# Technical specification of external interfaces for market participants

## V1.20

# MARKET ORGANIZER INFORMATION SYSTEM XMtrade®/ISOT

sféra
Grafické informačné systémy

***sféra,*** a.s. • Karadžičova 2 • 811 08 Bratislava
tel.: +421 2 502 13 142 • fax: +421 2 502 13 262

# TABLE OF CONTENT

# History of Changes

| Date | Version | Description | Author |
|---|---|---|---|
| 6/1/2009 | 1.0 | Document processing | Mgr. Miroslav Galajda,<br>Mgr. Libor Láznička |
| 6/3/2009 | 1.1 | Update of order structure | Mgr. Miroslav Galajda |
| 7/1/2009 | 1.2 | Detailed clarification of information to provided results and evaluations of DM.<br>Elaboration of examples to individual data flows.<br>Refinement of SOAP header specification.<br><br>Payments and fees are specified excluding VAT and tax on electricity. | Mgr. Miroslav Galajda,<br>Mgr. Libor Láznička |
| 8/18/2009 | 1.3 | Addition of service specification for publication of MCC values for market participant. | Mgr. Miroslav Galajda |
| 1/27/2012 | 1.4 | Introduction of negative prices on DM. | Bc. Anton Weissensteiner |
| 12/3/2014 | 1.5 | Hyperlinks change to a new web address:<br>https://isot.okte.sk,<br>https://test-isot.okte.sk. | Mgr. Svetlana Pražienková |
| 2/3/2015 | 1.6 | Description of processes correction in accordance with 4MMC – CZ/SK/HU/RO project. | Ing. Jozef Drgoňa |
| 2/23/2016 | 1.7 | Addition of specification for orders on intraday market and results retrieval of intraday for market participant | Ing. Jozef Drgoňa |
| 6/23/2016 | 1.8 | Correction of examples for data flow E-07_01 and E-07_02. Removal of the element *Period* from ISOTEDATA-VDT messages. | Ing. Jozef Drgoňa |
| 9/5/2017 | 1.9 | Addition of service specification for broader communication possibilities like managing intraday market orders, results of intraday market and automated notifications about events on intraday market using AMQP protocol. | Ing. Robert Maier |
| 10/1/2018 | 1.10 | Expansion of existing data flows E-06_03 and E-10_01, which contains anonymous user identifier of the user placing intraday market order. | Ing. Robert Maier |
| 11/2/2020 | 1.11 | Update of network TLS protocol support. | Ing. Robert Maier |

| Date | Version | Description | Author |
|---|---|---|---|
| 01/24/2021 | 1.12 | Extension of day-ahead market products with possibility to submit unlimited number of orders per MP and introduction of new simple block, linked block, flexible block products and exclusive group of block orders. | Ing. Robert Maier |
| 04/30/2021 | 1.13 | Extension of DAM evaluation with the parameter for specification of result status. | Ing. Robert Maier |
| 03/02/2022 | 1.14 | Disabling of access to cross-border capacities for CORE regime of cross-border day-ahead market. | Ing. Robert Maier |
| 09/27/2022 | 1.15 | Extension of external interfaces for intraday market according to products extension and creation of cross-border intraday market within the SIDC project. | Ing. Robert Maier |
| 01/17/2024 | 1.16 | Extension of IDM data flows by trading screen statistics in extent:<br>• Last price/quantity traded in the trading period,<br>• Overall traded quantity within the period. | Ing. Robert Maier |
| 03/22/2024 | 1.17 | Extension of external interfaces for communication of market participants within the intraday auctions (IDAs). | Ing. Robert Maier |
| 07/10/2024 | 1.18 | New SANDBOX environment https://sandbox-isot.okte.sk available for market participants. | Ing. Robert Maier |
| 11/21/2024 | 1.19 | New interfaces WEB API and WebSocket | Ing. Marcel Struhár |
| 12/11/2024 | 1.19 | Incorporation of feedback from consultations with market participants | Ing. Marcel Struhár |
| 02/10/2025 | 1.20 | Expansion of external interfaces for market participants' communication within 15 MTU. | Ing. Marcel Struhár |
| 03/24/2025 | 1.20 | Incorporation of feedback from consultations with market participants | Ing. Marcel Struhár |

# 1 INTRODUCTION

## 1.1 Characteristics of the document

### 1.1.1 Purpose of the document

The purpose of this document is to provide all necessary technical information for the implementation of automated data exchange between an external system of a market participant and the market organizer information system XMtrade®/ISOT. This document contains specification of communication methods and data structures that are used for data exchange.

### 1.1.2 Specification of the document

The document is for system implementers who are preparing integration with the market organizer information system XMtrade®/ISOT.

# 2 OVERVIEW OF EXTERNAL INTERFACES

The market organizer information system XMtrade®/ISOT provides automated interfaces for data exchange within processes of the day-ahead, intraday continuous and intraday auctionorder registration, and results and evaluations retrieval of day-ahead market based on web services that are used by trading systems of market participants.

**Table 1** Automated interface for data exchange of processes within day-ahead order registration

| ID | Name | Description |
|---|---|---|
| OB-01 | Management of market participant orders | Market participants are provided with an automated interface for submission and retrieval of own orders for purchase and sell on day-ahead market. |
| OB-02 | Management of intraday market participant orders | Market participants are provided with an automated interface for submission, modification and retrieval of own orders for purchase and sell on intraday market. |
| OB-03 | Access to intraday market order book | Market participants are provided with automated interface for intraday order book retrieval in its current status and available cross-border capacities. |
| OB-04 | Management of market participant orders for intraday auctions | Market participants are provided with an automated interface for submission and retrieval of own orders for purchase and sell on intraday auctions. |
| EV-01 | Results and evaluation of DM | Market participants are provided with automated interface for day-ahead market results and evaluations retrieval. |
| EV-02 | Results of IDM | Market participants are provided with automated interface for intraday market results and evaluations retrieval. |
| EV-03 | Results and evaluation of IDA | Market participants are provided with automated interface for intraday auctions results and evaluations retrieval. |
| SR-01 | MCC details | Market participants are provided with an automated interface for MCC values retrieval. |
| AMQP-01 | Notifications of IDM | Market participants are provided with an automated interface for automatic notifications about their own intraday orders, data from intraday order book and change of available cross-border capacities within intraday market. |

## 2.1 Overview of data flows

Basic communication scenarios are primarily dependent on the task that the market organizer currently performs:

- organizing of day-ahead market exclusively within domestic trading area,
- coordinated organizing of day-ahead market in multiple trading areas.
- organizing of intraday market exclusively within domestic trading area,
- coordinated organizing of intraday auctions within several trading areas.

### 2.1.1 Organization of domestic day-ahead market

Within the organization of domestic short-term day-ahead market, communication is established between the market organizer information system XMtrade®/ISOT, (ISOT) and systems of market participants (ISMP) through web services (Figure 1). Using automated method, market participants are able to submit orders into the PXS system, retrieve results and evaluations of day-ahead market.

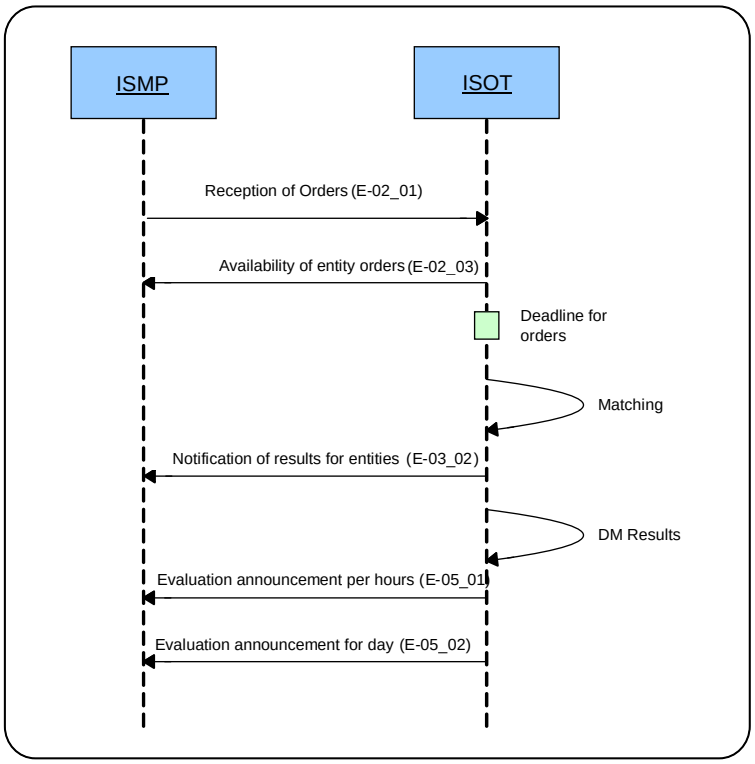


**Figure 1** Communication scenario in domestic day-ahead market mode

**Table 2** Overview of interfaces in domestic day-ahead market mode

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-02_01 | Reception of orders:<br>- allows market participant to submit orders by the deadline for order reception. | ISMP | ISOT | ISMP |
| E-02_03 | Availability of entity orders:<br>- allows market participant to retrieve information on own orders entered into ISOT. | ISOT | ISMP | ISMP |
| E-03_02 | Notification of results for entities:<br>- allows market participant to retrieve information on DM results after order matching is finished. | ISOT | ISMP | ISMP |
| E-05_01 | Evaluation announcement per hours:<br>- allows market participant to retrieve detailed information on DM evaluation after the end of DM matching. | ISOT | ISMP | ISMP |
| E-05_02 | Evaluation announcement for day:<br>- allows market participant to retrieve summary information on DM evaluation after the end of DM matching. | ISOT | ISMP | ISMP |

### 2.1.2 Coordinated organization of day-ahead market

Within the coordinated organization of short-term day-ahead market, communication is established between the market organizer information system XMtrade®/ISOT (ISOT) and systems of market participants (ISMP) through web services (Figure 2). Information on MCC results, evaluations of day-ahead market and market participants orders entered into ISOT system, are made available through an automated method.

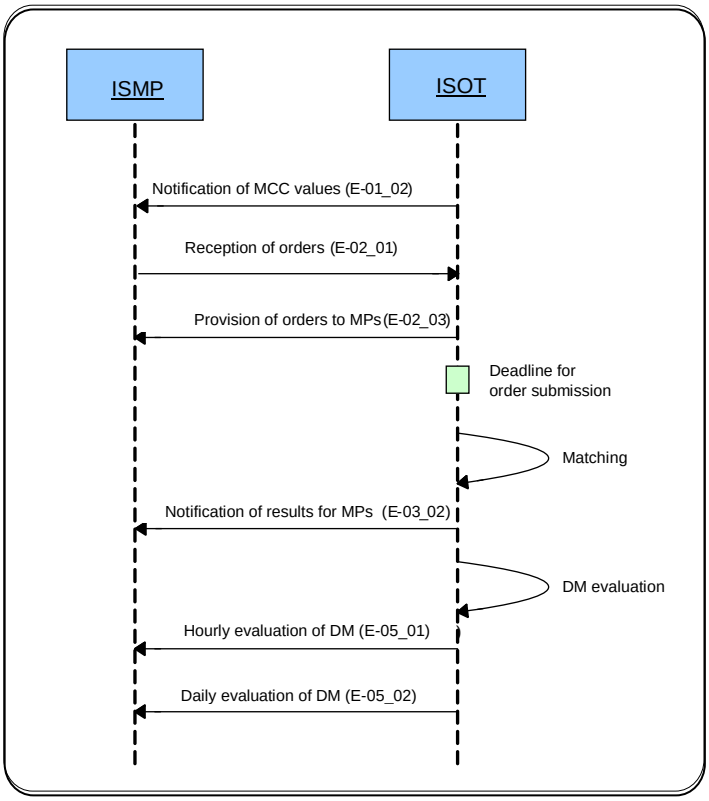


**Figure 2** Communication scenario in coordinated day-ahead market mode

**Table 3** Overview of interfaces in day-ahead market mode

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-01_02 | Notification of MCC values:<br>- allows market participant to retrieve information on MCC after the publication of MCC. | ISOT | ISMP | ISMP |
| E-02_01 | Reception of orders:<br>- allows market participant to submit orders by the deadline for order reception. | ISMP | ISOT | ISMP |
| E-02_03 | Availability of entity orders:<br>- allows market participant to retrieve information on own orders entered into ISOT. | ISOT | ISMP | ISMP |
| E-03_02 | DM evaluation announcement for entities:<br>- allows market participant to retrieve information on DM results after the end of matching. | ISOT | ISMP | ISMP |
| E-05_01 | DM evaluation announcement per hours:<br>- allows market participant to retrieve detailed information on DM evaluation after the end of DM matching. | ISOT | ISMP | ISMP |
| E-05_02 | DM evaluation announcement for day:<br>- allows market participant to retrieve summary information on DM evaluation after the end of DM matching. | ISOT | ISMP | ISMP |

### 2.1.3 Organizing domestic intraday market

Within the organization of domestic intraday market, communication is established between the market organizer information system XMtrade®/ISOT (ISOT), and systems of market participants (ISMP) through web services (Figure 3) and through interface based on AMQP communication protocol (Figure 4). Using automated methods, market participants submit orders into the ISOT system and receive related results and evaluations of intraday market and using the AMQP protocol, notifications about change of current market status are distributed to market participants.

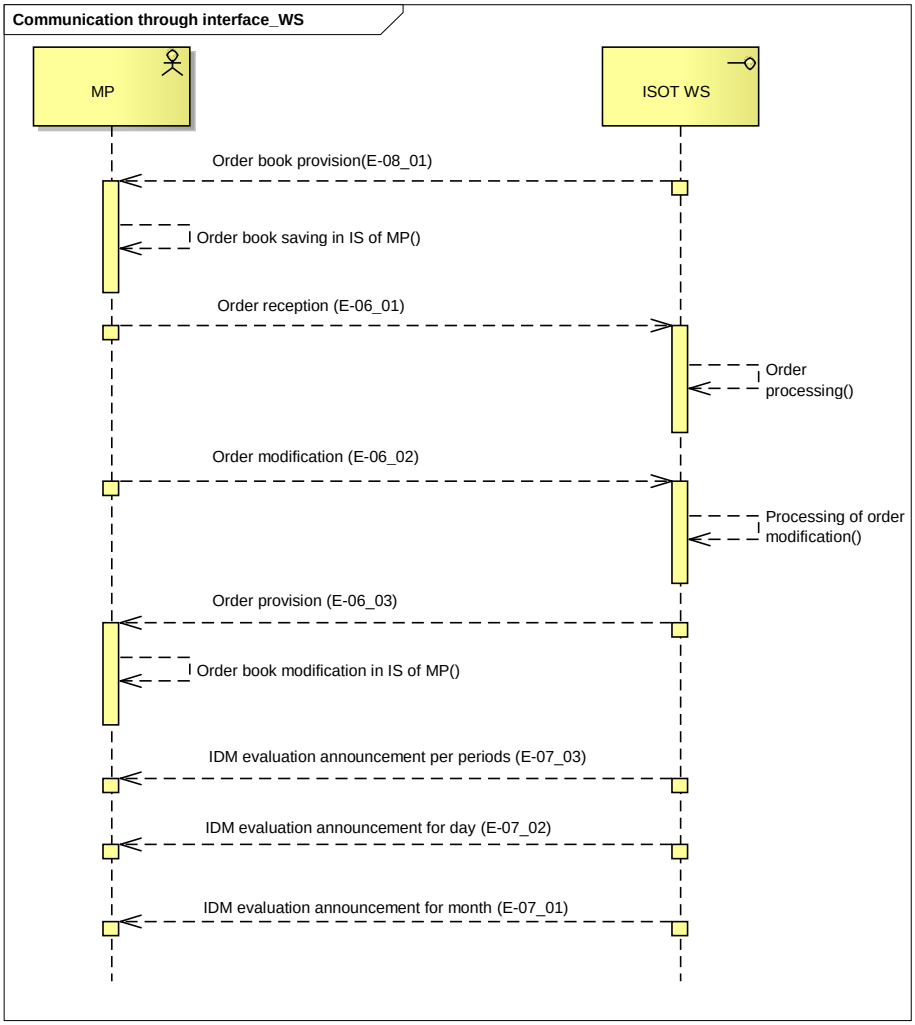


**Figure 3** Comunication scenario of the web services in domestic intraday market mode

**Table 4** Overview of interfaces in domestic intraday market mode

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-06_01 | Reception of intraday orders:<br>- allows market participant to submit orders until the gate closure order reception. | ISMP | ISOT | ISMP |
| E-06_02 | Modification of intraday orders:<br>- allows market participant to modify (activate, deactivate, cancel) own existing order. | ISMP | ISOT | ISMP |

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-06_03 | Availability of intraday entity orders:<br>- allows market participant to retrieve information on own orders entered into ISOT. | ISOT | ISMP | ISMP |
| E-07_01 | IDM evaluation announcement for day:<br>- allows market participant to access information about intraday evaluation after trading day closure and after intraday evaluation is finished. | ISOT | ISMP | ISMP |
| E-07_02 | IDM evaluation announcement for month:<br>- allows market participant to access information about intraday evaluation after trading month closure and after intraday per given month evaluation is finished. | ISOT | ISMP | ISMP |
| E-07_03 | IDM evaluation announcement per periods:<br>- allows market participant to access information about their intraday orders per periods. | ISOT | ISMP | ISMP |
| E-08_01 | Order book provision:<br>- allows market participant to access immediate data from order book (available quantities and prices) on intraday market. | ISOT | ISMP | ISMP |

Web services described above are intended for full automation of communication with ISOT system, which includes interface extension that uses AMQP or WebSocket protocol and allows market participant to receive notifications about real-time changes on IDM.

Through notifications, market participant is informed about these events:

- Successful creation of own order,
- Modification of own order (change of status),
- Order book status change (increase/decrease of available quantity).

**Table 5** Overview of AMPQ interface and WebSocket in domestic intraday market mode

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-10_01 | Change of status/creation of own order:<br>- informs market participant about successful creation or modification of own order. | ISMP | ISOT | ISOT |
| E-10_02 | Order book status change:<br>- informs market participant about change in order book status (increase/decrease of available quantity). | ISMP | ISOT | ISOT |

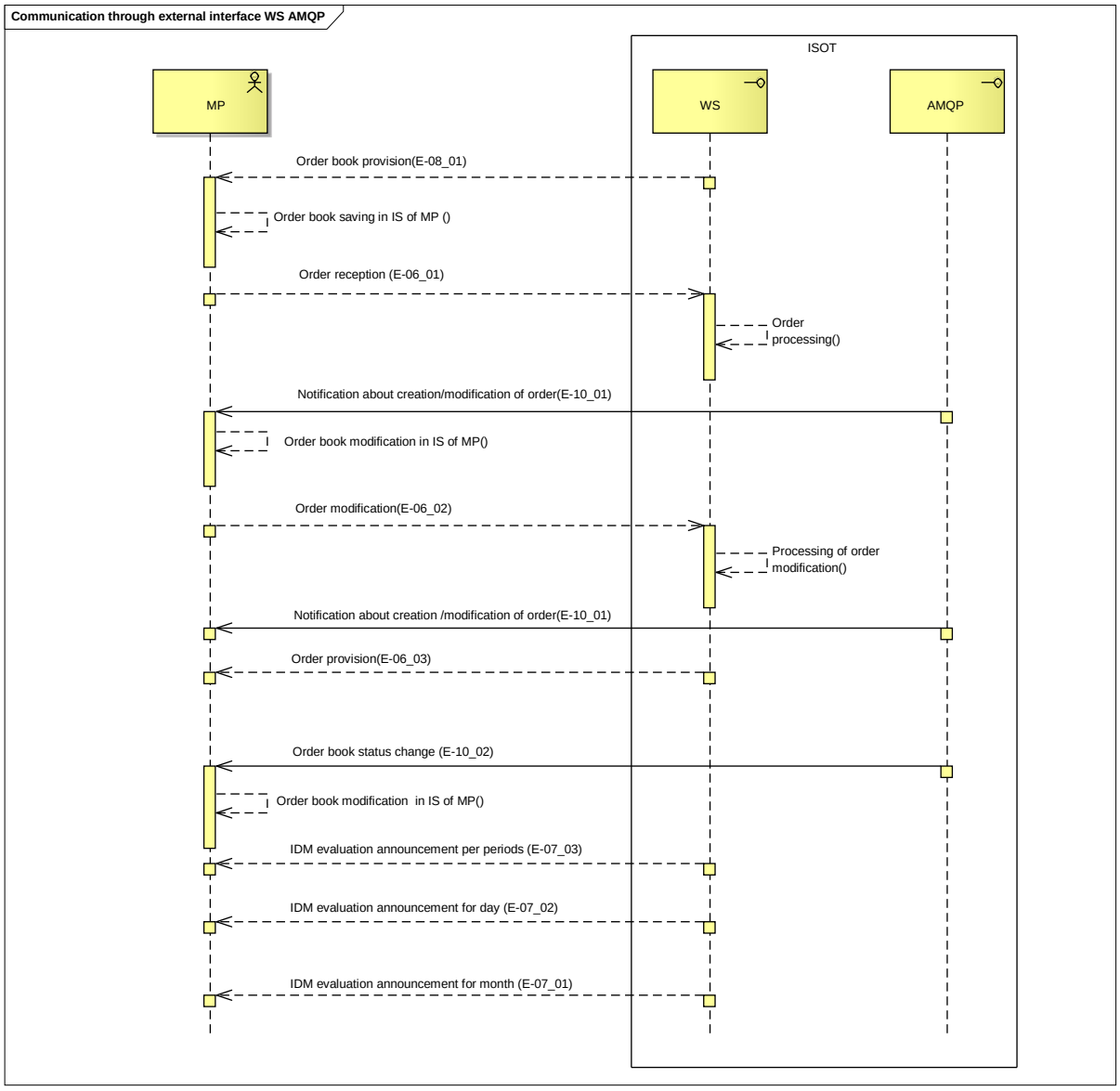


**Figure 4** Communication scenario of web services and AMQP interface in intraday domestic market mode

### 2.1.4 Coordinated organization of intraday market

Within the coordinated organization of intraday market, communication is established between the market organizer information system XMtrade®/ISOT (ISOT), and systems of market participants (ISMP) through web services (Figure 5) and through interface based on AMQP communication protocol (Figure 6). Using automated methods, market participants submit orders into the ISOT system and receive related results and evaluations of intraday market and using the AMQP protocol, notifications about change of current market status and available cross-border capacities are distributed to market participants.

