UTF-8"?>

<RightsDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://auctions.seecao.com/xsd/rights-document.xsd" DtdVersion="4" DtdRelease="0">

<DocumentIdentification v="A19\_A04\_10X--TRADER03---\_10001"/>

<DocumentVersion v="1"/>

<DocumentType v="A19"/>

<SenderIdentification v="10X--TRADER03---" codingScheme="A01"/>

<SenderRole v="A29"/>

<ReceiverIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<ReceiverRole v="A07"/>

<CreationDateTime v="2010-03-20T08:30:47Z"/>

<ApplicableTimeInterval v="2010-04-01T05:00Z/2010-05-01T05:00Z"/>

<Domain v="10YDOM-1010A024Y" codingScheme="A01"/>

<DocumentStatus v="A02"/>

<RightsTimeSeries>

<TimeSeriesIdentification v="10001"/>

<BusinessType v="A32"/>

<InArea v="10YCS-CG-TSO---S" codingScheme="A01"/>

<OutArea v="10YBA-JPCC-----D" codingScheme="A01"/>

<RightsHolder v="10X--TRADER03---" codingScheme="A01"/>

<ContractIdentification v="10X--TRADER03---\_00124"/>

<ContractType v="A04"/>

<MeasureUnitQuantity v="MAW"/>

<AuctionIdentification v="BAME-M-01042010-00554"/>

<Period>

<TimeInterval v="2010-04-01T05:00Z/2010-05-01T05:00Z“/>

<Resolution v="P1D"/>

<Interval>

<Pos v="1"/>

<Qty v="80"/>

</Interval>

...

<Interval>

<Pos v="30"/>

<Qty v="80"/>

</Interval>

</Period>

</RightsTimeSeries>

</RightsDocument>

### Acknowledgement Document

#### Acknowledgement Document Description

The acknowledgement document is sent to sender to acknowledge receipt of the document identified in the acknowledgement. This document is used to confirm each data flows for submitting or modifying data in Damas. The acknowledgement document notifies document recipient of any errors identified when processing document in Damas. If there are no problems identified in document, the acknowledgement document will contain only header. Acknowledgement message is generated according to the ENTSO-E Acknowledgement Document v5r0.

The acknowledgement document header contains document identification, current date and time, and identification of the document sender and recipient. Except for this, received document identification and version are included in the ReceivingDocumentIdentification and ReceivingDocumentVersion elements.

The TimeSeriesRejection element contains identification and version of the time series in which some error appeared. Code and detailed description of the errors are stored in the Reason element. For each hour with wrong value, the TimeIntervalError element is listed with time interval defining hour and reference to error description.

#### Specification of Acknowledgement Document Elements

List of the XML elements included in the Acknowledgement Document element

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Document Identification | Unique identification of the acknowledgement of the document that has been received. | The naming convention is: ACK\_<DocumentType>\_<ReceivingDocumentIdentificaion>\_<ReceivingDocumentVersion> | Mandatory |
| Document Date and Time | Date and time of the transmission of the acknowledgement. | The time must be expressed in UTC as YYYY-MM-DDTHH:MM:SSZ. | Mandatory |
| Sender Identification | Identification of the party that is originator of the acknowledgement. | 10XCS-SEECAO---O  EIC code of the sender. A01 coding scheme. | Mandatory |
| Sender Role | Identification of the role that is played by sender. | A18 (Grid operator) | Mandatory |
| Receiver Identification | Identification of the party who is recipient of the acknowledgement. | EIC code of the receiver. A01 coding scheme. | Mandatory |
| Receiver Role | Identification of the role played by receiver. | A29 (Trader) – for auctions bids, capacity transfers and resales.  A08 (Balance responsible party) – for nominations | Mandatory |
| Receiving Document Identification | Unique identification of the document that has been received. | String | Mandatory |
| Receiving Document Version | Version of the document received. | Number equal or greater than 1. | Mandatory |
| Receiving Document Type | Type of the document received. | String. | Mandatory |
| Reason | Description of the errors discovered in received document. |  | Mandatory |
| Time Series Rejection | Rejection of the time series. |  | Optional |

List of the XML elements included in the Time Series Rejection element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Senders Time Series Identification | Sender’s identification of the time series instance that uniquely identifies the Capacity time series. | String. | Mandatory |
| Reason | Identifies the nature of the time series for which product is handled. |  | Optional |
| Time Interval Error | Time interval containing errors. |  | Optional |

List of the XML elements included in the Time Interval Error element:

|  |  |  |  |
| --- | --- | --- | --- |
| **Element** | **Description** | **Values** | **Applicability** |
| Quantity Time Interval | Start and end date and time of the time interval of the period. | Coded in UTC.  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | Mandatory |
| Reason | Resolution defining the number of the periods that time interval is divided. |  | Mandatory |