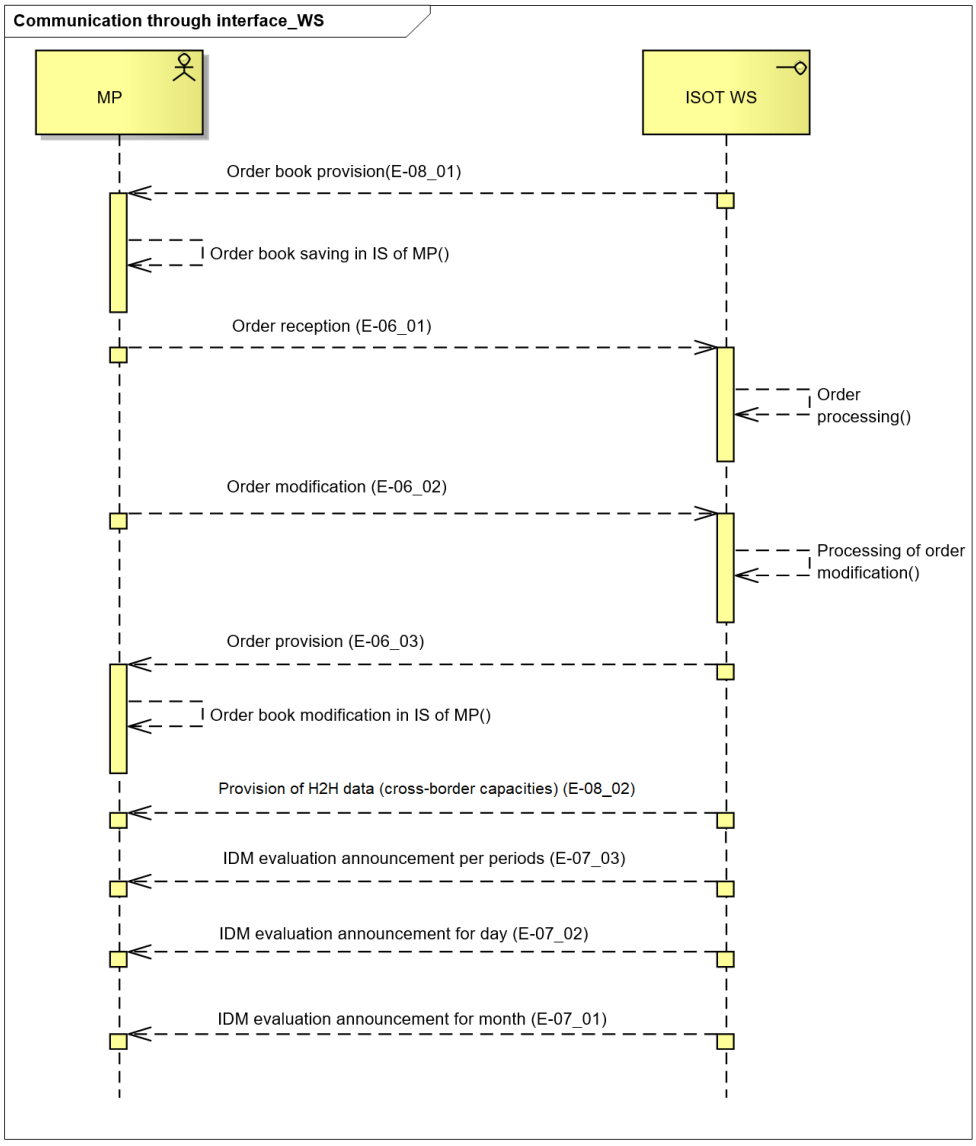


**Figure 5** Comunication scenario of the web services in coordinated intraday market mode

**Table 6** Overview of interfaces in coordinated intraday market mode

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-06_01 | Reception of intraday orders:<br>- allows market participant to submit orders until the gate closure order reception. | ISMP | ISOT | ISMP |

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|
| E-06_02 | Modification of intraday orders:<br>- allows market participant to modify (activate, deactivate, cancel) own existing order. | ISMP | ISOT | ISMP |
| E-06_03 | Availability of intraday entity orders:<br>- allows market participant to retrieve information on own orders entered into ISOT. | ISOT | ISMP | ISMP |
| E-07_01 | IDM evaluation announcement for day:<br>- allows market participant to access information about intraday evaluation after trading day closure and after intraday evaluation is finished. | ISOT | ISMP | ISMP |
| E-07_02 | IDM evaluation announcement for month:<br>- allows market participant to access information about intraday evaluation after trading month closure and after intraday per given month evaluation is finished. | ISOT | ISMP | ISMP |
| E-07_03 | IDM evaluation announcement per periods:<br>- allows market participant to access information about their intraday orders per periods. | ISOT | ISMP | ISMP |
| E-08_01 | Order book provision:<br>- allows market participant to access immediate data from order book (available quantities and prices) on intraday market. | ISOT | ISMP | ISMP |
| E-08_02 | Provision of Hub-to-Hub matrix (H2H) data with information about available cross-border capacities::<br>- allows market participant to access information about available cross-border capacities between SK market area and other delivery areas within SIDC.<br>This information serves for informative purposes only and its provision can be subject to delay against the real state within the central system for matching of intraday market orders within SIDC. | ISOT | ISMP | ISMP |

Web services described above are intended for full automation of communication with ISOT system, which includes interface extension that uses AMQP protocol and allows market participant to receive notifications about real-time changes on IDM.

Through notifications, market participant is informed about these events:

- Successful creation of own order,
- Modification of own order (change of status),
- Order book status change (increase/decrease of available quantity),
- Change of available cross-border capacities.

**Table 7** Overview of AMPQ interface in coordinated intraday market mode

| ID | Description | Source | Target | Initiator |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| E-10_01 | Change of status/creation of own order:<br>- informs market participant about successful creation or modification of own order. | ISMP | ISOT | ISOT |
| E-10_02 | Order book status change:<br>- informs market participant about change in order book status (increase/decrease of available quantity). | ISMP | ISOT | ISOT |
| E-10_03 | Change in Hub-to-Hub matrix (H2H) data with information about available cross-border capacities::<br>- informs market participant about a change in Hub-to-Hub matrix data containing available cross-border capacities between SK market area and other delivery areas within SIDC.<br>This information serves for informative purposes only and its provision can be subject to delay against the real state within the central system for matching of intraday market orders within SIDC. | ISMP | ISOT | ISOT |

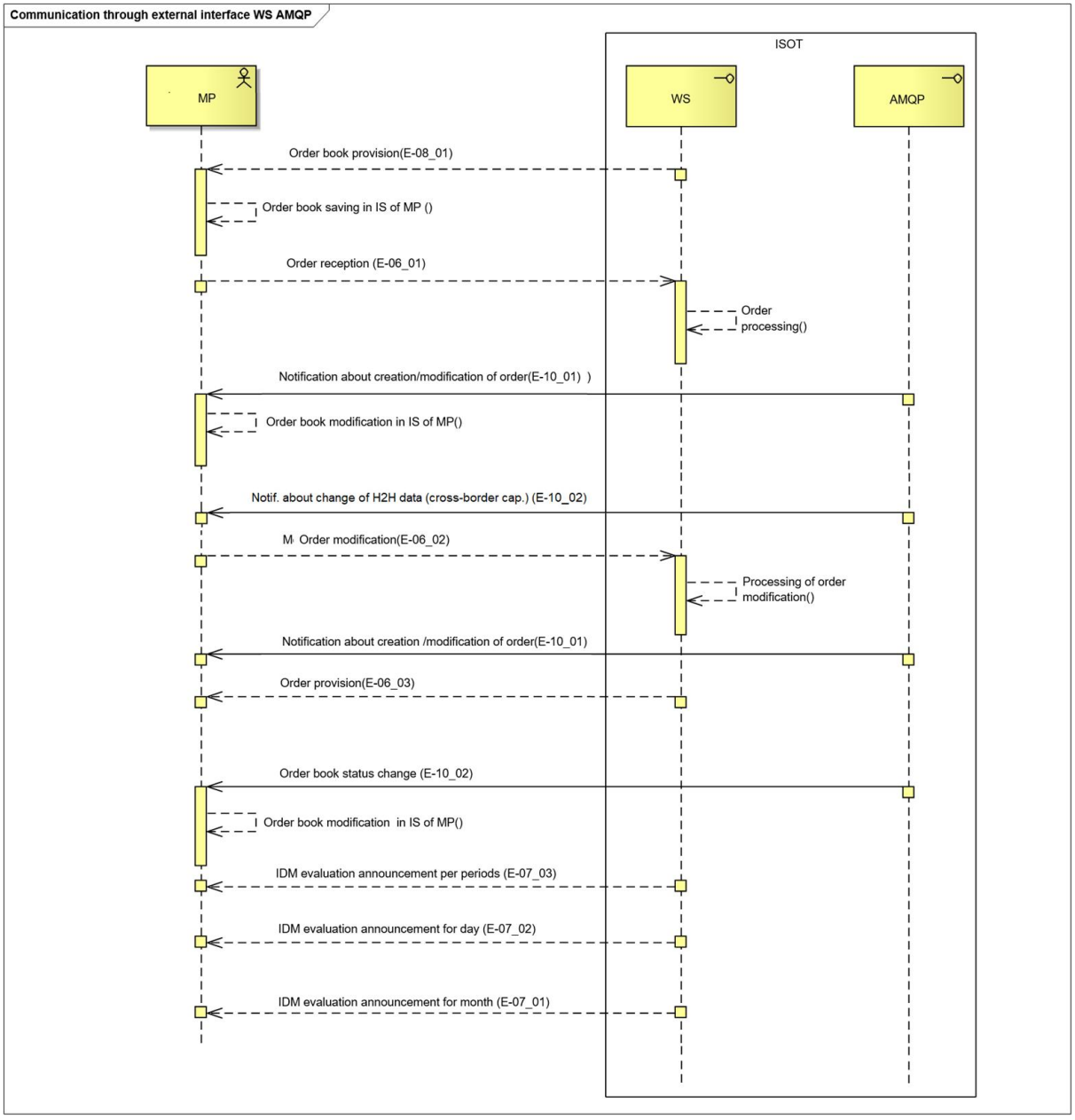


**Figure 6** Communication scenario of web services and AMQP interface in coordinated intraday market mode

### 2.1.5 Coordinated organization of intraday auctions

Within the coordinated organization of intraday auctions, communication is established between the market organizer information system XMtrade®/ISOT (ISOT) and systems of market participants (ISMP) through web services (Figure 7). Information on relevant auction results, evaluations of auctions and market participants orders entered into ISOT system, are made available through an automated method.

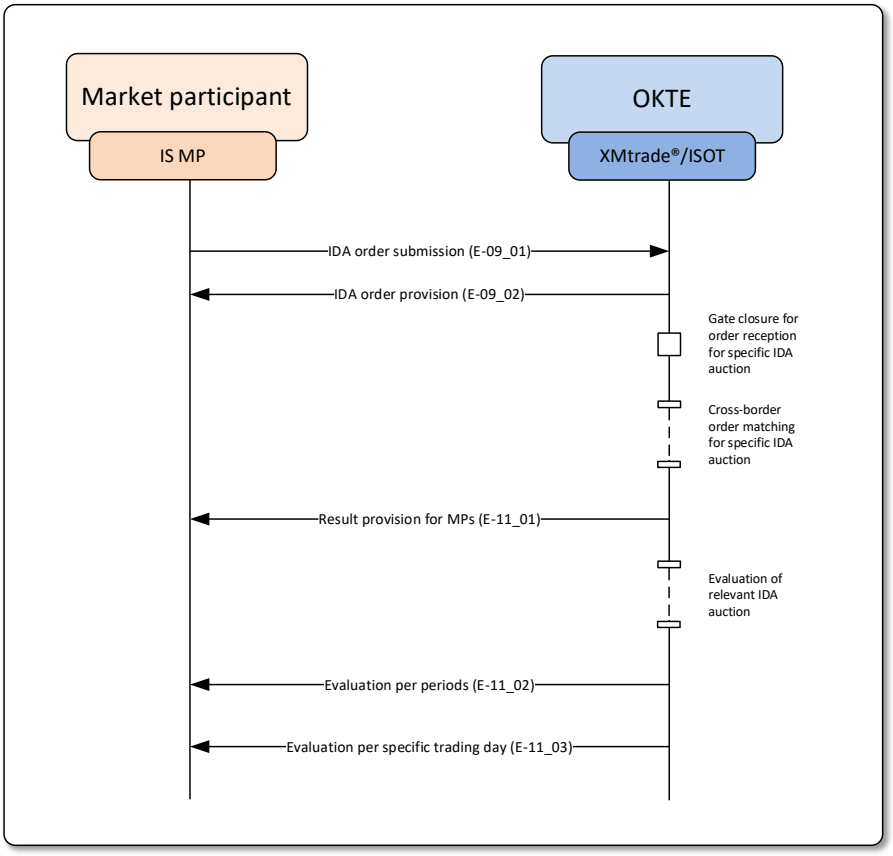


**Figure 7** Communication scenario in coordinated organization of intraday auctions

**Table 8** Overview of interfaces in intraday auctions mode

| ID | Opis | Zdroj | Cieľ | Iniciátor |
|---|---|---|---|---|
| E-09_01 | Reception of IDA orders:<br>- allows market participant to submit orders by the deadline for order reception for specific intraday auction. | ISÚT | ISOT | ISÚT |
| E-09_02 | Provision of IDA orders for market participants:<br>- allows market participant to retrieve information on own orders entered into ISOT for specific intraday auction. | ISOT | ISÚT | ISÚT |
| E-11_01 | IDA evaluation announcement for market participants:<br>- allows market participant to retrieve information on DM results after matching of specific intraday auction. | ISOT | ISÚT | ISÚT |
| E-11_02 | IDA evaluation announcement per hours:<br>- allows market participant to retrieve detailed information on IDA evaluation after matching of specific intraday auction. | ISOT | ISÚT | ISÚT |
| E-11_03 | ODA evaluation announcement for day:<br>- allows market participant to retrieve summary information on IDA evaluation after matching of specific intraday auction. | ISOT | ISÚT | IS ÚT |

# 3 SPECIFICATION OF COMMUNICATION

## 3.1 Web services

The market organizer information system XMtrade®/ISOT covers external interfaces with the following set of web services/web methods.

**Table 9** Overview of interfaces in the coordinated organization of day-ahead and intraday market mode

| ID | Name of web service | Name of web method | Description |
|---|---|---|---|
| OB-01 | Orders | Upload<br>Download | -market participants are provided with automated interface for submission and reception of own orders on day-ahead market |
| OB-02 | IdmOrders | Upload<br>Modify<br>Download | -market participants are provided with automated interface for submission, modification and reception of own orders on intraday market |
| OB-03 | IdmOrderBook | Download | - market participants are provided with automated interface for downloading intraday order book data and available cross-border capacities |
| OB-04 | IdaOrders | Upload<br>Download | - market participants are provided with automated interface for submission and reception of own orders on intraday auctions |
| EV-01 | Evaluations | Download | - market participants are provided with automated interface for retrieval of results/evaluations of day-ahead market |
| EV-02 | IdmEvaluations | Download | - market participants are provided with automated interface for retrieval of results/evaluations of intraday market |
| EV-03 | IdaEvaluations | Download | - market participants are provided with automated interface for retrieval of results/evaluations of intraday auctions |
| SR-01 | StatusRequest | DownloadMCC | - market participants are provided with automated interface for retrieval of MCC values |

### 3.1.1 Communication scenarios

#### *Synchronous communication*

Synchronous communication of web services in ISOT system can be in general depicted as follows:

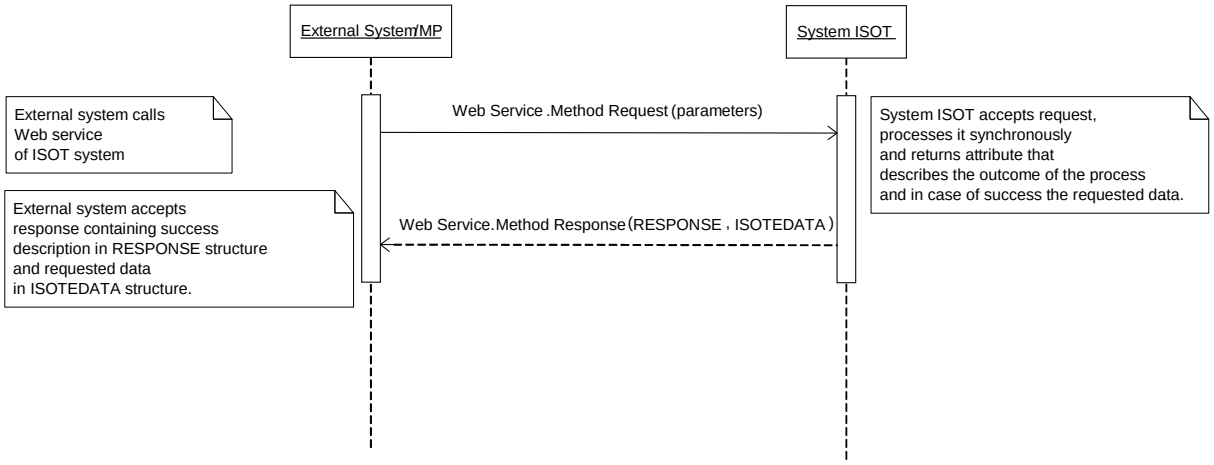


**Figure 8** The principle of synchronous communication.

Synchronous call of web method processes the request and returns corresponding response.

### 3.1.2 SOAP Protocol

The structure of SOAP messages is implemented in SOAP 1.2 version in accordance with recommendations of W3C consortium (http://www.w3.org/TR/soap12) and uses the following extensions:

- WS-Security (http://www.oasis-open.org/specs/index.php#wssv1.0),
- WS-Addressing (http://www.w3.org/Submission/2004/SUBM-ws-addressing-20040810).

In order to shorten the notation of individual SOAP messages, the following namespace aliases are used:

**Table 10** Namespace aliases

| Alias | Namespace |
|---|---|
| s | http://www.w3.org/2003/05/soap-envelope |
| o | http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd |
| a | http://schemas.xmlsoap.org/ws/2004/08/addressing |
| u | http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd |

Web services are implemented in the namespace of the following format:

http://sfera.sk/ws/xmtrade/isot/interfaces/**NameOfService**/services/**Version**

SOAP messages of systems web services contain two distinctive parts: header and body, while all messages of ISOT system are encoded in UTF-8. In addition to controlling data of the protocol, header contains details for authentication and authorization of the calling system (name, password and digital signature if applicable).

```
<s:Header>
  <!-- WS-Addressing -->
  <!-- WS-Security -->
</s:Header>
```

„WS-Security " contains security tokens necessary for source system authentication and verification of message integrity. This relates to electronic signature tokens, name and password of a user.

„WS-Addressing " contains details to secure addressing of a soap message. Detailed header structure can be found in this example.

Message body contains class element of the message for specific request. Body structure of the message can be defined in general as follows:

*Request:*

```
<s:Body>
  <MethodNameRequest xmlns=" http://sfera.sk/ws/xmtrade/isot/interfaces/
NameOfService/services/Version">
       <!—message document-->
  </MethodNameRequest>
</s:Body>
```

*Response:*

```
<s:Body>
  <MethodNameResponse xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/
NameOfService/services/Version">
       <!—message document -->
  </MethodNameResponse>
</s:Body>
```

### *SOAP Fault*

*SOAP Fault* element serves for generic transfer of error information, which are transferred within SOAP message in *<s:Fault>* element, in accordance with SOAP 1.2 specification (http://www.w3.org/TR/soap12-part1/#soapfault). It mainly relates to system error and exception handling during communication etc. However, defining custom types of Fault messages could be advantageously used to catch general application errors.

## 3.1.3 Orders

*Orders* web service allows market participants to use automated interface for submission and reception of own orders on day-ahead market.

The service implements the following methods:

- *Upload* – method for submission of order,
- *Download* – method for reception of own order.

### *SOAP Upload*

Upload method of *Orders* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Description of request structure*

**Table 11** Request structure descritpion – Upload method

| UploadRequest | Description |
|---|---|
| ISOTEDATA | Structure of order is in accordance with specification where message-code=811 (see description of E-02_01 data flow). |

*Description of response structure*

**Table 12** Response structure description - Upload method

| UploadResponse | Description |
|---|---|
| RESPONSE | Common return structure of request handling effectivity is in accordance with specification where message-code=812 (see description of data flow E-02_01). |
| ISOTEDATA | Structure of order is in accordance with specification where message-code=813 (see description data flow E-02_01). Description of order is returned as it was registered in the system. |

## *Example of SOAP message*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01" id="1"
message-code="811" date-time="2009-05-01T11:48:51" answer-required="1">
        <!-- order data -->
      </ns:ISOTEDATA>
    </ns:UploadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="812" ...>
        <!-- efficiency description -->
      </ns:RESPONSE>
      <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01"
message-code="813" ...>
        <!-- order description data -->
      </ns:ISOTEDATA>
    </ns:UploadResponse>
  </s:Body>
</s:Envelope>
```

## SOAP Download

*Download* method of *Orders* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Description of request structure*

**Table 13** Request structure description – Download method

| DownloadRequest | Description |
|---|---|
| CDSREQ | Request structure for reception of own order in accordance with specification, where message-code=831 (see description of E-02_03 data flow). |

*Description of response structure*

**Table 14** Response structure description – Download method

| DownloadResponse | Description |
|---|---|
| RESPONSE | Common return structure of request handling effectivity in accordance with specification, where message-code=832 (see description of E-02_03 data flow). |
| ISOTEDATA | Structure of order in accordance with specification, where message-code=833 (see description of E-02_03 data flow). Description of order is returned as it was registered in the system. |

### *Example of SOAP message*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:CDSREQ
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01" id="1"
message-code="831" date-time="2009-05-01T11:48:51" >
        <Trade trade-day="2009-05-01" />
      </ns:CDSREQ>
    </ns:DownloadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="832" ... >
            <!-- description of efficiency -->
      </ns:RESPONSE>
       <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01"
message-code="833" ... >
            <!-- description of order data -->
       </ns:ISOTEDATA>
    </ns:DownloadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.4 IdmOrders

*IdmOrders* web service allows market participants to use automated interface for submission, modification and reception of own orders on intraday market.

Service implements these methods:

- *Upload* - method for order submission,
- *Modify* – method for order modification (activation/deactivation/cancellation),
- *Download* – method for own order reception.

#### *SOAP Upload*

*Upload* method of *IdmOrders* service operates in synchronous mode, i.e. the request is handled by the response within the same call. Response to this request consists of confirmation of order submission and copy of order data from the request.

*Description or request structure*

**Table 15** Request structure description – Upload method

| UploadRequest | Description |
|---|---|
| ISOTEDATA-VDT | Structure of order is in accordance with specification, where message-code=801 (see description of E-06_01 data flow). |

*Description of request structure*

**Table 16** Response structure description - Upload method

| UploadResponse | Description |
|---|---|
| RESPONSE-VDT | Common structure of retrieving processing status is in accordance with specification, where message-code=802 (see description of E-06_01 data flow). |
| ISOTEDATA-VDT | Structure of order is in accordance with specification, where message-code=803 (see description of E-06_01 data flow). Returned order data sent in request. |

## *Example of SOAP message*

Request:

```
POST /IdmOrders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2009/04/0
1">
      <ns:ISOTEDATA xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01"
id="1" message-code="801" date-time="2016-05-01T11:48:51" answer-required="1">
        <!-- order data -->
      </ns:ISOTEDATA>
    </ns:UploadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2016 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2016/04/0
1">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="802" ...>
            <!--success description -->
      </ns:RESPONSE>
       <ns:ISOTEDATA xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01"
message-code="803" ...>
            <!--order data description -->
       </ns:ISOTEDATA>
    </ns:UploadResponse>
  </s:Body>
</s:Envelope>
```

## *SOAP Modify*

*Modify* method of *IdmOrders* web service operates in synchronous mode, i.e. the request is handled by the response within the same call. Response to this request consists of order modification processing status and order modification data copy from the request.

*Description of request structure*

**Table 17** Request structure description – Method Modify

| UploadRequest | Description |
|---|---|
| ISOTEDATA-VDT | Structure of order is in accordance with specification, where message-code=804 (see description of E-06_02 data flow). |

*Description of response structure*

**Table 18** Response structure description – Method Modify

| UploadResponse | Description |
|---|---|
| RESPONSE-VDT | Common structure of retrieving processing status is in accordance with specification, where message-code=805 (see description of E-06_02 data flow). |
| ISOTEDATA-VDT | Structure of order is in accordance with specification, where message-code=806 (see description of E-06_02 data flow). Response includes modification description sent in the request. |

## *Example of SOAP messages*

Request:

```
POST /IdmOrders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2009/04/0
1">
      <ns:ISOTEDATA xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01"
id="1" message-code="804" date-time="2016-05-01T11:48:51" answer-required="1">
        <!— data of order modification-->
      </ns:ISOTEDATA>
    </ns:UploadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2016 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2016/04/0
1">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="805" ...>
            <!-- processing status description-->
      </ns:RESPONSE>
       <ns:ISOTEDATA xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01"
message-code="806" ...>
            <!-- order description -->
       </ns:ISOTEDATA>
    </ns:UploadResponse>
  </s:Body>
</s:Envelope>
```

## SOAP Download

*Download* method of *IdmOrders* web service operates in synchronous mode, i.e. the request is handled by the response within the same call. Response to request for order submission consists of request processing status and data of chosen order or all orders within a specified time interval.

*Description of request structure*

**Table 19** Request structure description – Method Download

| UploadRequest | Description |
|---|---|
| CDSREQ-VDT | Order structure is in accordance with specification, where message-code=807 (see description of E-06_03 data flow). |

*Description of response structure*

**Table 20** Response structure description - Method Download

| UploadResponse | Description |
|---|---|
| RESPONSE-VDT | Common structure of retrieving processing status is in accordance with specification, where message-code=808 (see description of E-06_03 data flow). |
| ISOTEDATA-VDT | Structure of order is in accordance with specification, where message-code=809 (see description of E-06_03 data flow). Returned data contains information about specific order or all orders in specified time interval depending on request formulation. |

### *Example of SOAP messages*

Request:

```
POST /IdmOrders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2009/04/0
1">
      <ns:CDSREQ xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01" id="1"
message-code="807" date-time="2016-05-01T11:48:51" answer-required="1">
        <!-- data from order modification -->
      </ns:CDSREQ>
    </ns:UploadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2016 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2016/04/0
1">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="808" ...>
            <!-- processing status description -->
      </ns:RESPONSE>
       <ns:ISOTEDATA xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01"
message-code="809" ...>
            <!-- description of order -->
       </ns:ISOTEDATA>
    </ns:UploadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.5 IdmOrderBook

Web service *IdmOrderBook* allows market participant to access immediate data from order book (available quantities and prices) on intraday market.

Service implements the following method:

- *Download* – method for data acquisition of order book on intraday market,
- *DownloadH2HMatrix* – method for downloading of available cross-border capacities in the form of Hub-to-Hub matrix.

#### *SOAP Download*

*Download* method of *IdmOrderBook* web service operates in synchronous mode, i.e. the request is handled by the response within the same call.

**Note for AMQP interfaces:**

System of the market participant, which is concurrently connected to AMQP interface for intraday notifications, must insure that the processing of queues during *Download* method call will be paused. In other case order book update may be incorrectly processed, as during *Download* method call multiple changes can occur in order book, which won‘t be included in the results.

*Request structure description*

**Table 21** Request structure description - Method Download

| DownloadRequest | Description |
|---|---|
| CDSREQ-VDT | Structure of request for data acquisition of intraday order book is in accordance with specification, where message-code=810 (see description of E-08_01 data flow). |

**Table 22** Request structure description - Method DownloadH2HMatrix

| DownloadH2HMatrixRequest | Opis |
|---|---|
| CDSREQ-VDT | Structure of request for data acquisition of intraday cross-border capacities within cross-border intraday marketis in accordance with specification, where message-code=840 (see description of E-08_02 data flow). |

*Response structure description*

**Table 23** Response structure description - Method Download

| DownloadResponse | Description |
|---|---|
| RESPONSE-VDT | Common structure of retrieving processing status is in accordance with specification, where message-code=811 (see description of E-08_01 data flow). |
| ISOTEDATA-VDT | Structure for accessing data from order book is in accordance with specification, where message-code=812 (see description of E-08_01 data flow). |

## *SOAP message example*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/services/2009/0
4/01">
      <ns:CDSREQ
          xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01"
          id="1" message-code="810" date-time="2017-05-01T11:48:51" >
          <!-- message data for order book retrieval-->
      </ns:CDSREQ>
    </ns:DownloadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2016 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmorders/services/2016/04/0
1">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="811" ...>
            <!-- status of request processing -->
      </ns:RESPONSE>
       <ns:ISOTEDATA xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01"
message-code="812" ...>
            <!-- data from order book -->
       </ns:ISOTEDATA>
    </ns:UploadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.6 IdaOrders

Web service IdaOrders provides market participants with automated interface for submission and reception of own orders on intraday auctions.

Web service allows following methods:

- *Upload* – method for submission orders for intraday auction,
- *Download* – method for retrieval of own orders submitted on intraday auction.

#### *SOAP Upload*

Method *Upload* of the web service *IdaOrders* works in synchronous regime, i.e. the request is handled by the response within the same call.

*Description of request structure*

**Table 24** Request structure description –Upload method

| UploadRequest | Opis |
|---|---|
| ISOTEDATA | Štruktúra objednávky/pokynu podľa špecifikácie, pričom message-code=851 (pozri Opis dátového toku E-09_01). |

*Description of response structure*

**Table 25** Response structure description – Upload method

| UploadResponse | Opis |
|---|---|
| RESPONSE | Common return structure of request handling effectivity is in accordance with specification where message-code=812 (see description of data flow E-09_01). |
| ISOTEDATA | Structure of order is in accordance with specification where message-code=813 (see description data flow E-09_01). Description of order is returned as it was registered in the system. |

*Example of SOAP message*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01" id="1"
message-code="851" date-time="2009-05-01T11:48:51" answer-required="1">
        <!-- order data -->
      </ns:ISOTEDATA>
    </ns:UploadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:UploadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="812" ...>
        <!-- efficiency description -->
      </ns:RESPONSE>
      <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01" message-
code="813" ...>
        <!-- order data description -->
      </ns:ISOTEDATA>
    </ns:UploadResponse>
  </s:Body>
</s:Envelope>
```

## *SOAP Download*

*Download* method of *Orders* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Description of request structure*

**Table 26** Request structure description – Download method

| DownloadRequest | Description |
|---|---|
| CDSREQ | Request structure for reception of own order in accordance with specification, where message-code=831 (see description of E-09_02 data flow). |

*Description of response structure*

**Table 27** Response structure description – Download method

| DownloadResponse | Description |
|---|---|
| RESPONSE | Common return structure of request handling effectivity in accordance with specification, where message-code=832 (see description of E-09_02 data flow). |
| ISOTEDATA | Structure of order in accordance with specification, where message-code=833 (see description of E-09_02 data flow). Description of order is returned as it was registered in the system. |

## *SOAP message examples*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:CDSREQ
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01" id="1"
message-code="831" date-time="2009-05-01T11:48:51" >
        <Trade trade-day="2023-12-01" auction-id="IDA1" />
      </ns:CDSREQ>
    </ns:DownloadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="832" ... >
            <!-- efficiency description -->
      </ns:RESPONSE>
       <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01" message-
code="833" ... >
            <!-- order data description -->
       </ns:ISOTEDATA>
    </ns:DownloadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.7 Evaluations

Evaluations web service allows market participants to use automated interface for results/evaluations retrieval of day-ahead market.

The service implements the following methods:

- Download – method for retrieval of own results/evaluations of day-ahead market.

#### *SOAP Download*

Download method of *Evaluations* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Request structure description*

**Table 28** Request structure description – Download method

| DownloadRequest | Description |
|---|---|
| CDSREQ | Request structure for retrieval of own order is in accordance with specification, where message-code=941, 951 or 961 (see description of E-03_02, E-05_01, E-05_02 data flows). |

*Response structure description*

**Table 29** Response structure description – Download method

| DownloadResponse | Description |
|---|---|
| RESPONSE | Common return structure of retrieving processing status is in accordance with specification, whereas message-code=942, 952 or 962 (see description of E-03_02, E-05_01, E-05_02 data flows). |
| ISOTEDATA | Structure of results/evaluations is in accordance with specification where message-code=943, 953 or 963 (see description of E-03_02, E-05_01, E-05_02 data flows). |

## *Example of SOAP message*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/services/2009/0
4/01">
      <ns:CDSREQ
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01" id="1"
message-code="941" date-time="2009-05-01T11:48:51" >
        <Trade trade-day="2009-05-01" />
      </ns:CDSREQ>
    </ns:DownloadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="942" ... >
            <!-- description of processing status -->
      </ns:RESPONSE>
       <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01"
message-code="943" ... >
            <!-- description of DM results -->
       </ns:ISOTEDATA>
    </ns:DownloadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.8 IdmEvaluations

*IdmEvaluations* web service allows market participants to use automated interface for results/evaluations retrieval of intraday market.

The service implements the following methods:

- Download – method for retrieval of own results/evaluations on intraday market.

#### *SOAP Download*

*Download* method of *IdmEvaluations* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Request structure description*

**Table 30** Request structure description – Download method

| DownloadRequest | Description |
|---|---|
| CDSREQ-VDT | Request structure description for IDM results for market participant is in accordance with specification, where message-code=961 (see data flow description E-07_01) for daily results, message-code=571 (see data flow description E-07_02) for monthly results message-code=951 (see data flow description E-07_03) for results per periods. |

*Response structure description*

**Table 31** Response structure description – Download method

| DownloadResponse | Description |
|---|---|
| RESPONSE-VDT | Common return structure of retrieving processing status is in accordance with specification, whereas message-code=962,572 or 952 (see description of E-07_01, E-07_02, E-07_03 data flows). |
| ISOTEDATA-VDT | Structure of results/evaluations is in accordance with specification where message-code=963,573 or 953 (see description of E-07_021, E-07_02, E-07_03 data flows). |

## *SOAP message example*

Request:

```
POST /IdmOrders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmevaluations/services/201
6/04/01">
      <ns:CDSREQ
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01" id="1"
message-code="961" date-time="2016-05-01T11:48:51" >
        <Trade trade-day="2016-05-01″ />
      </ns:CDSREQ>
    </ns:DownloadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2016 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmevaluations/services/2016
/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="962" ... >
            <!—processing status -->
      </ns:RESPONSE>
       <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/idmevaluations/types/2009/04/01
" message-code="963" ... >
            <!—IDM results description -->
       </ns:ISOTEDATA>
    </ns:DownloadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.9 IdaEvaluations

IdaEvaluations web service allows market participants to use automated interface for results/evaluations retrieval of intraday auctions.

The service implements the following methods:

- Download – method for retrieval of own results/evaluations of intraday auctions.

#### *SOAP Download*

Download method of Ida*Evaluations* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Request structure description*

**Table 32** Request structure – Download method

| DownloadRequest | Opis |
|---|---|
| CDSREQ | Request structure for retrieval of own order is in accordance with specification, where message-code=941, 951 or 961 (see description of E-11_01, E-11_02, E-11_03 data flows). |

*Opis štruktúry odpovede*

**Table 33** Response structure - Download method

| DownloadResponse | Opis |
|---|---|
| RESPONSE | Common return structure of retrieving processing status is in accordance with specification, whereas message-code=942, 952 or 962 (see description of E-11_01, E-11_02, E-11_03 data flows). |
| ISOTEDATA | Structure of results/evaluations is in accordance with specification where message-code=943, 953 or 963 (see description of E-11_01, E-11_02, E-11_03 data flows). |

*SOAP message examples*

Request:

```
POST /Orders.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/services/2009/04/01"
>
      <ns:CDSREQ
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01" id="1"
message-code="941" date-time="2023-12-01T11:48:51" >
         <Trade trade-day="2023-12-01" auction-id="IDA1"/>
      </ns:CDSREQ>
    </ns:DownloadRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/services/2009/04/01">
      <ns:RESPONSE xmlns="http://sfera.sk/ws/xmtrade/isot/ut/types/2009/04/01"
message-code="942" ... >
            <!-- efficiency description -->
      </ns:RESPONSE>
       <ns:ISOTEDATA
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01" message-
code="943" ... >
            <!-- IDA results description -->
       </ns:ISOTEDATA>
    </ns:DownloadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.10 StatusRequest

*StatusRequest* web service allows market participants to use automated interface for status/information retrieval about processes within *ENTSO-E* specification. Currently it provides an option to retrieve MCC values.

The service implements the following methods:

- DownloadMCC – method for MCC values retrieval for a given trading day.

#### *SOAP DownloadMCC*

*DownloadMCC* method of *StatusRequest* service operates in synchronous mode, i.e. the request is handled by the response within the same call.

*Request structure description*

**Table 34** Request structure description – DownloadMCC method

| DownloadMCCRequest | Description |
|---|---|
| StatusRequest | Request structure for retrieval of MCC values is in accordance with ESR.StatusRequest specification. |

*Response structure description*

**Table 35** Response structure description – DownloadMCC method

| DownloadMCCResponse | Description |
|---|---|
| Acknowledgement | Common return structure of for retrieving processing status is in accordance with EAD.Acknowledgement specification. |
| CapacityDocument | Structure is in accordance with ECAN.CapacityDocument specification. |

## *Example of SOAP message*

Request:

```
POST /StatusRequest.WCF.Host/ServiceReference.svc HTTP/1.1
Content-Type:application/soap+xml; charset=utf-8
Host: ...
Content-Length: ...
Expect: 100-continue
Connection: Keep-Alive

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadMCCRequest
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/statusrequest/services/2009/
04/01">
      <ns:StatusRequest DtdVersion="1" DtdRelease="1"
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/esrv1r1/2009/04/01">
       <!--  Request description -->
      </ns:StatusRequest>
    </ns:DownloadMCCRequest>
  </s:Body>
</s:Envelope>
```

Response:

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Sun, 30 Nov 2008 16:58:25 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/soap+xml; charset=utf-8
Content-Length: ...
Connection: Close

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd">
  <s:Header>
    <!-- WS-Addressing -->
    <!-- WS-Security -->
  </s:Header>
  <s:Body u:Id="_1">
    <ns:DownloadMCCResponse
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/interfaces/statusrequest/services/2009/
04/01">
      <ns:Acknowledgement
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ackv5r0/2009/04/01">
            <!-- request processing status -->
      </ns:Acknowledgement >
       <ns:CapacityDocument
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01">
            <!-- MCC Description -->
       </ns:CapacityDocument >
    </ns:DownloadResponse>
  </s:Body>
</s:Envelope>
```

### 3.1.11 Communication Security

Web services are available exclusively through secured *https* protocol that allows encryption of transmitted messages. Therefore, messages at the SOAP protocol level are not encrypted.

Web service interfaces are secured in accordance with *WS-Security (WSS)* standard, version 1.0, pursuant to which the following techniques of security are designed:

- Electronic signature of sent SOAP requests and responses,
- Transmission of authentication details within SOAP request (username/password, certificate).

#### *Electronic signature*

Support for electronic signature of SOAP messages is secured within implementation of *WS-Security* standard, version 1.0.

(http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss).

Signature is stored in the header of SOAP message, i.e. separately from the message body transmitting data. *WSS* standard implements the signature on *xmldsig* standard (http://www.w3.org/TR/xmldsig-core).

Following signed elements are required:

- message body (s:Body),
- username/password token of a user (o:UsernameToken),
- timestamp (u:Timestamp),
- specification of method name of web service (a:Action),
- sender specification (a:ReplyTo),
- message identification (a:MessageID),

- specification of service target address (a:To).

***Example of SOAP message***

The following example demonstrates the message structure consisting of soap message elements (envelope), header (header), address specification header elements and security and message body.

Beginning

```
<s:Envelope
  xmlns:s="http://www.w3.org/2003/05/soap-envelope"
  xmlns:a="http://schemas.xmlsoap.org/ws/2004/08/addressing"
  xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
<s:Header>
```

WS-Addressing

```
<a:Action s:mustUnderstand="1" u:Id="id-17567474" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">http://sfera.sk/ws/xmtrade/isot/interfaces/NameOfService/services/2009/04/01/NameOfS
ervice Contract/NameOfMethod</a:Action>
<a:ReplyTo s:mustUnderstand="1" u:Id="id-235207" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <a:Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</a:Address>
</a:ReplyTo>
<a:MessageID s:mustUnderstand="1" u:Id="id-11090325" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">b83ac27b-9a4f-40e3-a782-
96df2cbea73e</a:MessageID>
<a:To s:mustUnderstand="1" u:Id="id-27256294" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">ServiceAddress</a:To>
```

WS-Security

```
<o:Security xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd">
  <o:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
s-message-security-1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-x509-token-profile-1.0#X509v3" u:Id="CertId-17206535" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"><!-- REMOVED --
></o:BinarySecurityToken>
  <d:Signature Id="Signature-190585" xmlns:d="http://www.w3.org/2000/09/xmldig#">
    <d:SignedInfo>
      <d:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
      <d:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldig#rsa-sha1"/>
      <d:Reference URI="#UsernameToken-13236543">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>lm0E+rpDJ8oSP8Fh+ZlqZRiMjc8=</d:DigestValue>
      </d:Reference>
      <d:Reference URI="#Timestamp-2175170">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>02CsUF1As77a6I3+BkQZ22TogWI=</d:DigestValue>
      </d:Reference>
      <d:Reference URI="#id-4652787">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>ktXRJoijcGSFrHaUKaLXUnH43XU=</d:DigestValue>
      </d:Reference>
      <d:Reference URI="#id-17567474">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>lLOeuXRDI1Gs5IX+zvaWuFIhVzw=</d:DigestValue>
      </d:Reference>
      <d:Reference URI="#id-11090325">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>ZsiiDzGRLHuyb8bKASKDo8ryoqc=</d:DigestValue>
      </d:Reference>
      <d:Reference URI="#id-235207">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>9p44ZJinb/97IPlX0C7yFayRHpc=</d:DigestValue>
      </d:Reference>
      <d:Reference URI="#id-27256294">
        <d:Transforms><d:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
              </d:Transforms>
        <d:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldig#sha1"/>
        <d:DigestValue>BCxp9HRQ6cJAykEdliom9mU86vA=</d:DigestValue>
      </d:Reference>
    </d:SignedInfo>
    <d:SignatureValue><!-- REMOVED --></d:SignatureValue>
    <d:KeyInfo Id="KeyId-33119438">
      <o:SecurityTokenReference u:Id="STRId-28732159" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <o:Reference URI="#CertId-17206535" ValueType="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/>
      </o:SecurityTokenReference>
    </d:KeyInfo>
  </d:Signature>
  <o:UsernameToken u:Id="UsernameToken-13236543" xmlns:u="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
    <o:Username><!-- REMOVED --></o:Username>
    <o:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-
profile-1.0#PasswordText"><!-- REMOVED --></o:Password>
  </o:UsernameToken>
  <u:Timestamp u:Id="Timestamp-2175170" xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-utility-1.0.xsd">
    <u:Created>2009-07-01T09:24:06.011Z</u:Created>
    <u:Expires>2009-07-01T12:10:46.011Z</u:Expires>
  </u:Timestamp>
</o:Security>
```

Header ending + body + message ending

```
</s:Header>
  <s:Body u:Id="id-4652787">
    <!--  request body -->
  </s:Body>
</s:Envelope>
```

### *Authentication and authorization of web service call*

Web services are secured against unauthorized use. System user is required to have an assigned user account in XMtrade®/ISOT system with the client certificate for signing and verification of identity. User is required to have assigned rights for calling relevant web services.

## 3.1.12 Description of web services

Description of web services of the market organizer information system XMtrade®/ISOT is given in the form of WSDL (http://www.w3.org/TR/wsdl) documents on the following addresses.

### *Production environment*

**Table 36** Description of web services XMtrade®/ISOT – production environment

| ID | Web service name | Service address /WSDL document |
|---|---|---|
| OB-01 | Orders | https://isot.okte.sk/interfaces/Orders/Service.svc<br>https://isot.okte.sk/interfaces/Orders/Service.svc?wsdl |
| OB-02 | IdmOrders | https://isot.okte.sk/interfaces/IdmOrders/Service.svc<br>https://isot.okte.sk/interfaces/IdmOrders/Service.svc?wsdl |
| OB-03 | IdmOrderBook | https://isot.okte.sk/interfaces/IdmOrderBook/Service.svc<br>https://isot.okte.sk/interfaces/IdmOrderBook/Service.svc?wsdl |
| OB-04 | IdaOrders | https://isot.okte.sk/interfaces/IdaOrders/Service.svc<br>https://isot.okte.sk/interfaces/IdaOrders/Service.svc?wsdl |
| EV-01 | Evaluations | https://isot.okte.sk/interfaces/Evaluations/Service.svc<br>https://isot.okte.sk/interfaces/Evaluations/Service.svc?wsdl |
| EV-02 | IdmEvaluations | https://isot.okte.sk/interfaces/IdmEvaluations/Service.svc<br>https://isot.okte.sk/interfaces/IdmEvaluations/Service.svc?wsdl |
| EV-03 | IdaEvaluations | https://isot.okte.sk/interfaces/IdaEvaluations/Service.svc<br>https://isot.okte.sk/interfaces/IdaEvaluations/Service.svc?wsdl |
| SR-01 | StatusRequest | https://isot.okte.sk/interfaces/StatusRequest/Service.svc<br>https://isot.okte.sk/interfaces/StatusRequest/Service.svc?wsdl |

### *SANDBOX environment*

**Table 37** Description of web services XMtrade®/ISOT – SANDBOX environment

| ID | Web service name | Service address /WSDL document |
|---|---|---|
| OB-01 | Orders | https://sandbox-isot.okte.sk/interfaces/Orders/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/Orders/Service.svc?wsdl |
| OB-02 | IdmOrders | https://sandbox-isot.okte.sk/interfaces/IdmOrders/Service.svc |

| ID | Web service name | Service address /WSDL document |
|---|---|---|
| | | https://sandbox-isot.okte.sk/interfaces/IdmOrders/Service.svc?wsdl |
| OB-03 | IdmOrderBook | https://sandbox-isot.okte.sk/interfaces/IdmOrderBook/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/IdmOrderBook/Service.svc?wsdl |
| OB-04 | IdaOrders | https://sandbox-isot.okte.sk/interfaces/IdaOrders/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/IdaOrders/Service.svc?wsdl |
| EV-01 | Evaluations | https://sandbox-isot.okte.sk/interfaces/Evaluations/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/Evaluations/Service.svc?wsdl |
| EV-02 | IdmEvaluations | https://sandbox-isot.okte.sk/interfaces/IdmEvaluations/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/IdmEvaluations/Service.svc?wsdl |
| EV-03 | IdaEvaluations | https://sandbox-isot.okte.sk/interfaces/IdaEvaluations/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/IdaEvaluations/Service.svc?wsdl |
| SR-01 | StatusRequest | https://sandbox-isot.okte.sk/interfaces/StatusRequest/Service.svc<br>https://sandbox-isot.okte.sk/interfaces/StatusRequest/Service.svc?wsdl |

## *Testing environment*

**Table 38** Description of web services XMtrade®/ISOT – testing environment

| ID | Web service name | Service address /WSDL document |
|---|---|---|
| OB-01 | Orders | https://test-isot.okte.sk/interfaces/Orders/Service.svc<br>https://test-isot.okte.sk/interfaces/Orders/Service.svc?wsdl |
| OB-02 | IdmOrders | https://test-isot.okte.sk/interfaces/IdmOrders/Service.svc<br>https://test-isot.okte.sk/interfaces/IdmOrders/Service.svc?wsdl |
| OB-03 | IdmOrderBook | https://test-isot.okte.sk/interfaces/IdmOrderBook/Service.svc<br>https://test-isot.okte.sk/interfaces/IdmOrderBook/Service.svc?wsdl |
| OB-04 | IdaOrders | https://test-isot.okte.sk/interfaces/IdaOrders/Service.svc<br>https://test-isot.okte.sk/interfaces/IdaOrders/Service.svc?wsdl |
| EV-01 | Evaluations | https://test-isot.okte.sk/interfaces/Evaluations/Service.svc<br>https://test-isot.okte.sk/interfaces/Evaluations/Service.svc?wsdl |
| EV-02 | IdmEvaluations | https://test-isot.okte.sk/interfaces/IdmEvaluations/Service.svc<br>https://test-isot.okte.sk/interfaces/IdmEvaluations/Service.svc?wsdl |
| EV-03 | IdaEvaluations | https://test-isot.okte.sk/interfaces/IdaEvaluations/Service.svc<br>https://test-isot.okte.sk/interfaces/IdaEvaluations/Service.svc?wsdl |
| SR-01 | StatusRequest | https://test-isot.okte.sk/interfaces/StatusRequest/Service.svc<br>https://test- |

| ID | Web service name | Service address /WSDL document |
|---|---|---|
| | | isot.okte.sk/interfaces/StatusRequest/Service.svc?wsdl |

Addresses of testing environment services are almost identical to production addresses. They only differ in the domain name of the address: https://sandbox-isot.okte.sk, https://test-isot.okte.sk respectively instead of https://isot.okte.sk.

## 3.2 AMQP interface

AMQP interface is communication extension within intraday market. This interface improves possibilities of web services and allows full trading automation on intraday market through external interfaces.

AMPQ interface allows market participants to receive notifications about their own orders and public changes in order book:

- Successful order submission and own order modification (change of status),
- Status change of order book (increase/decrease of available quantity).

AMQP interface is built on open source product RabbitMQ that implements AMQP protocol.

### 3.2.1 AMQP Protocol

AMQP (Advanced Message Queuing Protocol) is network protocol for high performance and reliable communication based on message exchange. AMQP is open standard for so called middleware messaging layers.

Versions of AMQP components:

- AMQP protocol: 0-9-1,
- RabbitMQ server: 3.6.x.

AMQP 0-9-1 uses concept of intermediary communication through so called brokers. Brokers receive messages from sender (message producer) and sends/routes the messages to consumer.

Technically, there are following elements in the protocol:

- Exchanges: input point/container to which sender sends messages.
- Queues: queue/destination address, on which broker delivers message based on basic rules (bindings).
- Bindings: rules for message delivering.

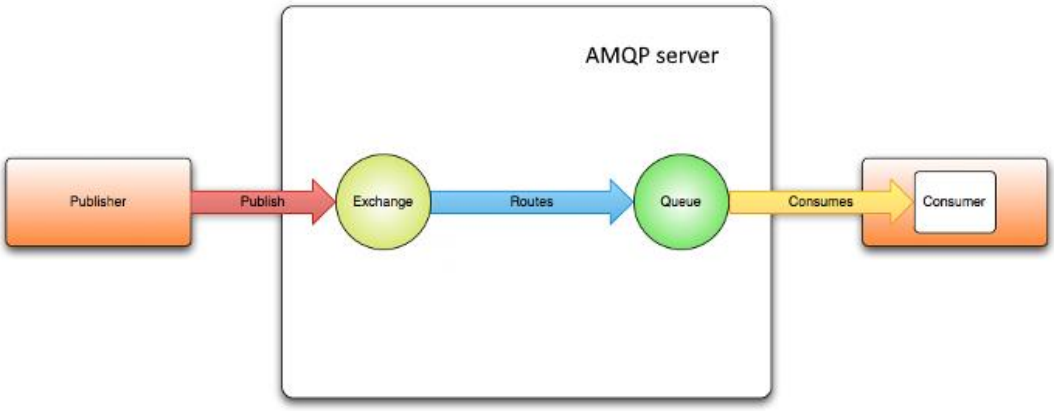


**Figure 9** AMQP communication concept

Communication works on channel principle within one connection (TCP connection).