List of the XML elements included in the Reason element:

|  |  |  |  |
| --- | --- | --- | --- |
| **Element** | **Description** | **Values** | **Applicability** |
| Reason Code | Code providing the acknowledgement status. | See Reason Code list. | Mandatory |
| Reason Text | Textual description of the rejection. | See Reason Code list. | Mandatory |

**Acknowledgement Document Example**

<?xml version="1.0" encoding="UTF-8"?>

<AcknowledgementDocument xmlns="http://auctions.seecao.com/xsd/AcknowledgementDocument.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" DtdVersion="3" DtdVersion="5" DtdRelease="0">

<DocumentIdentification v="20100801\_000000758463251"/>

<DocumentDateTime v="2010-07-31T12:56:44Z"/>

<SenderIdentification v="10XCS-SEECAO---O" codingScheme="A01"/>

<SenderRole v="A18"/>

<ReceiverIdentification v="10X--TRADER01---" codingScheme="A01"/>

<ReceiverRole v="A08"/>

<ReceivingDocumentIdentification v="20100801\_A01\_10X--TRADER01---"/>

<ReceivingDocumentVersion v="1"/>

<Reason>

<ReasonCode v="A02"/>

<ReasonText v="Message fully rejected"/>

</Reason>

<TimeSeriesRejection>

<SendersTimeSeriesIdentification v="284"/>

<SendersTimeSeriesVersion v="1"/>

<Reason>

<ReasonCode v="A20"/>

<ReasonText v="Time series fully rejected"/>

</Reason>

</TimeSeriesRejection>

</AcknowledgementDocument>

## Description of Damas specific XSD Schemas

This section describes the XSD adapted specifically for Damas as the standard ENTSO-E XSDs could not be used as-is.

### Long-term OC Information Document

#### Document Description

The Long-term Offered Capacity (LTOC) Information Document is the simple Damas specific document used to retrieve information about offered capacity for the Long-term auction. The ENTSO-E core components schema is used for specification of the elements inside.

#### LT OC Information Elements Description

List of the XML elements included in the MTOCInformation element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| Auction Identification | Unique identification of the set of the specifications that clearly identify the auction to which the capacity is addressed. | For example:  MEBA-A-06032010-00328 | Mandatory |
| Offered Capacity | Final Offered Capacity set for auction in general | Integer Number | Mandatory |
| Interval | One or more interval elements describing Reserve Price Interval |  | Optional |

List of the XML elements included in the Interval element:

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Description | Values | Applicability |
| TimeInterval | Planned Outage Period Time Interval | Coded in UTC.  YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | Mandatory |
| Offered Capacity | Offered Capacity for the defined Planned Outage Period | Integer number | Mandatory |

#### Long-term OC Information Examples

Reserve Price Information Document XSD schema definition

<?xml version="1.0" encoding="UTF-8"?>

<MTOCInformation xmlns="http://auctions.seecao.com/xsd/mtoc.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ecc="ENTSO-E-core-cmpts.xsd" DtdVersion="1" DtdRelease="0">

## Code Lists

The Code Lists listed below are either original ENTSO-E standard lists (or extract from it), or Damas specific, or sometimes with mixed content. The detailed information about type of the Code List and the contents is described in particular Code List chapter.

### Role

Code list for Role element is defined according to the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Role Name | Description |
| A04 | System operator | This code should be used TSOs. |
| A07 | Transmission capacity allocator | This code should be used by Auction Office. |
| A18 | Grid operator | This code is used for Border Administrator |
| A29 | Trader | This code should be used by Traders. |

### Capacity Contract Type

Code list for Capacity Contract Type element is extension of the ENTSO-E CodeList v6r4. Code List contains also Damas specific codes. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Contract Type Name | Description |
| A01 | Daily | This code should be used for capacity allocated by Daily auction or daily transmission procedures. |
| A03 | Monthly | This code should be used for capacity allocated by monthly auction or monthly transmission procedures. |
| A04 | Annual | This code should be used for capacity allocated by annual auction or annual transmission procedures. |
| A05 | Total | This is sum of all capacity contract types for period covered. |
| Z06 | Long-term contract | This code should be used for capacity allocated by all Long-term auctions |

### Measurement Unit

Code list for Measurement Unit element is defined according to the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Measurement Unit Name | Description |
| MAW | Mega watt | Unit of the bulk power flow, which can be defined as rate of the energy transfer/consumption where current of the 1000 amperes flows due to potential of the 1000 volts at unity power factor expressed in millions of the watt. |