Basic description of AMQP communication principles and its elements can be found here: https://www.rabbitmq.com/tutorials/amqp-concepts.html.

### 3.2.2 Connecting to RabbitMQ server

RabbitMQ on http://www.rabbitmq.com/clients.html provides complex documentation for technical realization of connection and communication with it on various development platforms (Java, .NET). For these platforms, it provides complete libraries for clients.

Prerequisites for successful connection:

- only one connection per user account intended for AMQP communication,
- name of user account does not contain white symbols (space) or '.' (dot) sign,
- login certificate must be assigned to user account. This certificate must be issued with client authentication function and from supported certification authority.

For reliable long-term connection with AMQP server, it is recommended to create AMQP connection with heartbeat setting to minimum 5-20 seconds. Technical details can be found here: https://www.rabbitmq.com/heartbeats.html.

### 3.2.3 Communication scenarios

XMtrade®/ISOT system has several supported communication scenarios:

- *Broadcast* communication where system XMtrade®/ISOT sends notifications that are public and available to all market participants or private which are addressed only to particular market-participant. This type of connection is initiated by XMtrade®/ISOT system, and participants can then register for message subscription, which are interesting for them.

#### *Broadcast*

System XMtrade®/ISOT sends following types of information in the form of broadcast messages:

- Notifications about changes on intraday market.

Broadcast messages are distributed from system XMtrade®/ISOT through AMQP server to all connected users that are subscribed for given message type.

##### *Notifications about changes on intraday market*

Within intraday market, system provides notifications about changes related to own orders or public changes in order book, which occurred as a result of changes invoked by market participant (order submission etc.) or indirectly as a result of trade creation etc.

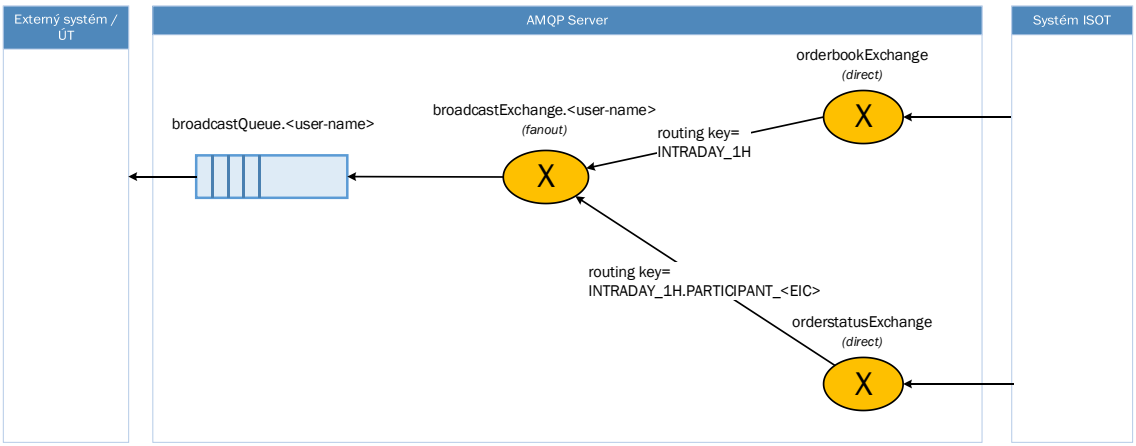


**Figure 10** Scheme of notification messages flow on intraday market

**Table 39** Data flow distribution keys tags description

| Distribution keys | Description |
|---|---|
| INTRADAY_1H | Information of public nature related to 60-min products:<br>- change in order book data (E-10_02). |
| INTRADAY_15M | Information of public nature related to 60-min products:<br>- change in order book data (E-10_02) |
| INTRADAY_1H.PARTICIPANT_<EIC> | Information relevant for particular market participant, where <EIC> is EIC code of a market participant:<br>- Change of status/creation of own order (E-10_01). |
| INTRADAY_H2H | Information of public nature:<br>- Change of cross-border capacities in form of Hub-to-Hub (H2H) matrix (E-10_03) |

*Receiving notifications*

For notification reception client (consumer) has to connect (subscribe) to private queue called *broadcastQueue.<user-name>*, where *<user-name>* is user name given after account registration. Queue on AMQP server is automatically created after account registration and access approval. Internally connection (binding) to system exchange objects with distribution keys corresponding to internal model for supported message routing is created.

User can register (subscribe) to message reception only from his private queue. Attempt to connect to a queue of a different user will fail.

All types of notifications that the user has authorization to will be distributed into private queue.

Broadcast queues are set up as permanent which means messages distributed into this queue will not be lost with client disconnect or short outage of XMtrade®/ISOT system. However, these broadcast messages have limited life span (for example 60 seconds) on grounds of AMQP server security. This precaution protects AMQP server from system resource depletion.

*Failure cases*

In case of AMQP server shutdown, all connections to broadcast queue are lost. However, if client is subscribed for so-called shutdown notification, AMQP server can respond to this situation. After the AMQP server is up, client has to create new connection and register private queues again.

In case of XMtrade®/ISOT system outage or malfunction, connection or registration to private queues is not lost but no new notifications will be added until XMtrade®/ISOT system is up and running.

In case of client outage (client is not connected to private queue) messages sent to this queue will be lost if the client will not connect for message consuming before maximum life span of broadcast messages is reached.

### 3.2.4 Communication security

Communication is cyphered on network layer with TLS protocol and authentication with client certificate while only TLS 1.2 and higher is accepted.

On https://www.rabbitmq.com/ssl.html web page, it is possible to find instructions for connecting with the required security prerequisites for example, Configuring TLS Version in Java Client chapters or more precisely Configuring the .NET Client and Presenting and validating certificates.

### *Authentication*

For authentication, it is necessary to create connection on AMQP server with the following login credentials:

- Name/password.
- Client certificate for authentication on TLS protocol layer where mutual validation of client and server certificates occurs at which moment key and cryptographic algorithm for further communication is arranged.

AMQP server identifies user based on user name while verifies credentials mentioned above (checks password and certificate validity of the given user).

### *Authorization*

Authorization works on two levels:

- AMQP server, which verifies users access to AMQP entities (queue, exchange, binding).
- System XMtrade®/ISOT, which sends messages only to authorized users (market participants).

## 3.2.5 Message format

All transferred AMQP messages contain UTF-8 coded data (payload) encapsulated in XML format and metadata on AMQP layer properties.

### *AMQP message properties*

Every message has the following properties filled in metadata.

**Table 40** Message properties in metadata description

| AMQP Message Property | Description |
|---|---|
| content-type | Indicates the type of XML data and its version encapsulated in AMQP.<br>These are the valid values:<br>- x-isot-vdt/order-status,<br>- x-isot-vdt/orderbook-status. |
| correlation-id | Serves for correlation on AMQP message layer.<br>Allows correlation of requests filled through web services with broadcast messages for example activation of order with notification about order activation. In this case, the *correlation-id* is same as id attribute of the request sent in ISOTEDATA-VDT structure (see chapter 4.3.3).<br>Correlation is also possible on XML data level through id attribute of reference structure element RESPONSE-VDT and ISOTEDATA-VDT. |

## 3.2.6 AMQP server

Connection to AMQP server is established through AMQP specific URI with accordance to specification https://www.rabbitmq.com/uri-spec.html.

### *Production environment*

**URI parameters:**

| | |
|---|---|
| **h**ostname | isot.okte.sk |
| **p**ort | 5671 (TLS) |
| **v**host | (empty) |

**Example URI on AMQP server:**

amqp://userName:password@isot.okte.sk:5671/

**TLS parameters:**

Server certificate**:** *.okte.sk

## *SANDBOX environment*

**URI parameters:**

| | |
|---|---|
| **h**ostname | sandbox-isot.okte.sk |
| **p**ort | 5671 (TLS) |
| **v**host | (empty) |

**Example URI on AMQP server:**

amqp://userName:password@sandbox-isot.okte.sk:5671/

**TLS parameters:**

Server certificate**:** *.okte.sk

## *Test environment*

**URI parameters:**

| | |
|---|---|
| **h**ostname | test-isot.okte.sk |
| **p**ort | 5671 (TLS) |
| **v**host | (empty) |

**Example URI on AMQP server:**

amqp://userName:password@test-isot.okte.sk:5671/

**TLS parameters:**

Server certificate**:** *.okte.sk

## 3.3 WEB API interface

The communication interface serves as an extension for interacting with the intraday market. This interface provides communication capabilities in JSON format, enhancing the automation of trading on the intraday market via external interfaces.

The Web API interface enables market participants to:

- Successfully creation their own orders and modify existing orders (status updates).
- Access to own orders, trades, and H2H intraday market data (VDT).

The market operator's information system, XMtrade®/ISOT, supports external interfaces through the following set of APIs.

**Table 41** Overview of WEB API IDM

| Name of WEB API | Method | Url | Description |
|---|---|---|---|
| IDM orders - create | POST | https://{hostname}/api/v1/idm/orders | Market participants are provided with automated interface for submission orders intraday market. |
| IDM orders - list | GET | https://{hostname}/api/v1/idm/orders/?{query} | Market participants are provided with automated interface for reception of own orders on intraday market. |
| IDM orders - order detail | GET | https://{hostname}/api/v1/idm/orders/{orderid} | Market participants are provided with automated interface for reception of own orders on intraday market |
| IDM orders - activate | POST | https://{hostname}/api/v1/idm/orders/{orderid}/activate | Provides market participants with an automated interface for activating an order by its ID. |
| IDM orders - deactivate | POST | https://{hostname}/api/v1/idm/orders/{orderid}/deactivate | Provides market participants with an automated interface for deactivating an order by its ID. |
| IDM orders - cancel | POST | https://{hostname}/api/v1/idm/orders/{orderid}/cancel | Provides market participants with an automated interface for canceling an order by its ID. |
| IDM orders - trades of order | GET | https://{hostname}/api/v1/idm/orders/{orderid}/trades | Provides market participants with an automated interface to retrieve a list of trades for an order by its ID. |
| IDM hut-to-hub | GET | https://{hostname}/api/v1/idm/hub-to-hub?{query} | Provides market participants with an automated interface to retrieve available cross-border capacities for the intraday market. |

| Name of WEB API | Method | Url | Description |
|---|---|---|---|
| IDM market status | GET | https://{hostname}/api/v1/idm/market-status | Provides market participants with an automated interface to retrieve the current connection status of the intraday market. |

**Production environment**

| | |
|---|---|
| hostname | isot.okte.sk |
| server certificate | *.okte.sk |

**SANDBOX environment**

| | |
|---|---|
| hostname | sandbox-isot.okte.sk |
| server certificate | *.okte.sk |

**Test environment**

| | |
|---|---|
| hostname | test-isot.okte.sk |
| server certificate | *.okte.sk |

### 3.3.1 Communication scenarios

**Communication with the XMtrade®/ISOT system can occur either synchronously or asynchronously.**

Below is a list of successful status codes used in the WEB API along with their descriptions:

- **OK**: When the WEB API returns an *HTTP status code of **200***, it indicates that the request was successfully processed, and the result is included in the response.
- **Created**: When the WEB API returns an *HTTP status code of **201***, it indicates that the request was successful, and a new record has been created. The URL where the record is accessible is provided in the Response Header Location.
- **Accepted**: When the WEB API returns an *HTTP status code of **202***, it indicates that the request was successfully received but has not yet been processed.
- **No Content**: When the WEB API returns an *HTTP status code of **204***, it indicates that the request was successfully processed, but the response does not include a body.

**List of Unsuccessful Status Codes Used in the WEB API with Descriptions:**

- **Bad Request**: If the WEB API returns an *HTTP status code **400***, it means the request is invalid, and the error originates from the client side.
- **Unauthorized**: If the WEB API returns an *HTTP status code **401***, it means the client failed to authenticate correctly.
- **Forbidden**: If the WEB API returns an *HTTP status code **403***, it means the client does not have the right to access the requested resource.
- **Not Found**: If the WEB API returns an *HTTP status code **404***, it means the resource does not exist or is unavailable.
- **Server Error**: If the WEB API returns an *HTTP status code **500***, it means an unexpected error occurred on the server side.
- **Too Many Requests**: If the WEB API returns an *HTTP status code **429***, it means the API limit (**X-RateLimit-Policy**) has been exceeded.

**Pagination headers**

Endpoints that support pagination include the following headers in the response:

- X-Page-Offset: The current offset used for the retrieved data.

- X-Page-Limit: The maximum number of records returned in the response.
- X-Page-HasNext: Indicates whether there are more pages available (true or false).

### 3.3.2 Communication security

Communication is cyphered on network layer with TLS protocol and authentication with client certificate while only TLS 1.2 and higher is accepted.

#### *Authentication*

Basic authentication is a simple and standardized mechanism for verifying users within the HTTP protocol. It is used to secure access to web or API resources. In this type of authentication, the username and password are encoded in Base64 format and sent as part of the HTTP header.

For user authentication, the following credentials are required:

- Username/Password
- Client Certificate: Used for verification at the TLS protocol level. This ensures mutual validation of the client and server certificates. During this process, a key and encryption algorithm are also agreed upon to secure all subsequent communication.

#### *X-RateLimit-Policy*

This mechanism controls the rate limit for requests via the automated interface. It defines a rule related to the number of requests a client can send to the server within a specified time window. This policy helps protect the server from excessive load (e.g., DDoS attacks) and ensures fair usage of the API across different clients. The client is informed about the policy and their current usage in the response headers:

- **X-RateLimit-Policy**: The current rate limit policy – the number of requests per time window in seconds (e.g., "50;w=10" means 50 requests every 10 seconds).
- **X-RateLimit-Limit**: The allowed number of requests within the time interval (e.g., 20 requests every 60 seconds).
- **X-RateLimit-Remaining**: The number of remaining requests the client can make before reaching the limit.
- **X-RateLimit-Reset**: The time in seconds when the limit will be reset.

### 3.3.3 Order creation

The WEB API interface *IdmOrders - create* provides market participants with an automated interface for submitting orders to the intraday market.

The interface implements the following methods:

- **POST** - an endpoint for submitting orders/requests.

#### *IDM orders – create (E06-01)*

Setting the URL for creating a request. The interface does not specify query parameters.

| Method | Basic url |
|---|---|
| POST | https://{hostname}/api/v1/idm/orders |

### *POST – Order creation*

The POST method in the API IDM orders – create operates in either synchronous or asynchronous mode, meaning the request is processed with one of the responses from the list of success status codes.

*Description of request structure*

**Table 42** Request structure description

| POST request | Description |
|---|---|
| JSON | Request structure (see description of E-06_01 data flow). |

Creating a new order is performed using the JSON format. The structure of the request is as follows:

ROOT Object

**Table 43** Root Object structure

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | String | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12-01). | Optional |
| groupIndication | rejectPartially / rejectAll | **rejectPartially**: In case of a single invalid order, continue processing the remaining ones.<br>**rejectAll**: In case of a single invalid order, stop processing the rest. | Optional |
| orders | Field (array) | One or more orders | Required |

*Orders*

**Table 44** Orders structure

| Field | Value | Description | Use |
|---|---|---|---|
| direction | buy/sell | Class of order:<br>• N - buy,<br>• P - sell. | Required |
| indication | noIndication/fok/ioc/aon | Identification of order:<br>• N - no limitations,<br>• fok – Fill Or Kill,<br>• ioc – Immediate or Cancel,<br>• aon – All or None – only for user-defined block orders. | Required |
| deliveryStart | YYYY-MM-DDTHH:mm:SSZ | It defines the start of the period for which the order is placed, in UTC. | Required |
| deliveryEnd | YYYY-MM-DDTHH:mm:SSZ | It defines the end of the period for which the order is placed, in UTC. | Required |
| expiration | YYYY-MM-DDTHH:mm:SSZ | The date and time of the order's expiration set in UTC. If not specified, the order will expire at the close of the trading period. | Optional |
| quantity | Decimal number | Quantity [MW] with one decimal position precision. Separator of decimal position is “.“ (dot). | Required |
| price | desatinné číslo | Price[EUR/MWh] with two decimal position precision. Separator of decimal position is “.“ (dot). | Required |
| active | true/false | Active/Inactive | Required |
| note | string | Commentary for order. | Optional |
| type | simple/block | Type order:<br>• simple<br>• block | Required |
| clientOrderId | string | Custom client identificator of order. | Optional |

### 3.3.4 Order modification

The WEB API interface IdmOrders provides market participants with an automated interface for modifying orders (status change).
The interface implements the following methods:

- POST - endpoint for activating an order,
- POST - endpoint for deactivating an order,
- POST - endpoint for canceling an order.

## *IDM orders – activate (E06-02)*

The IDM Orders Web API provides market participants with an automated interface for activating their own orders on the intraday market. The interface does not specify query parameters.

| Method | Basic url |
|---|---|
| POST | https://{hostname}/api/v1/idm/orders/{orderid}/activate |

### *POST - Order modification (activation)*

The POST method in the IDM Orders API – activate operates in synchronous or asynchronous mode, meaning the request is processed with one of the success status codes from the list of possible responses.

*Description of request structure*

**Table 45** Request structure description

| POST request | Description |
|---|---|
| JSON | Request structure (see description of E-06_02 data flow). |

The change of the order status is carried out through the JSON format. The body structure is as follows.

RootObject

**Table 46** Root object structure

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | string | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12- | Optional |

| Field | Value | Description | Use |
|---|---|---|---|
| | | 01). | |

## *IDM orders – deactivate (E06-02)*

The IDM Orders Web API provides market participants with an automated interface for deactivating their own orders on the intraday market. The interface does not specify query parameters.

| Method | Basic url |
|---|---|
| POST | https://{hostname}/api/v1/idm/orders/{orderid}/deactivate |

### *POST - Order modification (deactivation)*

The POST method in the IDM Orders API – deactivate operates in synchronous or asynchronous mode, meaning the request is processed with one of the success status codes from the list of possible responses.

*Description of request structure*

**Table 47** Request structure description

| POST request | Description |
|---|---|
| JSON | Request structure (see description of E-06_02 data flow). |

The change of the order status is carried out through the JSON format. The body structure is as follows.

RootObject

**Table 48** Root object structure

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | string | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12-01). | Optional |

## *IDM orders – cancel (E06-02)*

The IDM Orders Web API provides market participants with an automated interface for canceling their own orders on the intraday market. The interface does not specify query parameters.

| Method | Basic url |
|---|---|
| POST | https://{hostname}/api/v1/idm/orders/{orderid}/cancel |

*POST - Order modification (cancellation)*

The POST method in the IDM Orders API – cancel operates in synchronous or asynchronous mode, meaning the request is processed with one of the success status codes from the list of possible responses.

*Description of request structure*

**Table 49** Request structure description

| POST request | Desciption |
|---|---|
| JSON | Request structure (see description of E-06_02 data flow). |

The change of the order status is carried out through the JSON format. The body structure is as follows.

RootObject

**Table 50** Root object structure

| Pole | Hodnota | Opis | Použitie |
|---|---|---|---|
| correlationId | string | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12-01). | Optional |

## 3.3.5 Provision of orders

The WEB API interface IdmOrders - list provides market participants with an automated interface for accessing orders on the intraday market.
The interface implements the following methods:

- **GET** - endpoint for downloading data on the participant's own orders (list).
- **GET** - endpoint for downloading the details of a participant's own order.
- **GET** - endpoint for downloading the details of a trade related to the participant's own order.

*IDM orders – list (E06-03)*

The IDM Orders Web API provides market participants with an automated interface for accessing the list of their own orders. The interface is specified with query parameters.

*GET – List of orders*

| Method | Basic url |
|---|---|
| GET | https://{hostname}/api/v1/idm/orders/?{query} |

**Table 51** Query parameters for provison the list of orders

| Parameter | Value | Description | Use |
|---|---|---|---|
| status | active, inactive, canceled, partiallyMatched, matched, expired, expiredInactive | Order Status - comma-separated values. | Optional |
| product | 60/15 | Product Type (Period length in minutes)<br>60 – hourly product,<br>15 – quarter-hourly product. | Optional |
| createdFrom | YYYY-MM-DDTHH:mm:SSZ | Order created start time in UTC. | Optional |
| createdTo | YYYY-MM-DDTHH:mm:SSZ | Order created end time in UTC. | Optional |
| updatedFrom | YYYY-MM-DDTHH:mm:SSZ | Order update start time in UTC. | Optional |
| updatedTo | YYYY-MM-DDTHH:mm:SSZ | Order update end time in UTC. | Optional |
| deliveryFrom | YYYY-MM-DDTHH:mm:SSZ | Order delivery start time in UTC. | Optional |
| deliveryTo | YYYY-MM-DDTHH:mm:SSZ | Order delivery end time in UTC. | Optional |
| offset | Non-negative number | Pagination, for example, offset=0, starts from the first order. | Optional |
| limit | Non-negative number (max. 500) | The number of orders in the response. | Optional |
| expandTrades | true/false | Include trade details in the response. | Optional |

At least one parameter from the group createdFrom, updatedFrom, deliveryFrom is required.

The GET method for the API IDM orders – list operates in synchronous or asynchronous mode, meaning the request is processed with one of the success status codes from the list of responses.

*Description of request structure*

**Table 52** Description of the structure - order list request

| GET request | Description |
|---|---|
| JSON | Request structure (see description of E-06_03 data flow). |

*Description of respone structure*

*It is implemented in JSON format. The structure of the response is as follows*

RootObject

**Table 53** Root object structure

| Field | Value | Description | Use |
|---|---|---|---|
| id | Non-negative number | Order identificator in the system | Required |
| productType | Non-negative number | Product Type (Period length in minutes)<br>60 – hourly product,<br>15 – quarter-hourly product. | Required |
| deliveryStart | YYYY-MM-DDTHH:mm:SSZ | Order delivery start time in UTC. | Required |
| deliveryEnd | YYYY-MM-DDTHH:mm:SSZ | Order delivery end time in UTC. | Required |
| direction | buy/sell | Class of order:<br>• buy,<br>• sell. | Required |
| type | simple/block | Type order:<br>• simple<br>• block | Required |
| quantity | Decimal number | Quantity [MW] with one decimal position precision. Separator of decimal position is “.“ (dot). | Required |
| price | Decimal number | Price [Eur/MWh] with two decimal position precision. Separator of decimal position is “.“ (dot). | Required |
| status | active, inactive, canceled, partiallyMatched, matched, expired, expiredInactive | Order status in the system. | Required |
| isPending | true/false | Indicator of whether the order is still being processed by the system (e.g., waiting for | Required |

| Field | Value | Description | Use |
|---|---|---|---|
| | | activation). | |
| realizedQuantity | Decimal number | Realized quantity MW. | Required |
| realizedPriceWeighted | Decimal number | Realized price EUR/MWh. | Required |
| remainingQuantity | Decimal number | Remaining quantity MW. | Required |
| expiration | YYYY-MM-DDTHH:mm:SSZ / null | The date and time of the set expiration for the order in UTC. | Optional |
| createdAt | YYYY-MM-DDTHH:mm:SSZ | The time of the orders creation. | Required |
| updatedAt | YYYY-MM-DDTHH:mm:SSZ | The time of the orders latest update. | Required |
| createdBy | String | The name of the user who created the order. | Required |
| clientOrderId | String | Custom client identificator of order. | Optional |
| note | String | Commentary for order | Optional |

Trades

**Table 54** Trades structure

| Field | Value | Description | Use |
|---|---|---|---|
| id | Integer number | The trade identificator in the system. | Required |
| time | YYYY-MM-DDTHH:mm:SSZ | The time of the trade UTC. | Required |
| price | Decimal number | Price [Eur/MWh] with two decimal position precision. Separator of decimal position is “.“ (dot). | Optional |
| quantity | Decimal number | Quantity [MW] with one decimal position precision. Separator of decimal position is “.“ (dot). | Optional |

## *IDM orders – detail (E06-03)*

The IDM Orders Web API provides market participants with an automated interface for accessing the details of an order based on the order ID. The interface is not specified with query parameters.

### *HTTP GET- Detail order*

| Method | Basic url |
|---|---|
| GET | https://{hostname}/api/v1/idm/orders/{orderid} |

The GET method for the API IDM orders – detail operates in synchronous or asynchronous mode, meaning the request is processed with one of the success status codes from the list of responses.

*Description of request structure*

**Table 55** Description of the structure - order detail request

| GET request | Description |
|---|---|
| JSON | Request structure (see description of E-06_03 data flows). |

*Description of response structure*

It is implemented using the JSON format. The structure of the response is the same as specified above in Tables 53 and 54.

## *IDM orders trades-of-order (E06-03)*

The IDM Orders Web API provides market participants with an automated interface to access trade details based on the order ID. The interface is not specified with query parameters.

### *GET - Detail trades of order*

| Method | Basic url |
|---|---|
| GET | https://{hostname}/api/v1/idm/orders/{orderid}/trades |

The GET method in the API IDM orders – trades of order operates in either synchronous or asynchronous mode, meaning the request is processed with one of the success status codes from the list of response codes

*Description of request structure*

**Table 56** Description of the structure – order detail request

| GET request | Description |
|---|---|
| JSON | Request structure (see description of E-06_03 data flows). |

*Description of response structure*

It is implemented using the JSON format. The structure of the response is the same as specified above in Table 54.

### 3.3.6 IDM market-status

The IDM Orders web API provides market participants with an automated interface for accessing cross-border market status. The interface is not specified with a query parameter.

*HTTP GET - Detail market status*

*Description of request structure*

| Method | Basic url |
|---|---|
| GET | https://{hostname}/api/v1/idm/market-status |

*Description of response structure*

```
{
   "systemTime": "2024-11-19T22:00:50.3884144Z",
   "tradeDay": "2024-11-19",
   "tradingStatus": "xbidOk"
}
```

The response is provided in JSON format. The structure of the response contains the following main elements.

*RootObject:*

**Table 57** Root object structure

| Field | Value | Description | Use |
|---|---|---|---|
| systemTime | YYYY-MM-DDTHH:mm:ss.fffffffZ | System time UTC. | Required |
| tradeDay | YYYY-MM-DD | Current trade/delivery day | Required |
| tradingStatus | xbidOk, xbidNok, xbidHalt, xbidBatchMatching, localOn, localOff | Market trading status. | Required |

### 3.3.7 Hub-to-hub

*IDM hub-to-hub (E08-02)*

The IDM Orders Web API provides market participants with an automated interface to access data on available cross-border transmission capacities in the form of a Hub-to-Hub matrix for the intraday market. The interface is specified with query parameters.

*HTTP GET – Hub to hub snapshot*

| Metóda | Basic url |
|---|---|
| GET | https://{hostname}/api/v1/idm/hub-to-hub?{query} |

**Table 58** The query parameters for accessing cross-border transmission capacities.

| Parameter | Value | Description | Use |
|---|---|---|---|
| countryCodes | CZ, PL, HU... | Two-letter country codes separated by commas. | Optional |
| deliveryFrom | YYYY-MM-DDTHH:mm:SSZ | Start time of delivery range UTC. | Optional |
| deliveryTo | YYYY-MM-DDTHH:mm:SSZ | End time of delivery range UTC. | Optional |

The GET method in the IDM hub-to-hub API operates in synchronous mode, meaning the request is handled with one of the responses from the list of success status codes.

*Description of request structure*

**Table 59** Description of the structure – IDM hub to hub request

| GET request | Description |
|---|---|
| JSON | Request description (see description of E-08_02 data flow). |

*Description of reponse structure*

The response is provided in JSON format. The structure of the response contains the following main elements:

RootObject

**Table 60** Root object structure.

| Field | Value | Description | Use |
|---|---|---|---|
| eic | String | EIC of the Market Area. | Required |
| areaName | String | Name of the Market Area. | Required |
| countryCode | String | Two letter country code. | Required |
| deliveryDay | YYYY-MM-DD | Date of the delivery. | Required |
| deliveryStart | YYYY-MM-DDTHH:mm:SSZ | Delivery period start UTC | Required |
| deliveryEnd | YYYY-MM-DDTHH:mm:SSZ | Delivery period end UTC.. | Required |
| availableCapacityIn | Float number | Available import capacities in MW. | Required |
| availableCapacityOut | Float number | Available export capacities | Required |

| Field | Value | Description | Use |
|---|---|---|---|
| | | im MW. | |

## 3.4 WebSocket interface

The communication interface serves as an extension of communication with the intraday market. This interface enhances communication capabilities in JSON format, enabling full automation of intraday market trading through external interfaces.

The WebSocket interface enables real-time communication for market participants, allowing clients to:

- Create and modify their own orders.
- Retrieve the status of their orders and receive notifications about order changes.
- Access the orderbook status and receive notifications about updates (e.g., increased/decreased available quantities, changes in statistics).
- Retrieve the status of cross-border capacities and receive notifications about changes.
- Obtain the trading status and be notified of any updates.

### 3.4.1 Communication scenarios

The WebSocket connection is initiated as a standard HTTPS request using the wss:// protocol (encrypted via SSL/TLS). During this phase, the client requests the server to switch from HTTPS to the WebSocket protocol. Upon successfully establishing the connection, the server responds with the status code 101 Switching Protocols, confirming that the connection has been successfully switched to WebSocket.

Once the connection is established, a persistent, bidirectional communication channel is created, enabling both the client and the server to send messages at any time without needing to reopen the connection.

**Client disconnection by the server**:

If the client fails to process incoming updates via the WebSocket protocol in a timely manner, it will be disconnected. Upon disconnection, the client receives a message with the text:
"*Message queue too big. Try another change frequency."*

This message indicates that messages are accumulating in the pending queue because the client is unable to process incoming data in real-time. The solution is to adjust the frequency of received updates (orderbookfrequency) to ensure smooth processing and prevent system overload.

**Inconsistency or error on the Client Side**

- The client holds incorrect or outdated information (e.g., discrepancies in message sequences, errors in data processing).
- Improper message processing logic causes misalignment with the server-side state.
- The client may miss critical updates (e.g., changes in orders).

**Solution**:
The client should send a request to obtain the current orderbook-snapshot to realign its state. For more details, see [Orderbook-Snapshot Request].

### 3.4.2 Communication security

Communication is cyphered on network layer with TLS protocol and authentication with client certificate while only TLS 1.2 and higher is accepted.

#### *Authentication*

**Basic Authentication** is a simple and standardized mechanism for user verification within the HTTP protocol. It is used to secure access to web or API resources. In this type of authentication, the username and password are encoded in Base64 format and sent as part of the HTTP header.

**The following credentials are required for user authentication:**

- Name/password.
- Client certificate for authentication on TLS protocol layer where mutual validation of client and server certificates occurs at which moment key and cryptographic algorithm for further communication is arranged.

#### *X-RateLimit-Policy*

The rate-limiting mechanism controls the number of requests that can be sent by a client to the server within a specific timeframe. This policy is designed to protect the server from excessive load (e.g., DDoS attacks) and to ensure fair API usage among different clients.

If the limit is exceeded, the client will be notified with a message of type **"ratelimit-error"**.

### 3.4.3 WebSocket connection

The WebSocket connection is initiated with one or more flags that define what information and operations the user will perform on this connection. Each connected client can open multiple simultaneous connections; however, the maximum number of connections is determined by the number of topics the user subscribes to.

| Method | Basic url |
|---|---|
| GET | wss://{hostname}/api/v1/idm/ws?{query} |

**Table 61** Query Parameters for configuring WebSocket Connection

| Parameter | Value | Description | Use |
|---|---|---|---|
| topics | orderbook / orders / hubtohub / marketstus | Comma-separated message categories the client wishes to receive via this connection:<br>• orderbook: Orderbook changes.<br>• orders: Changes to own orders.<br>• hubtohub: Information on cross-border capacities. | Optional |

| Parameter | Value | Description | Use |
|---|---|---|---|
| | | • marketstatus: Changes in cross-border market status. | |
| orderbookfrequency | Non-negative number | The minimum time interval in milliseconds between messages about changes to the order book. The actual interval depends on market events and system configuration. The default value is 0. This parameter is used to limit the number of messages sent to the client. Order book changes will be aggregated. | Optional |

**Production environment**

| | |
|---|---|
| hostname | isot.okte.sk |
| Server certificate | *.okte.sk |

**SANDBOX environment**

| | |
|---|---|
| hostname | sandbox-isot.okte.sk |
| server certificate | *.okte.sk |

**Test environment**

| | |
|---|---|
| hostname | test-isot.okte.sk |
| server certificate | *.okte.sk |

The market organizer's information system, XMtrade®/ISOT, enables sending and receiving messages in JSON format through a WebSocket channel. These messages can include various information about the intraday market, orders, and changes in the order book.

**Table 62** Supported messages – Outgoing from the client and incoming to the server (in)

| Message | Description |
|---|---|
| orderbook-snapshot (E08-01) | Request for a new orderbook snapshot. |
| order-create (E06-01) | Request for placing orders on the intraday market. |
| order-activate (E06-02) | Request to activate an order by ID. |
| order-deactivate (E06-02) | Request to deactivate an order by ID. |

| | |
|---|---|
| order-cancel (E06-02) | Request to cancel an order by ID. |
| order-detail (E06-03) | Request for the details of own order by ID. |
| hubtohub-snapshot | Request fot the cross-border capacities for the intraday market (IDM). |
| ping | Request to verify that the connection is active on both sides. |
| pong | Response to the connection verification request from the server with a 'ping' message. |

**Table 63** Supported messages – outgoing from the server and incoming to the client (out)

| Message | Description |
|---|---|
| orderbook-snapshot (E08-01) | It provides market participants with an orderbook-snapshot message, which gives a complete overview of the current state of the orderbook. The client will automatically receive this message upon the first connection with the topic=orderbook or based on its own request for a refresh of the orderbook snapshot (orderbook-snapshot (E08-01)). |
| orderbook-change (E12-02) | It provides market participants with changes in the orderbook. |
| order-change (E12-01) | It provides market participants with changes in the status of their own orders. |
| order-error | It provides market participants with information about errors, such as validation errors. |
| hubtohub-snapshot | Provides market participants with information on the status of cross-border capacities for intraday trading (IDT). |
| hubtohub-change (E12-03) | It provides market participants with information about changes in cross-border capacities for the intraday market (IDM). |
| marketstatus | It provides market participants with information about changes in the cross-border market status. |
| ping | The server's request to verify the connection with the client. The client responds with the message "pong." |
| pong | The response to the client's connection verification request with the message "ping." |
| error | Information about a general error, such as in the case of an invalid request from the client. |
| ratelimit-error | Information about exceeding the rate limit for the |

| | |
|---|---|
| | number of requests within a specified time period. |

### 3.4.4 OrderBook

#### *Orderbook-snapshot (E08-01)*

The WebSocket orderbook-snapshot request provides market participants with an automated interface to retrieve the current state of the orderbook, including a sequence number for synchronization. It is used to initialize the orderbook state in case of exceptional situations (e.g., if the client has inconsistencies or errors).

To receive and send messages of type orderbook-snapshot, it is necessary to have an established WebSocket connection with topic orderbook or unspecified (refer to chapter 3.4.3 IDM WebSocket connection).

At the start of the connection, the server sends the client the current state of the order book, including a sequence number used to track subsequent changes. During exceptional situations, the server may send a new order book snapshot at any time to ensure data consistency. The client can send the orderbook-snapshot request message to request the current state of the orderbook anytime.

*Send message (in)*

**Table 64** Description of the request structure

| Send message | Description |
|---|---|
| JSON | Request structure (see description of E-08_01 data flow). |

*Received message (out)*

JSON represents the orderbook snapshot, which includes the market state at specific intervals. The details of each part are described in the following tables.

*Payload*

**Table 65** Payload structure

| Field | Value | Description | Use |
|---|---|---|---|
| seqNo | Integer number | Sequence number of the message. Data consistency check. | Required |
| timeDelta | Integer number | Time difference between the current and previous change in milliseconds. | Required |

***period***

**Table 66** Period structure

| Field | Value | Description | Use |
|---|---|---|---|
| start | YYYY-MM-DDTHH:mm:SSZ | Delivery interval start in UTC. | Required |

| Field | Value | Description | Use |
|---|---|---|---|
| end | YYYY-MM-DDTHH:mm:SSZ | Delivery interval end in UTC. | Required |
| isBlock | true/false | Indication of whether it is a period for block orders. | Required |
| tradingEnd | YYYY-MM-DDTHH:mm:SSZ | Trading end time for the period. | Required |

***statistics***

**Table 67** Statistics structure

| Field | Value | Description | Use |
|---|---|---|---|
| lastTradeTime | YYYY-MM-DDTHH:mm:SSZ | Time of the last trade in period in UTC | Optional |
| lastPrice | Decimal number | The last price for the trading period in EUR/MWh with a precision of two decimal places. | Optional |
| maxPrice | Decimal number | The highest price for the trading period in EUR/MWh with a precision of two decimal places. | Optional |
| minPrice | Decimal number | The lowest price for the trading period in EUR/MWh with a precision of two decimal places. | Optional |
| totalVolume | Decimal number | The total trading volume for the trading period in MW with a precision of one decimal place. | Optional |
| lastQuantity | Decimal number | The last quantity for the trading period in MW with a precision of one decimal place. | Optional |
| priceDirection | 1, -1, 0 | Price trend:<br>• Increasing price,<br>• Decreasing price,<br>• Stable price. | Optional |

***ownStatistics***

- market participant's own statistics based on their orders and trades.

**Table 68** ownStatistics structure

| Field | Value | Description | Use |
|---|---|---|---|
| buy | Object OwnStatistic (see below) | Statistics from purchase orders and transactions. | Optional |
| sell | Object OwnStatistic | Statistics from orders and sales transactions. | Optional |

| Field | Value | Description | Use |
|---|---|---|---|
| | (see below) | | |

***ownStatistic***

**Table 69** ownStatistics structure

| Field | Value | Description | Use |
|---|---|---|---|
| quantity | Decimal number | Own traded volume in period in MW. | Optional |
| remainingQuantity | Decimal number | Remaining quantity of own orders for period in MW | Optional |
| weigtedAveragePrice | Decimal number | Weighted average price of own trades in EUR/MWh. | Optional |

***buyList***

- Sorted offers from the highest offered price.

**Table 70** BuyList structure

| Field | Value | Description | Use |
|---|---|---|---|
| price | Decimal number | Price in EUR/MWh with a precision of two decimal places. | Optional |
| quantity | Decimal number | Quantity in MW with a precision of one decimal place. | Optional |
| ownQuantity | Decimal number | Own quantity in MW with a precision of one decimal place. | Optional |

***sellList***

- Sorted offers from the lowest offered price.

**Table 71** sellList structure

| Field | Value | Description | Use |
|---|---|---|---|
| price | Decimal number | Price in EUR/MWh with a precision of two decimal places. | Optional |
| quantity | Decimal number | Quantity in MW with a resolution of one decimal place. | Optional |
| ownQuantity | Decimal number | Own quantity. | Optional |

***blockOrders***

- Block orders list

**Table 72** blockOrders structure

| Field | Value | Description | Use |
|---|---|---|---|
| price | Decimal number | Price in EUR/MWh with a resolution of two decimal places. | Optional |
| quantity | Decimal number | Quantity in MW with a resolution of one decimal place. | Optional |
| direction | buy / sell | Direction of the order:<br>• buy<br>• sell | Optional |

## *Order-change out (E12-01)*

The order-change message is sent through the WebSocket interface and informs the market participant in real time about the current status (changes) of their orders. These changes may include:

- Creating a new order,
- Updating the status of an existing order (status change).

To receive messages of type order-change, it is necessary to have an established WebSocket connection with topic orders or unspecified (refer to chapter 3.4.3 IDM WebSocket connection).

### *Received message (out)*

**Table 73** Description of the request structure

| Send message | Description |
|---|---|
| JSON | Request structure (see description of E-12_01 data flow). |

*payload*

The field includes the details of the order.

**Table 74** payload structure

| Field | Value | Description | Use |
|---|---|---|---|
| id | Non-negative number | The order identifier recorded in the system. | Required |
| productType | Non-negative number | Product type (period length in minutes):<br>60 – hourly product<br>15 – quarter-hourly product | Required |
| deliveryStart | YYYY-MM-DDTHH:mm:SSZ | It defines the start of the period for which the order is placed in UTC. | Required |
| deliveryEnd | YYYY-MM-DDTHH:mm:SSZ | It defines the end of the period for which the order is placed in UTC. | Required |

| Field | Value | Description | Use |
|---|---|---|---|
| direction | buy/sell | The order class:<br>• buy,<br>• sell. | Required |
| type | simple/block | The order type:<br>• simple<br>• block | Required |
| quantity | Decimal number | The quantity in MW with one decimal place precision. | Required |
| price | Decimal number | The price in EUR/MWh with two decimal places precision. | Required |
| status | active, inactive, canceled, partiallyMatched, matched, expired, expiredInactive | The status of the order in the system. | Required |
| isPending | true/false | An indicator of whether the order is still being processed by the system (e.g., Waiting for activation). | Required |
| realizedQuantity | Decimal number | Realized quantity in MW. | Required |
| realizedPriceWeighted | Decimal number | Realized price in EUR/MWh. | Required |
| remainingQuantity | Decimal number | Remaining quantity in MW. | Required |
| expiration | YYYY-MM-DDTHH:mm:SSZ / null | The date and time of the order's expiration set in UTC. | Optional |
| createdAt | YYYY-MM-DDTHH:mm:SSZ | The time of order creation. | Required |
| updatedAt | Time of order update. | The time of order update. | Required |
| createdBy | String | The name of the user who created the order. | Required |
| clientOrderId | String | The identifier of the order in the client's system. | Optional |
| note | String | Commentary for order | Optional |
| action | added / removed / changed / activated / deactivated / matched / canceled | The event that triggered the order change. | Required |

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | String | Identifier of the related message. | Optional |

## *OrderBook-change out (E12-02)*

An orderbook-change message is sent through the WebSocket interface and provides the market participant with real-time information about changes in the order book. These changes may include:

- Changes triggered directly by the market participant,
- Changes triggered by another market participant,
- Changes triggered by the system (e.g., order expiration, cancellation of available volume in case of period closure).

To receive messages of type orderbook-change, it is necessary to have an established WebSocket connection with topic orderbook or unspecified (refer to chapter 3.4.3 IDM WebSocket connection).

Received message (out)

**Table 75** Description of the request structure

| Send message | Description |
|---|---|
| JSON | Request structure (see description of toku E-12_02 data flow). |

*Payload*

**Table 76** payload structure

| Field | Value | Description | Use |
|---|---|---|---|
| seqNo | Non-negative number | Sequence number of the message. Data consistency check. | Required |
| timeDelta | Non-negative number | Time difference between the current and previous change in milliseconds. | Required |

period

**Table 77** period structure

| Field | Value | Description | Use |
|---|---|---|---|
| start | YYYY-MM-DDTHH:mm:SSZ | Delivery interval start in UTC. | Required |
| end | YYYY-MM-DDTHH:mm:SSZ | Delivery interval end in UTC. | Required |
| isBlock | true/false | Indication of whether it is a period for block orders. | Required |

| Field | Value | Description | Use |
|---|---|---|---|
| tradingEnd | YYYY-MM-DDTHH:mm:SSZ | Trading end time for the period. | Required |

statistics

**Table 78** statistics structure

| Field | Value | Description | Use |
|---|---|---|---|
| lastTradeTime | YYYY-MM-DDTHH:mm:SSZ | Time of the last trade in period in UTC | Optional |
| lastPrice | Decimal number | The last price for the trading period in EUR/MWh with a precision of two decimal places. | Optional |
| maxPrice | Decimal number | The highest price for the trading period in EUR/MWh with a precision of two decimal places. | Optional |
| minPrice | Decimal number | The lowest price for the trading period in EUR/MWh with a precision of two decimal places. | Optional |
| totalVolume | Decimal number | The total trading volume for the trading period in MW with a precision of one decimal place. | Optional |
| lastQuantity | Decimal number | The last quantity for the trading period in MW with a precision of one decimal place. | Optional |
| priceDirection | 1, -1, 0 | Price trend:<br>• Increasing price,<br>• Decreasing price,<br>• Stable price. | Optional |

***ownStatistics***

- Own statistics of the market participant based on their orders and trades.

**Table 79** ownStatistics structure

| Field | Value | Description | Use |
|---|---|---|---|
| buy | Object OwnStatistic (see below) | Statistics from purchase orders and transactions. | Optional |
| sell | Object OwnStatistic (see below) | Statistics from orders and sales transactions. | Optional |

***ownStatistic***

**Table 80** OwnStatistics structure

| Field | Value | Description | Use |
|---|---|---|---|
| quantity | Decimal number | Own traded volume in period in MW. | Optional |
| remainingQuantity | Decimal number | Remaining quantity of own orders for period in MW | Optional |
| weigtedAveragePrice | Decimal number | Weighted average price of own trades in EUR/MWh. | Optional |

buyChanges/sellChanges

**Table 81** buyChanges/sellChanges structure

| Field | Value | Description | Use |
|---|---|---|---|
| index | Integer number | Position of the change within the market depth of the given period. | Optional |
| action | add / remove / update | Change type:<br>• add – addition of a level in the market depth,<br>• update – update at the given level,<br>• remove - removal of a level. | Optional |
| price | Decimal number | Price in EUR/MWh with a precision of two decimal places. | Optional |
| quantity | Decimal number | Quantity in MW with a precision of one decimal place. | Optional |
| quantityDelta | Decimal number | Change in quantity compared to the previous quantity in MW, with a precision of one decimal place. | Optional |
| ownQuantity | Decimal number | Own quantity in MW with a precision of one decimal place. | Optional |
| ownQuantityDelta | Decimal number | Change in own quantity compared to the previous own quantity in MW, with a precision of one decimal place. | Optional |

BlockOrderChanges

**Table 82** blockOrderChanges structure

| Field | Value | Description | Use |
|---|---|---|---|
| action | add / remove / update | Change type:<br>• add – addition of a level in the market depth,<br>• update – update at the given level,<br>• remove - removal of a level. | Optional |

| Field | Value | Description | Use |
|---|---|---|---|
| price | Decimal number | Price in EUR/MWh with a precision of two decimal places. | Optional |
| quantity | Decimal number | Quantity in MW with a precision of one decimal place. | Optional |
| direction | buy / sell | Direction of orders:<br>• buy<br>• sell | Optional |

action

**Table 83** action structure

| Field | Value | Description | Use |
|---|---|---|---|
| action | add / remove / update | Type of change for the given trading period:<br>• add – addition of the period (opening of trading)<br>• update – change within the period<br>• remove – removal of the period (closing of trading) | Optional |

type

**Table 84** type structure

| Field | Value | Description | Use |
|---|---|---|---|
| type | order-create | Order creation. | Required |

### 3.4.5 Order creation

The WebSocket **order-create** interface provides market participants with an automated interface for placing orders on the intraday market. The interface implements the following messages:

- **Send** – placing an order/instruction

#### *IDM Order–create (E06-01)*

To send messages of type order-create, it is necessary to have an established WebSocket connection with topic orders or unspecified (refer to chapter 3.4.3 IDM WebSocket connection).

After successfully establishing the connection, it is possible to send an order-create message, which is used to create a new single order or a list of orders.

*Send message*

**Table 85** Request structure description

| Send message | Description |
|---|---|

| Send message | Description |
|---|---|
| JSON | Request structure (see description of E-06_01 data flow). |

The creation of a new order is carried out through the JSON format. The structure of the message's payload object is the same as specified in Tables 43 and 44 above.

### 3.4.6 Order modification

The WebSocket interface provides market participants with an automated interface for modifying orders (changing their status). The interface implements the following messages:

- order-activate - message for activating an order,
- order-deactivate - message for deactivating an order,
- order-cancel - message for canceling an order.

#### *IDM order–activate (E06-02)*

To send messages of type order-activate, it is necessary to have an established WebSocket connection with topic orders or unspecified (refer to chapter 3.4.3 IDM WebSocket connection). After successfully establishing the connection, you can send an order-activate message, which is used to modify an order by its ID.

*Send message*

**Table 86** Request structure description

| Send message | Desciption |
|---|---|
| JSON | Request structure (see description of E-06_02 data flow). |

The change of the order status is implemented through the JSON format. The structure of the message is as follows.

payload

**Table 87** payload structure

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | String | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12-01). | Optional |
| Id | Integer number | The order ID whose status will be changed. | Required |

type

**Table 88** type structure

| Field | Value | Description | Use |
|---|---|---|---|
| type | order-activate | Type of operation. | Required |

## *IDM order–deactivate (E06-02)*

To send messages of type order-deactivate, it is necessary to have an established WebSocket connection with topic orders or unspecified (refer to chapter 3.4.3 IDM WebSocket connection). After successfully establishing the connection, you can send an order-deactivate message, which is used to modify an order by its ID.

### *Send message*

**Table 89** Request structure description

| Send message | Desciption |
|---|---|
| JSON | Request structure (see description of E-06_02 data flow). |

The change of the order status is implemented through the JSON format. The structure of the message is as follows.

payload

**Table 90** payload structure

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | String | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12-01). | Optional |
| Id | Integer number | The order ID whose status will be changed. | Required |

type

**Table 91** type structure

| Field | Value | Description | Use |
|---|---|---|---|
| type | order-deactivate | Type of operation. | Required |

### *IDM order-cancel (E06-02)*

To send messages of type order-cancel, it is necessary to have an established WebSocket connection with topic orders or unspecified (refer to chapter 3.4.3 IDM WebSocket connection). After successfully establishing the connection, you can send an order-cancel message, which is used to modify an order by its ID.