### Message Type

Code list for Message (Document) Type is extension of the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Message Type Name | Description |
| A01 | Balance responsible schedule | The schedule that has been prepared by Nominator providing planned schedule information. This code should be used for schedules. |
| A02 | Allocated capacity schedule | The schedule that has been prepared by capacity allocator providing allocated capacity. |
| A13 | Interconnection Capacity | The document for cross-border capacity exchanges. This code should be used for Offered Capacity files. |
| A19 | Capacity for Resale | The document providing information about capacity for resale. |
| A21 | Capacity transfer notification | The document notifying a capacity transfer. |
| A23 | Allocations | The document providing capacity allocations for Border. |
| A24 | Bid document | The document providing bid information. |
| A25 | Allocation result document | The document providing allocation results of the auction. |

### Process Type

Code list for Process Type is defined according to the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Process Type Name | Description |
| A01 | Day ahead | The information provided concerns the day-ahead schedule. |
| A07 | Capacity allocation | The information provided concerns the capacity allocation process. |

### Classification Type

Code list for Classification Type is defined according to the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Classification Type Name | Description |
| A01 | Exchange type | The schedule is classified as providing detailed trades between two entities (all external trades between two entities). It is used for data without aggregation. |

### Coding Scheme

Code list for Coding Scheme is extension of the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Coding Scheme Name | Description |
| A01 | ENTSO-E | The coding scheme for preceding attribute is the ENTSO-E Identification Coding Scheme (EIC), maintained by ENTSO-E. |

### Business Type

Code list for Business Type is extension of the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Business Type | Description |
| A03 | External trade explicit capacity | The nature of the business being described is external trade details between two areas with limit capacity requiring allocation information. |
| A26 | Available Transfer Capacity (ATC) | Available transfer capacity, resp. the Offered Capacity. |
| A32 | Capacity transfer notification | The time series provides information concerning notification of the transfer of the capacity to another market participant, |

### Object Aggregation

Code list for Object Aggregation is extension of the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Role Name | Description |
| A01 | Area | The object being described concerns area. |
| A03 | Party | The object being described concerns party. |

### Currency

Code list for Currency is defined according to the ENTSO-E CodeList v6r4. Following codes are used in Damas:

|  |  |  |
| --- | --- | --- |
| Code | Currency Name | Description |
| EUR | EURO | The European legal tender. |

### Reason Codes

Code list for Reason codes is defined according to the ENTSO-E CodeList v6r4. The following codes are used in Damas:

|  |  |  |  |
| --- | --- | --- | --- |
| Code | Reason Code Name | Acknowledgement | Description |
| A01 | Message fully accepted | Message | The message has been fully accepted for application processing. |
| A02 | Message fully rejected | Message | No part of the message has been accepted for application processing. |
| A03 | Message contains errors at the time series level | Message | Part of the message contents (i.e. certain time series) has been accepted for application processing. It is necessary to look at time series level to determine time series that have been rejected. The time series is excluded from global position. |
| A04 | Time interval incorrect | Message/Period | The schedule time interval is not within contractual agreement or period does not agree with schedule time interval. |
| A05 | Sender without valid contract | Message | The sender has no current valid contract with the TSO. The message consequently will be fully rejected. |
| A10 | Credit limit exceeded | Message | The contractual credit limit has been exceeded. |
| A20 | Time series fully rejected | TS | The time series has been fully rejected. In case of the confirmation report, this reason code is used in conjunction with either A26 or A30. |
| A21 | Time series accepted with specific time interval errors | TS | The time series has been accepted but some time interval quantities have been rectified or zeroed out |
| A22 | In party / Out party invalid | TS | There is no contract for parties indicated or rules for cross border nominations are not being respected. The time series has been rejected |
| A23 | Area invalid | TS | The area is unknown or not allowed. The time series has been rejected. |
| A27 | Cross border capacity exceeded | TS | The cross border capacity has been exceeded. The time series has been rejected or adjusted |
| A41 | Resolution inconsistency | TS | The resolution is not coherent with the time interval, or resolution not valid |
| A42 | Quantity inconsistency | Interval | The quantity is not coherent. For example time period with the same version number but different quantities or non permitted number of the digits after decimal point, etc. |
| A46 | Quantities must not be signed values | Interval | The quantity proposed is illegal since signed values are only permitted in specific circumstances |
| A49 | Position inconsistency | Interval | The position is missing or too many. |
| A50 | Senders time series version conflict | TS | There is error in senders’ time series version. (i.e. it could be superior to message version or is inconsistent with existing data). The time series has been rejected. |
| A51 | Message identification or  version conflict | Message | The message identification is already in receiving system. Or higher version already exists. Message rejected |
| A52 | Time series missing from new version of message | Message | The time series is not contained in new version of the message. Message rejected |
| A53 | Receiving party incorrect | Message | The receiving party is incorrect. Message rejected |
| A54 | Global position not in balance | Message | The message does not balance out to zero. Market rules might require that message is rejected. |
| A55 | Time series identification conflict | TS | The identification of the time series is duplicated or incorrect. Time series will be rejected |
| A56 | Corresponding Time series not netted | TS | All corresponding time series must be netted. Time series rejected. |
| A57 | Deadline limit exceeded / Gate not open | Message / TS | The deadline for receiving schedule messages has passed. Message or time series rejected. |
| A58 | One to one nomination  inconsistency | TS | There is one to one nomination inconstancy with in/out parties or areas. Time series rejected |
| A59 | Not compliant to local market rules | Message / TS / Interval | The level in which this is identified is not in compliance with local market rules. The level in question has been rejected. |
| A60 | Inter-area transit schedule exceeds nominated schedule | Message | The inter-area transit schedule exceeds nominated schedule for the same time interval. The inter-area transit schedule is rejected. |
| A61 | Currency invalid | TS | The currency is not in compliance with ISO 4217. |
| A62 | Invalid business type | TS | The business type does not belong to valid set of the business types for process in question. |
| A64 | Resource Object Invalid | TS | The Resource Object defined in document is not valid |
| A94 | Document cannot be processed by receiving system |  | The receiving system cannot process that document in question |
| 999 | Errors not specifically identified |  | This code is used to identify errors that have not been specifically addressed in the Reason code list. It can be used at any level and refers to level for which it has been identified. |