*Send message*

**Table 92** Request structure description

| Send message | Desciption |
|---|---|
| JSON | Request structure (see description of E-06_02 data flow). |

The change of the order status is implemented through the JSON format. The structure of the message is as follows.

payload

**Table 93** payload structure

| Field | Value | Description | Use |
|---|---|---|---|
| correlationId | String | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01 or WebSocket protocol data flow E-12-01). | Optional |
| Id | Integer number | The order ID whose status will be changed. | Required |

type

**Table 94** type structure

| Field | Value | Description | Use |
|---|---|---|---|
| type | order-cancel | Type of operation. | Required |

## 3.4.7 Provision of orders

The WebSocket interface provides market participants with an automated way to retrieve details of a specific order using its ID. However, this interface does not support access to the list of all orders.

The interface implements the following message:

- Send Message (order-detail)

This message is used to access the details of an order based on its unique ID. The client sends a request to the server, which then responds with detailed information about the specified order.

### *IDM Order–detail (E06-03)*

To send messages of type order-detail and to receive response, it is necessary to have an established WebSocket connection with topic orders or unspecified (refer to chapter 3.4.3 IDM WebSocket connection). After a successful connection is established, you can send an **order-detail** message, which is used to access the details of an order.

#### *Send message*

**Table 95** Request structure

| Send message | Description |
|---|---|
| JSON | Request structure (see description of E-06_03 data flow). |

#### *Received mesage*

It is implemented through the JSON format. The structure of the response payload is the same as described in Tables 53 and 54 above.

type

**Table 96** Type structure

| Field | Value | Description | Use |
|---|---|---|---|
| type | order-detail | Type of operation. | Required |

## 3.4.8 Hub-to-hub

To receive messages of type hubtohub-snapshot and hubtohub-change, it is necessary to have an established WebSocket connection with topic hubtohub or unspecified (refer to chapter 3.4.3 IDM WebSocket connection).

At the start of the connection, the server sends the client the current state of the hub-to-hub matrix in a hubtohub-snapshot message. During exceptional situations, the server may send a new hubtohub-snapshot at any time to ensure data consistency.

A hubtohub-change message is sent via the WebSocket interface to inform market participants in real-time about the current status of cross-border capacities.

#### *Received message (out)*

*payload*

**Table 97** Payload structure.

| Field | Value | Description | Use |
|---|---|---|---|
| eic | String | EIC of the Market Area. | Required |
| areaName | String | Name of the Market Area. | Required |
| countryCode | String | Two-letter country code | Required |
| deliveryDay | YYYY-MM-DD | Date of the delivery day. | Required |
| deliveryStart | YYYY-MM-DDTHH:mm:SSZ | Delivery period start in UTC | Required |
| deliveryEnd | YYYY-MM-DDTHH:mm:SSZ | Delivery period end in UTC | Required |
| availableCapacityIn | Decimal number | Available import capacieties in MW. | Required |
| availableCapacityOut | Decimal number | Available export capacities in MW. | Required |

The client can request the current state of cross-border capacities by sending a hubtohub-snapshot message without a payload.

### 3.4.9 Market status

To receive messages of type marketstatus, it is necessary to have an established WebSocket connection with topic marketstatus or unspecified (refer to chapter 3.4.3 IDM WebSocket connection).

A marketstatus message provides information about the current status of the cross-border market. It is sent to the client upon connection and whenever the market status changes.

*Received message (out)*

*payload*

**Table 98** Payload structure.

| Field | Value | Description | Use |
|---|---|---|---|

| Field | Value | Description | Use |
|---|---|---|---|
| systemTime | YYYY-MM-DDTHH:mm:ss.ffffffZ | System time in UTC | Required |
| tradeDay | YYYY-MM-DD | Trade day | Required |
| tradingStatus | xbidOk, xbidNok, xbidHalt, xbidBatchMatching, localOn, localOff | Trading status | Required |

### 3.4.1 Ping

Ping and Pong messages are used to verify the activity and proper functioning of the connection between the client and server over WebSocket.

**Ping (in, out):** This message is used to verify whether the connection between the client and server is still active. Either the server or the client can send a Ping message. The other party responds with a Pong message, indicating that the connection is active and functioning properly.

The client is not required to send a ping message at any interval. However, if the client does not respond with a pong message within 5 seconds of receiving a ping message from the server, the connection is considered inactive and will be terminated by the server.

Ping and pong messages are not tied to any specific topic; both the client and server can send them over any connection.

*Ping message (in, out):*

```
connect: wss://{hostname}/api/v1/idm/ws

JSON:

{
  "type":"ping"
}
```

*Pong message – (the response to a Ping message):*

```
JSON:

{
  "type":"pong"
}
```

# 4 SPECIFICATION OF DATA STRUCTURES

The market organizer information system XMtrade®/ISOT allows automated data exchange through interface that involves use of data structures based on XML format in accordance with *ENTSO-E* standards and data structures based on established XML formats of the IS OTE and IS OKTE systems:

- ENTSO-E Capacity Allocation and Nomination (ECAN)
  (System of capacity allocation and nomination),
- ENTSO-E Acknowledgement Process (EAD)
  (Process for acknowledgement documents),
- ENTSO-E Status Request (ESR)
  (Retrieval of status information),
- Energy Identification Coding Scheme (EIC)
  (System of identification in the energy sector),
- ENTSO-E General Code List for Data Interchange (ECL)
  (Code lists used in *ENTSO-E* standards),
- ENTSO-E Harmonized Electricity Market Role Model
  (Role model on the electricity market),
- ISOT: ISOTEDATA, ISOTEDATA-VDT, RESPONSE, RESPONSE-VDT, CDSREQ, CDSREQ-VDT.

(Structures are derived from the information system of market operator in the Czech Republic).

Date and time items in structures are used in UTC (Universal Time) format.

**Table 99** Structures and data flows overview

| Interface | Process | ID | Direction | Format |
|---|---|---|---|---|
| Market participant order administration | Reception of orders | E-02_01 | Input | ISOTEDATA.811 |
| | | | Output | RESPONSE.812<br>ISOTEDATA.813 |
| | Provision of orders | E-02_03 | Input | CDSREQ.831 |
| | | | Output | RESPONSE.832<br>ISOTEDATA.833 |
| ID market Order administration | Reception of orders | E-06_01 | Input | ISOTEDATA-VDT.801 |
| | | | Output | RESPONSE-VDT.802<br>ISOTEDATA-VDT.803 |
| | Modification of orders | E-06_02 | Input | ISOTEDATA-VDT.804 |
| | | | Output | RESPONSE-VDT.805<br>ISOTEDATA-VDT.806 |
| | Provision of orders | E-06_03 | Input | CDSREQ-VDT.807 |
| | | | Output | RESPONSE-VDT.808<br>ISOTEDATA-VDT.809 |

| Interface | Process | ID | Direction | Format |
|---|---|---|---|---|
| ID market order book data | Access to order book | E-08_01 | Input | CDSREQ-VDT.810 |
| | | | Output | RESPONSE-VDT.811<br>ISOTEDATA-VDT.812 |
| Results and evaluations of DM | Notification of results and evaluations | E-03_02 | Input | CDSREQ.941 |
| | | | Output | RESPONSE.942<br>ISOTEDATA.943 |
| | | E-05_01 | Input | CDSREQ.951 |
| | | | Output | RESPONSE.952<br>ISOTEDATA.953 |
| | | E-05_02 | Input | CDSREQ.961 |
| | | | Output | RESPONSE.962<br>ISOTEDATA.963 |
| ID Results | Notification about evaluation | E-07_01 | Input | CDSREQ-VDT.961 |
| | | | Output | RESPONSE-VDT.962<br>ISOTEDATA-VDT.963 |
| | | E-07_02 | Input | CDSREQ-VDT.571 |
| | | | Output | RESPONSE-VDT.572<br>ISOTEDATA-VDT.573 |
| | | E-07_03 | Input | CDSREQ-VDT.951 |
| | | | Output | RESPONSE-VDT.952<br>ISOTEDATA-VDT.953 |
| MCC Details | Notification of MCC values | E-01_02 | Input | ESR.StatusRequest |
| | | | Output | EAD.AcknowledgementDocument<br>ECAN.CapacityDocument |
| Own order status change | Notification about modification/creation of own order through AMQP | E-10_01 | Output | ISOTEDATA-VDT.820 |
| Change in order book | Notification about change in order book through AMQP | E-10_02 | Output | ISOTEDATA-VDT.830 |

## 4.1 Common data structures

### 4.1.1 ISOTEDATA

ISOTEDATA structure is a generic data structure that is derived from structures used by the IS OTE system within data exchange with market participants in the Czech Republic.

The principle of using this structure in different communication scenarios is based on the specification, i.e. message code (*message-code* attribute) that determines the type/purpose of the content. Due to clarity, types of this structure are further used in the text in *ISOTEDATA.message-code* format.

#### *DAM order structure*

Order for day-ahead market represented by *ISOTEDATA* structure consists of the following parts:

- *ISOTEDATA* - contains general details relating to the entire message (message header),
- *Trade* – represents the order itself (order header),
- *ProfileData* – order blocks, stated always in a pair – once it represents amount of electricity, next time it represents the price (*profile-role* attribute specifies the type),
- *Data* – contains values for specific hours of a trading day in the meaning according to *ProfileData* type.

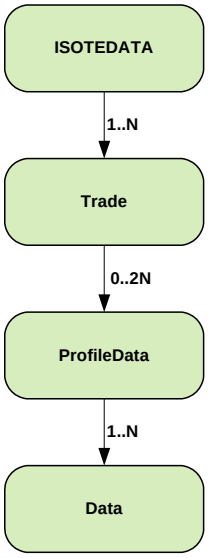


**Figure 11** Scheme of order structure

*ISOTEDATA*

ISOTEDATA root element contains attribute values in accordance with the following table.

**Table 100** ISOTEDATA root element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message-code | number/code of the message | – It determines the purpose of structure content:<br>– 811 – submission of order, | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| | | – 813 – description of order data (response to 811),<br>– 833 - description of order data (response to CDSREQ.831 request). | |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>– YYYY – year,<br>– MM – month,<br>– DD – day,<br>– HH – hour,<br>– mm – minute,<br>– SS – second. | Required |
| answer-required | 0/1 | Attribute of response requirement:<br>– 0 – no,<br>– 1 – yes. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 101** Message header of displayed values of particular elements

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Receiver Identification/id | 24X-OT-SK------V | Message receiver identification.<br>EIC = 24X-OT-SK------V is used. | Required |
| Reference/id | String | Identification of related message, e.g. message involving request for own order data (it is used in responses to the message correlation). | Optional |

*Trade*

Trade order element contains values of particular attributes in accordance with the following table.

**Table 102** Trade order element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | non-negative number | Order identification in the system (it is used in responses to the order or | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| | | in the request for modification of existing order). | |
| version | non-negative number | Order version registered in the system (it is used in responses to the order). | Optional |
| trade-day | YYYY-MM-DD | Date of trading day.<br>Date and time format:<br>– YYYY – year,<br>– MM – month,<br>– DD – day. | Required |
| trade-type | N/P | Order class:<br>– N – purchase (in Slovak: nákup),<br>– P – sale (in Slovak: predaj). | Required |
| trade-stage | N/P | Order status in the system:<br>– N – invalid (in Slovak: neplatný),<br>– P – valid (in Slovak: platný),<br>– (it is used in the response/ description of the order). | Optional |
| acceptance | A/N | Indicator of total acceptance of block No. 1.<br>– A - yes, block no. 1 is accepted in whole (in Slovak: áno).<br>– N - no, order is divisible by the time periods.<br>In case of new form of orders this element will not be used. | Optional |
| block-order | A/N | Indication of block order.<br>– A – yes, the submitted order is block.<br>– N – no, the submitted order is a simple hourly order.<br>Attibute is required in case of new form of orders. | Optional |
| block-type | SB/LB/FB/EG | Block order type.<br>– SB – simple block order,<br>– LB – linked block order,<br>– FB – flexible block order,<br>– EG – exclusive group of simple block orders.<br>Required in case block-order = “A”. | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| linked-order-id | non-negative number | Identification of linked order.<br>Required in case block-type=”LB”. | Optional |
| sett-curr | EUR | Trading currency. | Required |
| market-area | SK | Trading area (SK). | Required |
| market | DAM/IDA | Type of market | Required |
| delivery-duration | 60/15 | Length of the trading period | Required |

Trade order element contains particular elements values in accordance with the following table.

**Table 103** Trade order element

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Party/id | Data owner EIC. | Identification of data owner on behalf of whom the data are sent. When owner of the data is sending them for himself, it is identical to SenderIdentification.<br>Entity EIC is used.<br>Maximum 16 characters. | Required |
| Party/role | TO | Role of the owner.<br>– TO – owner of trade. | Required |
| Comment | string | Comment on the order. | Optional |
| TimeData/ datetime | YYYY-MM-DDTHH:mm:SSZ | Order timestamp in UTC (Universal Time):<br>– YYYY – year,<br>– MM – month,<br>– DD – day,<br>– HH – hour,<br>– mm – minute,<br>– SS – second<br>*(system generated value).* | Optional |
| TimeData/ datetime-type | DTC | Type of timestamp:<br>– DTC – date and time of order submission into the system (system generated value). | Optional |

*ProfileData*

ProfileData element of order block contains values of particular attributes in accordance with the following table.

**Table 104** ProfileData element of order block

| Attribute | Value | Description | Use |
|---|---|---|---|
| profile-role | BC01 - BC25<br>BP01 - BP25 | Determines the order and purpose of a block.<br>BC01 - BC24: 1. - 24. block containing the amount of energy.<br>BP01 - BP24: 1. - 24. block containing price of corresponding amount of energy.<br>During time shift from Central European time to Central European Summer time and backwards, 23 or 25 values are used, respectively.<br>BC01 - BC25, or BC01 - BC23 respectively.<br>BP01 - BP23, or BP01 - BP23 respectively. | Required |

ProfileData are always entered in a pair, where one element contains details on energy amounts (BC01-BC25) and the second contains the corresponding price details to amounts (BP01-BP25).

*Data*

Data element of data block contains values for specific hours within the day in the meaning according to block type (ProfileData/@profile-role).

**Table 105** Data element of data block

| Attribute | Value | Description | Use |
|---|---|---|---|
| period | Non-negative number | Specifies the index of the period within the day. For a 60-minute interval 1…24, and for a 15-minute interval: 1..96<br>During time shift from Central European time to Central European Summer time and backwards, 23 or 25 for the 60-minute interval and 92 or 100 values for the 15-minute interval. | Required |
| value | Decimal number | Contains Amount/Price.<br>Amount with precision to one decimal place.<br>Price with precision to two decimal places.<br>Separator of decimal places “.“ (point). | Required |
| unit | MWH/EUR | Data unit used in value. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| splitting | A/N | Divisibility – performance divisibility:<br>– A - yes, amount is divisible (default value) (in Slovak: áno),<br>– N - no, amount is not divisible (in Slovak: nie). | Optional |

## *Structure of DM results and evaluations*

Results of DM trading are represented by *ISOTEDATA* structure consisting of the following parts:

- *ISOTEDATA* - contains general details relating to the entire message (message header),
- *Trade* – represents DM results in a given day (result header),
- *ProfileData* – blocks of results (*profile-role* attribute specifies the type),
- *Data* – contains values for specific hours within the day with the meaning according to *ProfileData* type.

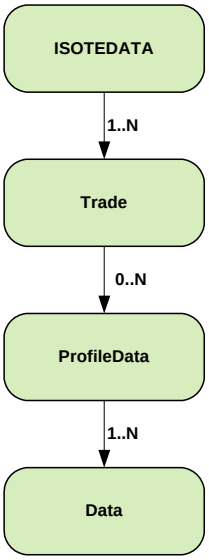


**Figure 12** DM results structure scheme

### *ISOTEDATA*

ISOTEDATA root element contains attribute values according to the following table.

**Table 106** ISOTEDATA root element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message -code | number/code of the message | Determines the purpose of structure content:<br>– 943 – results per entity,<br>– 953 – evaluation per periods,<br>– 963 – evaluation for day. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>– YYYY – year,<br>– MM – month,<br>– DD – day,<br>– HH – hour,<br>– mm – minute,<br>– SS – second. | Required |
| answer-required | 0/1 | Attribute of response requirement:<br>– 0 - no,<br>– 1 - yes. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 107** Displayed values message header of particular elements

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Receiver Identification/id | 24X-OT-SK------V | Message receiver Identification.<br>EIC = 24X-OT-SK------V is used. | Required |
| Reference/id | String | Identification of related message, e.g. the message on request for own order data (it is used in response to message correlation). | Optional |

*Trade*

Trade element of DM result/evaluation for a given trading day contains values of particular attributes in accordance with the following table.

**Table 108** Trade element 1 of DM result/evaluation for a given trading day

| Attribute | Value | Description | Use |
|---|---|---|---|
| trade-day | YYYY-MM-DD | Date of trading day.<br>Date and time format:<br>– YYYY - year,<br>– MM - month,<br>– DD - day. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| delivery-duration | 60/15 | Duration of trading period:<br>15 – quarter-hour (15 min.)<br>60 – hour (60 min.) | Required |

Trade element of DM result/evaluation for a given trading day contains values of particular elements in accordance with the following table.

**Table 109** Trade element 2 of DM result/evaluation for a given trading day

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Party/id | Data owner EIC | Identifier of data owner for whom the data are sent. In case of owner sending data for himself, it is identical with SenderIdentification. Entity EIC is used. Maximum 16 characters. | Required |
| Party/role | TO | Role of the owner:<br>– TO - owner of trade. | Required |
| ResultStatus/status | P, F | Result status:<br>– P – preliminary,<br>– F – final.<br>Usage of element only in case of data flows E-03_02, E-05_01 a E-05_02.<br>Status of the result for already closed days before D (while D is the trading day) is by default assigned as final. Element is used only for trading days which are operated under Interim Coupling regime or during local auction resulted from this regime. | Optional |

*ProfileData*

ProfileData element of result block contains values of particular attributes in accordance with the following table.

**Table 110** ProfileData element of result block

| Attribute | Value | Description | Use |
|---|---|---|---|
| profile-role | SP02, SC02, SP03, SC03, SP05, SC05, SC19, SP20, SC20, SP90, SC90, SP91, SC91, SP92, SC92, SP93, SC93, ST16 | Specifies the meaning of result blocks.<br>*Results:*<br>– SC19 – amount of electricity purchased for null or positive prices (positive number),<br>– SC20 – amount of electricity sold for null or positive prices (positive number),<br>– SP20 – marginal price (negative, null or positive number),<br>– SC92 – amount of electricity purchased for negative prices (positive number),<br>– SC93 – amount of electricity sold for negative prices (positive number).<br><br>*Evaluation:*<br>– SP02 – clearing/payment for electricity purchased for null or positive prices (null or positive number),<br>– SC02 – amount of electricity purchased for null or positive prices (positive number),<br>– SP03 – clearing/payment for electricity sold for null or positive prices (null or positive | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| | | number),<br>– SC03 – amount of electricity sold for null or positive prices (positive number),<br>– SP05 – fee for traded electricity (positive number),<br>– SC05 – amount of traded electricity (sum of electricity sold and purchased for positive, null and also negative prices) (positive number),<br>– ST16 – monthly fee for access to daily market (positive number); stated in the period 0 on the last day of the month,<br>– SP90 – fee for trading transactions relating to data manipulation (positive number),<br>– SC90 – amount of trading transactions relating to data manipulation (positive number),<br>– SP91 – fee for trading transactions using automated interfaces (positive number),<br>– SC91 – amount of trading transactions using automated interfaces (positive number),<br>– SP92 – clearing/payment for electricity purchased for negative prices (positive number),<br>– SC92 – amount of electricity purchased for negative prices (positive number),<br>– SP93 – clearing/payment for electricity sold for negative prices (positive number),<br>– SC93 – amount of electricity sold for negative prices (positive number).<br><br>*Note: payments and fees are stated excluding VAT and electricity tax* | |

*Data*

Data element of data block contains values for specific hours within the day in the meaning according to block type (ProfileData/@profile-role).

**Table 111** Data element of data block

| Attribute | Value | Description | Use |
|---|---|---|---|
| period | Non-negative number | Determines the index of the period within a day.<br><br>For a 60-minute interval, the range is 1 to 24, and for a 15-minute interval, the range is 1 to 96.<br><br>During the transition from standard time (CET) to daylight saving time (CEST) and vice versa, 23 or 25 values are used for the 60-minute interval, and 92 or 100 values for the 15-minute interval.~~Specifies the index of hour within the day,~~ | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| | | ~~whereas it can hold values from 1 to 24 (during time shift from Central European time to Central European Summer time and backwards, 23 or 25 periods are used, respectively).~~ Summary daily values are stated in the period with index 0. | |
| value | Decimal number | In case of amount with the precision to: one decimal place.<br>In case of price with the precision to: two decimal places.<br>Separator of decimal places "." (point). | Required |
| unit | MWH, EUR | Unit of amount stated in values. | Required |

## *IDA order structure*

Order for intraday auction represented by *ISOTEDATA* structure consists of the following parts:

- *ISOTEDATA* - contains general details relating to the entire message (message header),
- *Trade* – represents the order itself (order header),
- *ProfileData* – order blocks, stated always in a pair – once it represents amount of electricity, next time it represents the price (*profile-role* attribute specifies the type),
- *Data* – contains values for specific hours of a trading day in the meaning according to *ProfileData* type.

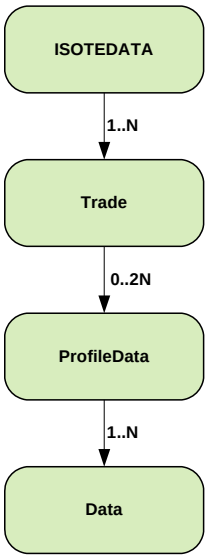


**Figure 13** Scheme of order structure

### *ISOTEDATA*

ISOTEDATA root element contains attribute values in accordance with the following table.

**Table 112** ISOTEDATA root element

| Attribute | Value | Description | Use |
|---|---|---|---|

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message-code | number/code of the message | – It determines the purpose of structure content:<br>– 851 – submission of order,<br>– 813 – description of order data (response to 851),<br>– 833 – description of order data (response to CDSREQ.831 request). | Required |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>– YYYY – year,<br>– MM – month,<br>– DD – day,<br>– HH – hour,<br>– mm – minute,<br>– SS – second. | Required |
| answer-required | 0/1 | Attribute of response requirement:<br>– 0 – no,<br>– 1 – yes. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 113** Message header of displayed values of particular elements

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Receiver Identification/id | 24X-OT-SK------V | Message receiver identification.<br>EIC = 24X-OT-SK------V is used. | Required |
| Reference/id | String | Identification of related message, e.g. message involving request for own order data (it is used in responses to the message correlation). | Optional |

*Trade*

Trade order element contains values of particular attributes in accordance with the following table.

**Table 114** Trade order element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | non-negative number | Order identification in the system (it is used in responses to the order or in the request for modification of existing order). | Optional |
| version | non-negative number | Order version registered in the system (it is used in responses to the order). | Optional |
| trade-day | YYYY-MM-DD | Date of trading day.<br>Date and time format:<br>– YYYY – year,<br>– MM – month,<br>– DD – day. | Required |
| auction-id | String | IDA auction identifier:<br>• IDA1 – first intraday auction for relevant trading day<br>• IDA2 – second intraday auction for relevant trading day<br>• IDA3 – third intraday auction for relevant trading day | Required |
| trade-type | N/P | Order direction:<br>– N – purchase (in Slovak: nákup),<br>– P – sale (in Slovak: predaj). | Required |
| trade-stage | N/P | Order status in the system:<br>– N – invalid (in Slovak: neplatný),<br>– P – valid (in Slovak: platný),<br>– (it is used in the response/ description of the order). | Optional |
| block-order | A/N | Indication of block order.<br>– A – yes, the submitted order is block.<br>– N – no, the submitted order is a simple hourly order.<br>Attibute is required in case of new form of orders. | Optional |
| block-type | SB/LB/FB/EG | Block order type.<br>– SB – simple block order,<br>– LB – linked block order,<br>– FB – flexible block order,<br>– EG – exclusive group of simple block orders.<br>Required in case block-order = “A”. | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| market | IDA | Market type:<br>IDA = IDA – intraday auction | Required |
| delivery-duration | 15 | Duration of trading period:<br>15 – quarter-hour (15 min.) | Required |
| sett-curr | EUR | Trading currency. | Required |
| market-area | SK | Trading area (SK). | Required |

Trade order element contains particular elements values in accordance with the following table.

**Table 115** Trade order element

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Party/id | Data owner EIC. | Identification of data owner on behalf of whom the data are sent. When owner of the data is sending them for himself, it is identical to SenderIdentification.<br>Entity EIC is used.<br>Maximum 16 characters. | Required |
| Party/role | TO | Role of the owner.<br>– TO – owner of trade. | Required |
| Comment | string | Comment on the order. | Optional |
| TimeData/ datetime | YYYY-MM-DDTHH:mm:SSZ | Order timestamp in UTC (Universal Time):<br>– YYYY – year,<br>– MM – month,<br>– DD – day,<br>– HH – hour,<br>– mm – minute,<br>– SS – second<br>*(system generated value).* | Optional |
| TimeData/ datetime-type | DTC | Type of timestamp:<br>– DTC – date and time of order submission into the system (system generated value). | Optional |

### *ProfileData*

ProfileData element of order block contains values of particular attributes in accordance with the following table.

**Table 116** ProfileData element of order block

| Attribute | Value | Description | Use |
|---|---|---|---|
| profile-role | BC01 - BC25<br>BP01 - BP25 | Determines the order and purpose of a block.<br>BC01 - BC24: 1. - 24. block containing the amount of energy.<br>BP01 - BP24: 1. - 24. block containing price of corresponding amount of energy.<br>During time shift from Central European time to Central European Summer time and backwards, 23 or 25 values are used, respectively.<br>BC01 - BC25, or BC01 - BC23 respectively.<br>BP01 - BP23, or BP01 - BP23 respectively. | Required |

ProfileData are always entered in a pair, where one element contains details on energy amounts (BC01-BC25) and the second contains the corresponding price details to amounts (BP01-BP25).

### *Data*

Data element of data block contains values for specific hours within the day in the meaning according to block type (ProfileData/@profile-role).

**Table 117** Data element of data block

| Attribute | Value | Description | Use |
|---|---|---|---|
| period | Non-negative number | Specifies the index of trading period within the day.<br>1…96.<br>During time shift from Central European time to Central European Summer time and backwards, 92 or 100 values are used, respectively. | Required |
| value | Decimal number | Contains Amount/Price.<br>Amount with precision to one decimal place.<br>Price with precision to two decimal places.<br>Separator of decimal places “.“ (point). | Required |
| unit | MW/EUR | Data unit used in value. | Required |
| splitting | A/N | Divisibility – performance divisibility:<br>– A - yes, amount is divisible (default value) (in Slovak: áno),<br>– N - no, amount is not divisible (in Slovak: nie). | Optional |

## *Structure of IDA results and evaluations*

Results of IDA trading are represented by *ISOTEDATA* structure consisting of the following parts:

- *ISOTEDATA* - contains general details relating to the entire message (message header),
- *Trade* – represents IDA results in a specific intraday auction (result header),
- *ProfileData* – blocks of results (*profile-role* attribute specifies the type),
- *Data* – contains values for specific periods within the day with the meaning according to *ProfileData* type.

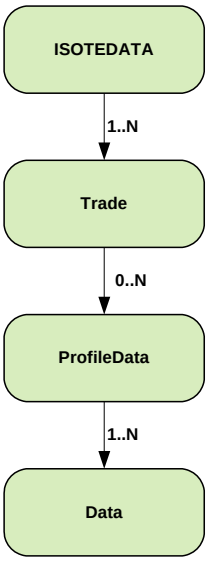


**Figure 14** IDA results structure scheme

### *ISOTEDATA*

ISOTEDATA root element contains attribute values according to the following table.

**Table 118** ISOTEDATA root element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message -code | number/code of the message | Determines the purpose of structure content:<br>– 943 – results per entity,<br>– 953 – evaluation per hour,<br>– 963 – evaluation for day. | Required |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>– YYYY – year,<br>– MM – month,<br>– DD – day,<br>– HH – hour,<br>– mm – minute,<br>– SS – second. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| answer-required | 0/1 | Attribute of response requirement:<br>– 0 - no,<br>– 1 - yes. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 119** Displayed values message header of particular elements

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Receiver Identification/id | 24X-OT-SK------V | Message receiver Identification.<br>EIC = 24X-OT-SK------V is used. | Required |
| Reference/id | String | Identification of related message, e.g. the message on request for own order data (it is used in response to message correlation). | Optional |

*Trade*

Trade element of DM result/evaluation for a given trading day contains values of particular attributes in accordance with the following table.

**Table 120** Trade element 1 of DM result/evaluation for a given trading day

| Attribute | Value | Description | Use |
|---|---|---|---|
| trade-day | YYYY-MM-DD | Date of trading day.<br>Date and time format:<br>– YYYY - year,<br>– MM - month,<br>– DD - day. | Required |
| auction-id | String | IDA auction identifier:<br>• IDA1 – first intraday auction for relevant trading day<br>• IDA2 – second intraday auction for relevant trading day<br>• IDA3 – third intraday auction for relevant trading day | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| delivery-duration | 15 | Duration of trading period:<br>15 – quarter-hour (15 min.) | Required |

Trade element of IDA result/evaluation for a given trading day and auction contains values of particular elements in accordance with the following table.

**Table 121** Trade element 2 of DM result/evaluation for a given trading day

| Element/Attribute | Value | Description | Use |
|---|---|---|---|
| Party/id | Data owner EIC | Identifier of data owner for whom the data are sent. In case of owner sending data for himself, it is identical with SenderIdentification. Entity EIC is used.<br>Maximum 16 characters. | Required |
| Party/role | TO | Role of the owner:<br>– TO - owner of trade. | Required |
| ResultStatus/status | P, F | Result status:<br>– P – preliminary,<br>– F – final.<br>Usage of element only in case of data flows E-03_02, E-05_01 a E-05_02.<br>Status of the result for already closed days before D (while D is the trading day) is by default assigned as final. Element is used only for trading days which are operated under Interim Coupling regime or during local auction resulted from this regime. | Optional |

*ProfileData*

ProfileData element of result block contains values of particular attributes in accordance with the following table.

**Table 122** ProfileData element of result block

| Attribute | Value | Description | Use |
|---|---|---|---|
| profile-role | SP02, SC02, SP03, SC03, SP05, SC05, SC19, SP20, SC20, SP90, SC90, SP91, SC91, SP92, SC92, SP93, SC93, ST16 | Specifies the meaning of result blocks.<br>*Results:*<br>– SC19 – amount of electricity purchased for null or positive prices (positive number),<br>– SC20 – amount of electricity sold for null or positive prices (positive number),<br>– SP20 – marginal price (negative, null or positive number),<br>– SC92 – amount of electricity purchased for negative prices (positive number),<br>– SC93 – amount of electricity sold for negative prices (positive number).<br><br>*Evaluation:*<br>– SP02 – clearing/payment for electricity purchased for null or positive prices (null or positive number),<br>– SC02 – amount of electricity purchased for null or positive prices (positive number),<br>– SP03 – clearing/payment for electricity sold for null or positive prices (null or positive | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| | | number),<br>– SC03 – amount of electricity sold for null or positive prices (positive number),<br>– SP05 – fee for traded electricity (positive number),<br>– SC05 – amount of traded electricity (sum of electricity sold and purchased for positive, null and also negative prices) (positive number),<br>– ST16 – monthly fee for access to daily market (positive number); stated in the period 0 on the last day of the month,<br>– SP90 – fee for trading transactions relating to data manipulation (positive number),<br>– SC90 – amount of trading transactions relating to data manipulation (positive number),<br>– SP91 – fee for trading transactions using automated interfaces (positive number),<br>– SC91 – amount of trading transactions using automated interfaces (positive number),<br>– SP92 – clearing/payment for electricity purchased for negative prices (positive number),<br>– SC92 – amount of electricity purchased for negative prices (positive number),<br>– SP93 – clearing/payment for electricity sold for negative prices (positive number),<br>– SC93 – amount of electricity sold for negative prices (positive number).<br><br>*Note: payments and fees are stated excluding VAT and electricity tax* | |

*Data*

Data element of data block contains values for specific hours within the day in the meaning according to block type (ProfileData/@profile-role).

**Table 123** Data element of data block

| Attribute | Value | Description | Use |
|---|---|---|---|
| period | Non-negative number | Specifies the index of trading period within the day.<br>1…96.<br>During time shift from Central European time to Central European Summer time and backwards, 92 or 100 values are used, respectively.<br>Summary daily values are stated in the period with index 0. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| value | Decimal number | In case of amount with the precision to: three decimal places.<br>In case of price with the precision to: two decimal places.<br>Separator of decimal places "." (point). | Required |
| unit | MWH, EUR | Unit of amount stated in values. | Required |

### 4.1.2 ISOTEDATA-VDT

ISOTEDATA-VDT structure is common data structure used for data exchange with market participants within intraday continuous market.

The principle of using this structure in various communication scenarios resides in so-called message code (attribute message-code), with which the type/purpose of content is determined. Types of this structure due to clarity are used in text bellow as ISOTEDATA-VDT.message-code.

ISOTEDATA-VDT structure is used even for sending notification messages through AMQP protocol.

#### *ISOTEDATA-VDT Structure*

Order or order book description represented by ISOTEDATA-VDT structure consists of these parts:

- *ISOTEDATA* – contains common data relating to the whole message (message header),
- *Trade* – represents the order itself (order header), group of quantities/prices and last price and quantity in the case of order book data transfer,
- *ProfileData* – order blocks, always shown in pair – first time it states amount of energy and second it states the price (attribute profile-role determines the type). Except profile-role= [TC01, LC01, LP01] in case of orderbook blocks conatin information such as overall and last traded quantity including last price and price direction.
- *Data* – contains the values for specific hours of trade day in meaning according to type ProfileData.

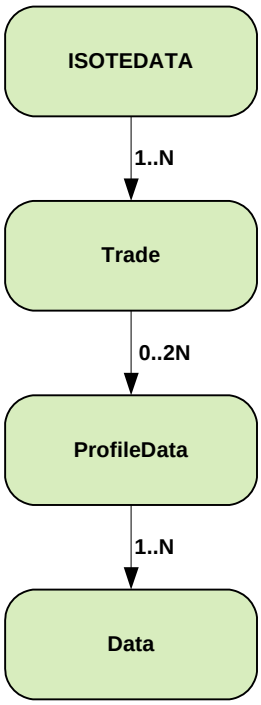


**Figure 15** Scheme of order structure

*ISOTEDATA-VDT*

ISOTEDATA-VDT root element contains attribute values in accordance with the following table.

**Table 124** ISOTEDATA-VDT root element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message-code | number/code of the message | Determinates purpose of structure content:<br>– 573 – monthly evaluation IDM (response to CDSREQ-VDT.571),<br>– 801 – submission of order,<br>– 803 – description of order data (response to ISOTEDATA-VDT.801),<br>– 804 – order modification,<br>– 806 - data description of modified order (response to ISOTEDATA-VDT.804),<br>– 809 – data description of order (response to CDSREQ-VDT.807),<br>– 812 – description of order book data (response to CDSREQ-VDT.810),<br>– 820 – notification about status change of order through AMQP (automatically),<br>– 830 – notification about order book status change through AMQP (automatically),<br>– 953 – IDM evaluation per periods (response to CDSREQ-VDT.951),<br>– 963 – daily IDM evaluation (response to CDSREQ-VDT.961). | Required |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>– YYYY - year,<br>– MM - month,<br>– DD - day,<br>– HH - hour,<br>– mm - minute,<br>– SS - second. | Required |
| answer-required | 0/1 | Attribute of response requirement.<br>– 0 – no,<br>– 1 – yes. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 125** Message header displayed values of single elements

| Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification. Depending on the direction of communication:<br>• In direction MP -> ISOT: Entity EIC is used (determinates coding-schema=15 attribute),<br>• In direction ISOT -> MP: EIC = 24X-OT-SK------V.<br>Maximum is 16 symbols. | Required |
| Receiver Identification/id | Receiver EIC | Message receiver. Depending on the direction of communication:<br>• In direction MP -> ISOT: EIC = 24X-OT-SK------V,<br>• In direction ISOT -> MP: Entity EIC (determinates coding-scheme=15 attribute).<br>Maximum is 16 symbols.<br>It is not filled in case of E-10_02 data flow. | Optional |
| Reference/id | String | Reference message identifier for example requests on data of own orders (used in responses for message correlation). Used with notification binding about change of own orders sent through AMQP protocol (data flow E-10_01). | Optional |

*Trade*

Trade element of order, contains values of single attributes in accordance with the following table.

**Table 126** Elements of order, Trade

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | Non-negative number | Identifier of order in the system (used in responses to order). | Optional |
| version | Non-negative number | Version of order registered in the system (used in responses to order). | Optional |
| trade-day | YYYY-MM-DD | Date of trade day.<br>Format of date and time:<br>– YYYY - year,<br>– MM - month,<br>– DD - day. | Optional |
| trade-month | YYYY-MM | Date of month for which evaluation is provided. | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| order-expiration | YYYY-MM-DDTHH:MM:SS | Date and time of order expiration. If not stated it expires at the time of trade period closure.<br>Format of date and time:<br>– YYYY - year,<br>– MM - month,<br>– DD - day,<br>– HH - hour,<br>– MM - minute,<br>– SS - second. | Optional |
| block-order | A/N | Attribute for block order:<br>– A - yes,<br>– N - no. | Required |
| Block-type | BL/PL/OP/V | Type of block order used only for block orders:<br>– BL - Base Load,<br>– PL - Peak Load,<br>– OP - Off Peak Load,<br>– V - User defined block. | Optional |
| Indication | N/FOK/IOC/AON | Identification of order:<br>– N - no limitations,<br>– FOK - Fill or Kill,<br>– IOC - Immediate or Cancel,<br>– AON - All or None – only for user-defined block orders. | Required |
| trade-type | N/P | Class of order:<br>– N - buy,<br>– P - sell. | Required |
| trade-stage | N/P | Status of order in the system:<br>– N - inactive,<br>– P - active,<br>– C - partially traded,<br>– S - completely traded,<br>– E - expired,<br>– R - expired inactive,<br>– Z - cancelled. | Optional |
| trader-id | Non-negative number | Anonymous identifier of market participant. Used in data flows E-06_03 and E-10_01. | Optional |
| delivery-duration | Non-negative number | Product type (period length in minutes):<br>60 – hourly product<br>15 – quarter-hourly product | Optional |
| sett-curr | EUR | Trade currency. | Required |
| market-area | SK | Trade location (SK). | Required |
| Market | VDT | Type of market:<br>– VDT - intraday market. | Required |

| Element /Attribute | Value | Description | Use |
|---|---|---|---|
| Party/id | EIC of data owner. | Identifier of data owner for whom the data are sent. In case of owner sending data for himself, it is identical with enderIdentification.<br>Uses EIC of subject. Max. 16 symbols. | Required |
| Party/role | TO | Role of owner.<br>– TO - owner of trade. | Required |
| Comment | string | Commentary for order. | Optional |
| TimeData/ datetime | YYYY-MM-DDTHH:mm:SSZ | Order timestamp in UTC (Universal Time):<br>– YYYY - year,<br>– MM - month,<br>– DD - day,<br>– HH - hour,<br>– mm - minute,<br>– SS - second<br>(value is submitted by system). | Optional |
| TimeData/ datetime-type | DTC<br>DTR<br>DTO | Type of timestamp:<br>– DTC - date and time of order input into the system,<br>– DTR - date and time of order modification,<br>– DTO - date and time of data access<br>(value is submitted by system). | Optional |

*ProfileData*

ProfileData element of order contains values of single attributes in accordance with the following table.

**Table 127** Element of order, ProfileData

| Attribute | Value | Description | Use |
|---|---|---|---|
| trade-id | string | Identifier of trade in the system (used in responses for order description or notifications about change in order status in case the order has traded quantity).<br>Anonym ID of user-defined block order (used in responses for order book access or notifications about change in order book).<br>It serves as an identifier for user-defined block orders with same parameters. | Optional |
| profile-role | BC01<br>BP01<br>TC01<br>TP01<br>AC01<br>AC02<br>LC01<br>LP01 | Designates order and purpose of a block.<br>BC01: block containing energy quantity.<br>BP01: block containing price corresponding to energy quantity.<br>TC01: block containing amount of traded energy quantity.<br>TP01: block containing paired price (mean value in case of several prices) corresponding to amount of traded energy.<br>AC01: available cross-border transmission capacity in the direction from exporting area to | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| | | importing area (IN).<br>AC02: available cross-border transmission capacity in the opposite direction from exporting area to importing area (OUT).<br>LC01: block containing last traded quantity within the period.<br>LP01: block containing last price including price direction. | |

ProfileData is inputted always in pair. One element contains energy quantity data (BC01) and the second corresponding prices to quantity data (BP01). Except profile-role= [TC01, LC01, LP01] in case of orderbook blocks conatin information such as overall and last traded quantity including last price and price direction.

*Data*

Element of block data, *Data*, contains values for specific periods within day in relation to block type (ProfileData/@profile-role).

**Table 128** Element of block data, Data

| Attribute | Value | Description | Use |
|---|---|---|---|
| period-from | Non-negative number | Designates beginning of term for which the order is entered.<br>0…24 for 60-minutes products,<br>0...96 for 15-minutes products.<br><br>During transition from CET to CEST and back, there are:<br>23, or 25 periods for 60-minutes products,<br>92 or 100 periods for 15-minutes products. | Optional |
| period-to | Non-negative number | Designates end of term for which the order is entered.<br>1…25 for 60-minutes products,<br>0...96 for 15-minutes products.<br><br>During transition from CET to CEST and back, there are:<br>23, or 25 periods for 60-minutes products,<br>92 or 100 periods for 15-minutes products. | Optional |
| value | Decimal number | Contains quantity/price. Quantity with one decimal position precision.<br>Price with two decimal position precision.<br>Separator of decimal position is “.“ (dot). | Required |
| unit | MW, EUR | Unit of entered value. | Required |
| seq-num | Non-negative number | Sequence number of records about quantity for given price and period. Used within message for order book data access. (data flow E-08_01 and E-10_02). | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| price-direction | N, I, D | Price direction:<br>N=stagnant,<br>I=increasing,<br>D=decreasing. | Optional |

### 4.1.3 RESPONSE

RESPONSE structure, derived from the IS OTE system, is used in communication scenarios for confirmation of transaction success during data exchange in *ISOTEDATA* structures.

The so-called message type (message-code attribute) determines the meaning of content, similarly to ISOTEDATA. Due to clarity, specific structure types are further used in *RESPONSE.message-code* format.

## *RESPONSE structure*

Response is represented by *RESPONSE* structure consisting of the following parts:

- *RESPONSE* - contains general details relating to the entire message (message header),
- *Reason* – represents reason/response of request processing.

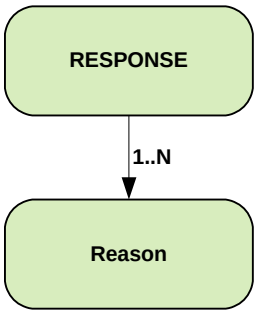


**Figure 16** Scheme of RESPONSE structure

### *RESPONSE*

RESPONSE root element contains attribute values according to the following table.

**Table 129** RESPONSE root element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message-code | number/code of the message | Determines the purpose of structure content:<br>– 812 - error/confirmation at submission/replacement of order on DM (E-02_01),<br>– 832 - error/confirmation at sending of request for retrieval of order status on DM (E-02_03),<br>– 942 - error/confirmation at sending of request for data – DM results (E-03_02),<br>– 952 - error/confirmation at sending of request for data – evaluation per hours (E-05_01),<br>– 962 - error/confirmation at sending of request for data – evaluation for day (E-05_02). | Required |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>– YYYY - year,<br>– MM - month,<br>– DD - day,<br>– HH - hour,<br>– mm - minute,<br>– SS - second. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 130** Message header of displayed values of particular elements

| Element /Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | 24X-OT-SK------V | Message sender identification.<br>EIC = 24X-OT-SK------V is used. | Required |
| Receiver Identification/id | Receiver EIC | Message receiver identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Reference/id | String | Identification of related message, for which the response is given. | Optional |

*REASON*

Reason element contains attribute values according to the following table.

**Table 131** Reason element

| Attribute | Value | Description | Use |
|---|---|---|---|
| code | String | Detailed reason/code of response:<br>• -1 – Nonexistence of details,<br>• 0 – Not specified,<br>• 1 – Noncompliance with ascending/descending price development of blocks,<br>• 2 - Maximal number of blocks exceeded,<br>• 3 – Noncompliance with condition for divisibility of the first block,<br>• 4 – Noncompliance with permitted minimal and maximal amount,<br>• 5 - Noncompliance with permitted minimal and maximal price,<br>• 6 – Noncompliance with requested resolution,<br>• 7 – Noncompliance with input of amount and price in at least one hour of block,<br>• 8 – Noncompliance with paired input of details,<br>• 9 – Insufficient financial security towards clearing agent,<br>• 10 – Insufficient financial security towards market organizer. | Required |
| type | AXY | Type of response:<br>• A01 – Rejection due to syntax error,<br>• A02 – Rejection due to application reasons,<br>• A03 – Acceptance without reservation,<br>• A04 – Acceptance with reservation | Required |
| trade-id | Non-negative number | Order identification registered in the system. | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| version | Non-negative number | Order version registered in the system. | Optional |

### 4.1.4 RESPONSE-VDT

RESPONSE-VDT structure, derived from the IS OTE system, is used in communication scenarios for confirmation of transaction success during data exchange in *ISOTEDATA-VDT* structures.

The so-called message type (message-code attribute) determines the meaning of content, similarly to ISOTEDATA-VDT. Due to clarity, specific structure types are in *RESPONSE-VDT.message-code* format.

#### *RESPONSE-VDT structure*

Response is represented by *RESPONSE-VDT* structure consisting of the following parts:

- *RESPONSE* - contains general details relating to the entire message (message header),
- *Reason* – represents reason/response of request processing.

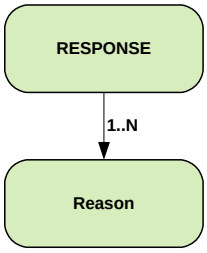


**Figure 17** Scheme of RESPONSE-VDT structure

*RESPONSE*

RESPONSE root element contains attribute values in accordance with the following table.

**Table 132** Root element RESPONSE

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |
| message-code | number/code of message | Determines the purpose of structure content:<br>• 572 – error/confirmation after data request – results for month (E-07_02),<br>• 802 – error/confirmation after order submission to IDM (E-06_01),<br>• 805 – error/confirmation after IDM order modification (E-06_02),<br>• 808 – error/confirmation after IDM order data provision request (E-06_03),<br>• 811 – error/confirmation after order book provision request (E-08_01),<br>• 952 – error/confirmation after results per periods data request (E-07_03),<br>• 962 – error/confirmation after results for day data request (E-07_01). | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• mm – minute,<br>• SS – second. | Required |

Message header contains values of particular elements in accordance with the following table.

**Table 133** Message header of displayed values of particular elements

| Element /Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | 24X-OT-SK------V | Message sender identification.<br>EIC = 24X-OT-SK------V is used. | Required |
| Receiver Identification/id | Receiver EIC | Message receiver identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Reference/id | String | Identification of related message, for which the response is given. | Optional |

REASON

Reason element contains attribute values according to the following table.

**Table 134** Reason element

| Attribute | Value | Description | Use |
|---|---|---|---|
| code | String | Detailed reason/code of response:<br>• -1 – Nonexistence of details,<br>• 0 – Not specified,<br>• 4 – Noncompliance with permitted minimal and maximal amount,<br>• 5 - Noncompliance with permitted minimal and maximal price,<br>• 8 – Noncompliance with paired input of details,<br>• 9 – Insufficient financial security towards clearing agent,<br>• 10 – Insufficient financial security towards market organizer,<br>• 11- Incorrect combination of order parameters,<br>• 12- Validity of order after given time period closure,<br>• 13-Order submission for close trade period. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| type | AXY | Type of response:<br>• A01 – Rejection due to syntax error,<br>• A02 – Rejection due to application reasons,<br>• A03 – Acceptance without reservation,<br>• A04 – Acceptance with reservation. | Required |
| trade-id | Non-negative number | Order identification registered in the system. | Optional |
| version | Non-negative number | Order version registered in the system. | Optional |

### 4.1.5 CDSREQ

CDSREQ structure, derived from the IS OTE system, is used in communication scenarios for data retrieval from the ISOT system in *ISOTEDATA* structures.

The meaning of content, i.e. the type of requested data is determined by so-called message type (*message-code* attribute) that is further used in the text in *CDSREQ.message-code* format due to clarity.