### Curtailment Types

Damas specific Code List for Curtailment Types.

|  |  |  |
| --- | --- | --- |
| Code | Type Name | Description |
| A01 | Capacity Curtailment | Allocated capacity curtailed and compensated |
| A99 | No Curtailment | No Curtailment applied |

### Invoicing Data Types

Damas specific codes for Invoicing Data Types:

|  |  |  |
| --- | --- | --- |
| Code | Type Name | Description |
| A01 | Auctions | Invoices for capacity gained in auctions performed in Damas |
| A02 | Resales | Resold Resales Compensations |
| A03 | UIOSI | Resold UIOSI Compensations |
| A04 | Curtailment | Curtailment Compensations |

### Invoice States

Damas specific codes for Invoice States:

|  |  |  |
| --- | --- | --- |
| Code | Type Name | Description |
| A01 | Unpaid | Invoice has been generated, but not yet paid or overdue |
| A02 | Overdue | Invoice is overdue |
| A03 | Paid | Invoice has been paid |

# Appendix A – Examples of Xml Messages

|  |  |  |
| --- | --- | --- |
| Data Flows | FID | XML Message |
| Submit / Modify Auction Bids | DMSWS\_BID\_IN |  |
|  |
| Submit / Modify Capacity Transfers | DMSWS\_TRA\_IN |  |
| Submit / Modify Capacity Resale | DMSWS\_RES\_IN |  |

# Appendix B – Examples of SOAP Request Messages

|  |  |  |
| --- | --- | --- |
| Data Flows | FID | SOAP Message |
| Submit / Modify Auction Bids | DMSWS\_BID\_IN |  |
| Submit / Modify Capacity Transfers | DMSWS\_TRA\_IN |  |
| Submit / Modify Capacity Resale | DMSWS\_RES\_IN |  |
| Download Offered Capacity | DMSWS\_ATC\_OUT |  |
| Download Detailed Auction Results | DMSWS\_DAR\_OUT |  |
| Download ACH | DMSWS\_ENT\_OUT |  |
| Download Actual Date and Time | GETDATETIME |  |
| Download Long-term OC and Planned Outage Periods | DMSWS\_POP\_OUT |  |
| Download Invoicing Data | DMSWS\_INV\_OUT |  |
| Download Auction Specification (CASD) | DMSWS\_CASD\_OUT |  |

# Appendix C – Configuration of SoapUI

SoapUI is a tool that can be used for initial phase when it is necessary to have a valid SOAP message for Damas WS interface.

Configuration and usage of the tool SoapUI:

* After installation run it and create new SOAP project by inserting WSDL web address which is provided in article 5.3 of this document.
* Select GetActualDateTime and click Request 1. Once the request is open, in the request properties tab insert username and password (which has to be Base64 encoded). Then right click on the request and select Add WSS Username Token and Add WS-Timestamp to add credentials and timestamp to the request. Finally, execute the request.
* Select action RunSynchrous and create new request. Add WSS Username Token and new WS-Timestamp. Insert FID and StringParam contents from the SOAP request files in Appendix B and then execute the request.