#### *CDSREQ structure*

Data request represented by *CDSREQ* structure consists of the following parts:

- *CDSREQ* - contains general details relating to the entire message (message header),
- *Trade* – identification of requested data.

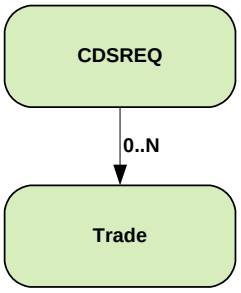


**Figure 18** Scheme of CDSREQ structure

*CDSREQ*

CDSREQ root element contains attribute values according to the following table.

**Table 135** CDSREQ root element

| Attribute | Value | Description | Use |
|---|---|---|---|
| Id | string | Unique message identifier within the sender system.<br>Maximum 35 characters. | Required |

| Attribute | Value | Description | Use |
|---|---|---|---|
| message-code | number/code of the message | It determines the purpose of structure content:<br>• 831 – Request for data on own order DM/IDA (retrieval of status),<br>• 941 – Request for DM/IDA results for subject of settlement,<br>• 951 – Request for DM/IDA evaluation per ~~hours~~periods,<br>• 961 – Request for DM/IDA evaluation for day.<br>• The requests are generic and are sent to various URLs of independent web services. | Required |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• mm – minute,<br>• SS – second. | Required |

Message header contains values of particular elements according to the following table.

**Table 136** Message header of displayed values of particular elements

| Element /Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Receiver Identification/id | 24X-OT-SK------V | Message receiver identification.<br>EIC = 24X-OT-SK------V is used. | Required |

*Trade*

Trade element contains attribute values according to the following table.

**Table 137** Trade header element

| Attribute | Value | Description | Use |
|---|---|---|---|
| Id | non-negative number | Order identification in the system. | Optional |
| Version | non-negative number | Order version registered in the system. | Optional |

| Attribute | Value | Description | Use |
|---|---|---|---|
| trade-day | YYYY-MM-DD | Date of trading day.<br>Date and time format:<br>• YYYY – year,<br>• MM – month,<br>• DD – day. | Optional |
| delivery-duration | 15/60 | Trading period duration:<br>15 – quarter-hour (15 min.)<br>60 – hour (60 min.) | Optional |
| auction-id | string | IDA auction identifier:<br>• IDA1 – first intraday auction for specific trading day<br>• IDA2 – secpnd intraday auction for specific trading day<br>• IDA3 – third intraday auction for specific trading day | Optional |

### 4.1.6 CDSREQ-VDT

CDSREQ structure, derived from the IS OTE system and further expanded for intraday continuous trading, is used in communication scenarios for data retrieval from the ISOT system in *ISOTEDATA-VDT* structures.

The meaning of content, i.e. the type of requested data is determined by so-called message type (*message-code* attribute) that is further used in the text in *CDSREQ-VDT.message-code* format due to clarity.

#### *CDSREQ structure*

Data request represented by *CDSREQ-VDT* structure consists of the following parts:

- *CDSREQ* - contains general details relating to the entire message (message header),
- *Trade* – identification of requested data.

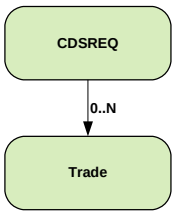


**Figure 19** Scheme of CDSRWQ-VDT structure

*CDSREQ*

CDSREQ-VDT root element contains attribute values according to the following table.

**Table 138** CDSREQ root element

| Attribute | Value | Description | Use |
|---|---|---|---|

| Attribute | Value | Description | Use |
|---|---|---|---|
| Id | string | Unique message identifier within the sender system.<br><br>Maximum 35 characters. | Required |
| message-code | number/code of the message | Determines the purpose of structure content:<br>• 571 - Request for IDM evaluation for month (E-07_02),<br>• 807 - Request for order data access,<br>• 810 - Request for order book access (E-08_01),<br>• 951 – Request for IDM evaluation per periods (E-07_03),<br>• 961 – Request for IDM evaluation for day (E-07_01). | Required |
| date-time | YYYY-MM-DDTHH:mm:SSZ | Date and time of message sending in UTC (Universal Time):<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• mm – minute,<br>• SS – second. | Required |

Message header contains values of particular elements according to the following table.

**Table 139** Message header of displayed values of particular elements

| Element /Attribute | Value | Description | Use |
|---|---|---|---|
| Sender Identification/id | Sender EIC | Message sender identification.<br>Entity EIC is used (it determines *coding-scheme=15* attribute).<br>Maximum 16 characters. | Required |
| Receiver Identification/id | 24X-OT-SK------V | Message receiver identification.<br>EIC = 24X-OT-SK------V is used. | Required |

*Trade*

Element Trade contains values of particular elements according to the following table.

**Table 140** Trade header element

| Attribute | Value | Description | Use |
|---|---|---|---|
| id | non-negative number | Order identification on the system | Optional |
| version | non-negative number | Order version registered in the system | Optional |
| trade-day | YYYY-MM-DD | Date of trading day.<br>Date and time format:<br>• YYYY – year,<br>• MM - month,<br>• DD – day. | Optional |
| trade-month | YYYY-MM | Date of trading month.<br>Date and time format:<br>• YYYY – year,<br>• MM – month. | Optional |
| period-from | non-negative number | Designates the beginning of the term for which the data are accessed.<br>0…24 for 60-minutes products,<br>0...96 for 15-minutes products.<br><br>During transition from CET to CEST and back, there are:<br>23, or 25 periods for 60-minutes products,<br>92 or 100 periods for 15-minutes products. | Optional |
| period-to | non-negative number | Designates the end of the term for which the data are accessed.<br>1…25 (during transition from CET to CEST and back, there are 23, or 25 periods). | Optional |
| delivery-duration | non-negative number | Product type (period length in minutes):<br>60 – hourly product<br>15 – quarter-horuly product | Optional |

## 4.2 Administration of DAM orders

Market participant DAM order administration is carried out through operations of order reception and availability registered in ISOT via Orders web service.

### 4.2.1 Processing level

Market participants enter their orders into ISOT by the deadline for order registration at the latest (during D-1). Orders can be entered into the system in advance, even several days in advance. It is possible to place orders with a 60-minute trading interval or a 15-minute trading interval.

Market participant can register unlimited number of sale orders and unlimited number of purchase orders for a single trading day. Order can contain following characteristics:

- Standard order in 60 minutes (order containing maximum of 25 blocks without the possibility to define total acceptance of block no. 1) or an order in 15 minutes (an order containing up to 25 blocks without the possibility of defining the total acceptance of the 1st block). This order type is defined by element *block-order="N".*
- Block order with one of the following types:
    - simple profile block order with possibility to fill up one block containing the quantity for maximum of 24 (23/25 in case of clock change day) periods for an order in 60 minutes or 96 (92/100 on transition days) trading periods for an order in 15 minutes with definition of one weighted-average price, defined by *block-order="A" a block-type="SB",*
    - linked block order in 60 minutes or 15 minutes with possibility to defined parent simple block order, which must be accepted in order to trade the linked block order, defined by *block-order="A", block-type="LB"* and *linked-order-id="nnnnnn",*
    - flexible block order in 60 minutes or 15 minutes with definition of price and quantity for one trading period, which will be selected by the matching algorithm according to matching rules, defined by *block-order="A"* and *block-type="FB",*
    - exclusive group of block orders in 60 minutes or 15 minutes with possibility to submit at least 2 and maximum 8 simple block orders, where only one order may be accepted according to matching rules, defined by *block-order="A"* and *block-type="EG"*. Relevant orders within the group are submitted as separate blocks within the order.

In the case that market participant, using the modification of existing order, replaces already existing order, or removes the order from the system, it is not allowed to return to the previous version of the order i.e. not even in case that the original order was valid and the new order is invalid.

In case of invalid order, user is required to remove reasons for its invalidity by replacement with a new version or by deleting the existing order and creation of new one respectively. Orders non-compliant with the check at submission (invalid) shall be removed and will not enter the matching process.

Removal of already entered and accepted orders is carried out through submission of a new order with null values in the first block (paired values of amount/price).

### 4.2.2 Reception of orders (E-02_01)

Reception of orders is carried out by request for order submission in *ISOTEDATA.811* structure (message-code=811) and by response in *RESPONSE.812* structure (indication of success/fail) and *ISOTEDATA.813* structure (description of order registered in the system).

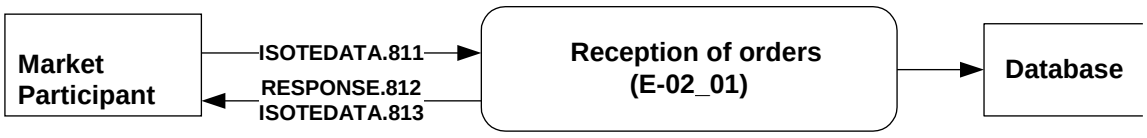


**Figure 20** Scheme of DAM order reception

*ISOTEDATA.811*

The structure contains message-code=811 attribute in the header and is filled out in accordance with order structure. Only a single order can be concurrently entered, i.e. submission of multiple orders is carried out through multiple calls.

```
<ISOTEDATA id="1" message-code="811" date-time="2009-06-20T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2009-09-21" trade-type="P" block-order="N" market-area="SK" sett-
curr="EUR" market="DAM" delivery-duration="60">
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MWH" splitting="A" />
      <Data period="2" value="100.0" unit="MWH" splitting="A" />
      <Data period="3" value="100.0" unit="MWH" splitting="A" />
      <Data period="4" value="100.0" unit="MWH" splitting="A" />
      <Data period="5" value="100.0" unit="MWH" splitting="A" />
      <Data period="6" value="100.0" unit="MWH" splitting="N" />
      <Data period="7" value="100.0" unit="MWH" splitting="N" />
      <Data period="8" value="100.0" unit="MWH" splitting="N" />
      <Data period="9" value="100.0" unit="MWH" splitting="N" />
      <Data period="10" value="100.0" unit="MWH" splitting="N" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A" />
      <Data period="2" value="15.00" unit="EUR" splitting="A" />
      <Data period="3" value="15.00" unit="EUR" splitting="A" />
      <Data period="4" value="15.00" unit="EUR" splitting="A" />
      <Data period="5" value="15.00" unit="EUR" splitting="A" />
      <Data period="6" value="15.00" unit="EUR" splitting="N" />
      <Data period="7" value="15.00" unit="EUR" splitting="N" />
      <Data period="8" value="15.00" unit="EUR" splitting="N" />
      <Data period="9" value="15.00" unit="EUR" splitting="N" />
      <Data period="10" value="15.00" unit="EUR" splitting="N" />
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 1** Submission of standard sale order for a 60 minute interval

```
<ISOTEDATA id="1" message-code="811" date-time="2009-06-20T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2009-09-21" trade-type="P" block-order="A" block-type="SB" market-
area="SK" sett-curr="EUR" market="DAM" delivery-duration="60" >
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MWH" splitting="A" />
      <Data period="2" value="100.0" unit="MWH" splitting="A" />
      <Data period="3" value="100.0" unit="MWH" splitting="A" />
      <Data period="4" value="100.0" unit="MWH" splitting="A" />
      <Data period="5" value="100.0" unit="MWH" splitting="A" />
      <Data period="6" value="100.0" unit="MWH" splitting="N" />
      <Data period="7" value="100.0" unit="MWH" splitting="N" />
      <Data period="8" value="100.0" unit="MWH" splitting="N" />
      <Data period="9" value="100.0" unit="MWH" splitting="N" />
      <Data period="10" value="100.0" unit="MWH" splitting="N" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A" />
      <Data period="2" value="15.00" unit="EUR" splitting="A" />
      <Data period="3" value="15.00" unit="EUR" splitting="A" />
      <Data period="4" value="15.00" unit="EUR" splitting="A" />
      <Data period="5" value="15.00" unit="EUR" splitting="A" />
      <Data period="6" value="15.00" unit="EUR" splitting="N" />
      <Data period="7" value="15.00" unit="EUR" splitting="N" />
      <Data period="8" value="15.00" unit="EUR" splitting="N" />
      <Data period="9" value="15.00" unit="EUR" splitting="N" />
      <Data period="10" value="15.00" unit="EUR" splitting="N" />
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
```

**Example 2** Submission of simple block sale order for a 60 minute interval

*RESPONSE.812*

In case of successful or failed order processing, response is returned in accordance with specification of RESPONSE structure, with *message-code=812* in the header. Identification, under which the order is registered in the system, is returned in *Reason/@trade-id* attribute.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03" trade-id="1016"/>
</RESPONSE>
```

**Example 3** Response on success of order submission

*ISOTEDATA.813*

In case of successful order processing, the structure is returned as it was registered in the system, where *message-code=813* can be found in the header. Identification and version, under which the order is registered in the system, are returned in *Trade/@id* and *Trade/@version* attribute. Own order data can be retrieved back based on this identification (see E-02_03).

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="813"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade id="1016" trade-day="2009-09-21" version="1" trade-type="P" block-order="N" trade-
stage="P" sett-curr="EUR" market-area="SK" market="DAM" delivery-duration="60">
    <TimeData datetime="2009-07-03T13:46:26Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MWH" splitting="A"/>
      <Data period="2" value="100.0" unit="MWH" splitting="A"/>
      <Data period="3" value="100.0" unit="MWH" splitting="A"/>
      <Data period="4" value="100.0" unit="MWH" splitting="A"/>
      <Data period="5" value="100.0" unit="MWH" splitting="A"/>
      <Data period="6" value="100.0" unit="MWH" splitting="N"/>
      <Data period="7" value="100.0" unit="MWH" splitting="N"/>
      <Data period="8" value="100.0" unit="MWH" splitting="N"/>
      <Data period="9" value="100.0" unit="MWH" splitting="N"/>
      <Data period="10" value="100.0" unit="MWH" splitting="N"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A"/>
      <Data period="2" value="15.00" unit="EUR" splitting="A"/>
      <Data period="3" value="15.00" unit="EUR" splitting="A"/>
      <Data period="4" value="15.00" unit="EUR" splitting="A"/>
      <Data period="5" value="15.00" unit="EUR" splitting="A"/>
      <Data period="6" value="15.00" unit="EUR" splitting="N"/>
      <Data period="7" value="15.00" unit="EUR" splitting="N"/>
      <Data period="8" value="15.00" unit="EUR" splitting="N"/>
      <Data period="9" value="15.00" unit="EUR" splitting="N"/>
      <Data period="10" value="15.00" unit="EUR" splitting="N"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 4** Response with description of entered order in the system

### 4.2.3 Removal of orders (E-02_01)

Removal of order is carried out by submission of specific order in *ISOTEDATA.811* structure (message-code=811) containing null values for amount and price in the first block for all periods of a given trading day. Response to order removal is returned in *RESPONSE.812* structure (indication of

success/failure) and *ISOTEDATA.813* structure (description of order that was removed from the system).

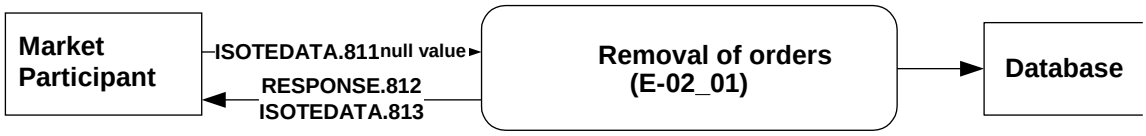


**Figure 21** Scheme of market participant order removal

*ISOTEDATA.811*

The structure contains *message-code=811* attribute in the header and is filled out in accordance with order structure specification, where only the 1. block is entered containing null values for amount and price. Request can be entered for removal of orders for specific trading day (*trade-day* attribute), removal of specific trading day and order type (*trade-day* and *trade-type* attributes) or removal of specific order via order identification of relevant order (*id* attribute returned in responses at order submission).

```
<ISOTEDATA id="1" message-code="811" date-time="2009-06-20T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade id="1016" trade-day="2009-09-21" trade-type="P" market-area="SK" sett-curr="EUR"
market="DAM" delivery-duration="60">
    <ProfileData profile-role="BC01">
      <Data period="1" value="0.0" unit="MWH" splitting="A"/>
      <Data period="2" value="0.0" unit="MWH" splitting="A"/>
      <Data period="3" value="0.0" unit="MWH" splitting="A"/>
      <Data period="4" value="0.0" unit="MWH" splitting="A"/>
      <Data period="5" value="0.0" unit="MWH" splitting="A"/>
      <Data period="6" value="0.0" unit="MWH" splitting="A"/>
      <Data period="7" value="0.0" unit="MWH" splitting="A"/>
      <Data period="8" value="0.0" unit="MWH" splitting="A"/>
      <Data period="9" value="0.0" unit="MWH" splitting="A"/>
      <Data period="10" value="0.0" unit="MWH" splitting="A"/>
      <Data period="11" value="0.0" unit="MWH" splitting="A"/>
      <Data period="12" value="0.0" unit="MWH" splitting="A"/>
      <Data period="13" value="0.0" unit="MWH" splitting="A"/>
      <Data period="14" value="0.0" unit="MWH" splitting="A"/>
      <Data period="15" value="0.0" unit="MWH" splitting="A"/>
      <Data period="16" value="0.0" unit="MWH" splitting="A"/>
      <Data period="17" value="0.0" unit="MWH" splitting="A"/>
      <Data period="18" value="0.0" unit="MWH" splitting="A"/>
      <Data period="19" value="0.0" unit="MWH" splitting="A"/>
      <Data period="20" value="0.0" unit="MWH" splitting="A"/>
      <Data period="21" value="0.0" unit="MWH" splitting="A"/>
      <Data period="22" value="0.0" unit="MWH" splitting="A"/>
      <Data period="23" value="0.0" unit="MWH" splitting="A"/>
      <Data period="24" value="0.0" unit="MWH" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="0.0" unit="EUR" splitting="A"/>
      <Data period="2" value="0.0" unit="EUR" splitting="A"/>
      <Data period="3" value="0.0" unit="EUR" splitting="A"/>
      <Data period="4" value="0.0" unit="EUR" splitting="A"/>
      <Data period="5" value="0.0" unit="EUR" splitting="A"/>
      <Data period="6" value="0.0" unit="EUR" splitting="A"/>
      <Data period="7" value="0.0" unit="EUR" splitting="A"/>
      <Data period="8" value="0.0" unit="EUR" splitting="A"/>
      <Data period="9" value="0.0" unit="EUR" splitting="A"/>
      <Data period="10" value="0.0" unit="EUR" splitting="A"/>
      <Data period="11" value="0.0" unit="EUR" splitting="A"/>
      <Data period="12" value="0.0" unit="EUR" splitting="A"/>
      <Data period="13" value="0.0" unit="EUR" splitting="A"/>
      <Data period="14" value="0.0" unit="EUR" splitting="A"/>
      <Data period="15" value="0.0" unit="EUR" splitting="A"/>
      <Data period="16" value="0.0" unit="EUR" splitting="A"/>
      <Data period="17" value="0.0" unit="EUR" splitting="A"/>
      <Data period="18" value="0.0" unit="EUR" splitting="A"/>
      <Data period="19" value="0.0" unit="EUR" splitting="A"/>
      <Data period="20" value="0.0" unit="EUR" splitting="A"/>
      <Data period="21" value="0.0" unit="EUR" splitting="A"/>
      <Data period="22" value="0.0" unit="EUR" splitting="A"/>
      <Data period="23" value="0.0" unit="EUR" splitting="A"/>
      <Data period="24" value="0.0" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 5** Removal of specific sale order

*RESPONSE.812*

In accordance with specification of RESPONSE structure, response is given in case of successful or unsuccessful order processing. In this case message-code=812 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 6** Response on success of order removal

*ISOTEDATA.813*

In case of successful order removal, the original order that was removed from the system will be returned. Message-code=813 can be found in the header.

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="813"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade id="1016" trade-day="2009-09-21" version="1" trade-type="P" block-order="N" trade-
stage="P" sett-curr="EUR" market-area="SK" market="DAM" delivery-duration="60">
    <TimeData datetime="2009-07-03T13:46:26Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MWH" splitting="A"/>
      <Data period="2" value="100.0" unit="MWH" splitting="A"/>
      <Data period="3" value="100.0" unit="MWH" splitting="A"/>
      <Data period="4" value="100.0" unit="MWH" splitting="A"/>
      <Data period="5" value="100.0" unit="MWH" splitting="A"/>
      <Data period="6" value="100.0" unit="MWH" splitting="N"/>
      <Data period="7" value="100.0" unit="MWH" splitting="N"/>
      <Data period="8" value="100.0" unit="MWH" splitting="N"/>
      <Data period="9" value="100.0" unit="MWH" splitting="N"/>
      <Data period="10" value="100.0" unit="MWH" splitting="N"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A"/>
      <Data period="2" value="15.00" unit="EUR" splitting="A"/>
      <Data period="3" value="15.00" unit="EUR" splitting="A"/>
      <Data period="4" value="15.00" unit="EUR" splitting="A"/>
      <Data period="5" value="15.00" unit="EUR" splitting="A"/>
      <Data period="6" value="15.00" unit="EUR" splitting="N"/>
      <Data period="7" value="15.00" unit="EUR" splitting="N"/>
      <Data period="8" value="15.00" unit="EUR" splitting="N"/>
      <Data period="9" value="15.00" unit="EUR" splitting="N"/>
      <Data period="10" value="15.00" unit="EUR" splitting="N"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 7** Response with description of removed order from the system

### 4.2.4 Order modification (E-02_01)

Modification of order is carried out by entry of specific order that is to be modified in *ISOTEDATA.811* structure (message-code=811) containing updated values for amount and price for desired trading periods of a given trading day. Response to order removal is returned in *RESPONSE.812* structure (indication of success/failure) and *ISOTEDATA.813* structure (description of modified order saved in the system). Order header modification is not possible. In order to modify the order header (i.e. type or direction), delete the existing order and submit a new one with required parameters.

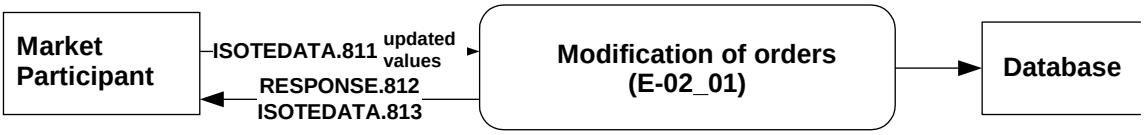


**Figure 22** Scheme of market participant order modification

*ISOTEDATA.811*

The structure contains *message-code=811* attribute in the header and is filled out in accordance with order structure specification. The order that is to be modified must be identified by specification of its ID in the attribute *Trade/id*. Only a single order can be concurrently modified, i.e. modification of multiple orders is carried out through multiple calls.

```
<ISOTEDATA id="1" message-code="811" date-time="2009-06-20T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade id="1016" trade-day="2009-09-21" trade-type="P" block-order="N" market-area="SK" sett-
curr="EUR" market="DAM" delivery-duration="60">
    <ProfileData profile-role="BC01">
      <Data period="1" value="10.0" unit="MWH" splitting="A"/>
      <Data period="2" value="10.0" unit="MWH" splitting="A"/>
      <Data period="3" value="10.0" unit="MWH" splitting="A"/>
      <Data period="4" value="10.0" unit="MWH" splitting="A"/>
      <Data period="5" value="20.0" unit="MWH" splitting="A"/>
      <Data period="6" value="20.0" unit="MWH" splitting="A"/>
      <Data period="7" value="20.0" unit="MWH" splitting="A"/>
      <Data period="8" value="10.0" unit="MWH" splitting="A"/>
      <Data period="9" value="10.0" unit="MWH" splitting="A"/>
      <Data period="10" value="10.0" unit="MWH" splitting="A"/>
      <Data period="11" value="10.0" unit="MWH" splitting="A"/>
      <Data period="12" value="10.0" unit="MWH" splitting="A"/>
      <Data period="13" value="10.0" unit="MWH" splitting="A"/>
      <Data period="14" value="10.0" unit="MWH" splitting="A"/>
      <Data period="15" value="20.0" unit="MWH" splitting="A"/>
      <Data period="16" value="20.0" unit="MWH" splitting="A"/>
      <Data period="17" value="20.0" unit="MWH" splitting="A"/>
      <Data period="18" value="20.0" unit="MWH" splitting="A"/>
      <Data period="19" value="20.0" unit="MWH" splitting="A"/>
      <Data period="20" value="20.0" unit="MWH" splitting="A"/>
      <Data period="21" value="10.0" unit="MWH" splitting="A"/>
      <Data period="22" value="10.0" unit="MWH" splitting="A"/>
      <Data period="23" value="10.0" unit="MWH" splitting="A"/>
      <Data period="24" value="10.0" unit="MWH" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="50.0" unit="EUR" splitting="A"/>
      <Data period="2" value="50.0" unit="EUR" splitting="A"/>
      <Data period="3" value="50.0" unit="EUR" splitting="A"/>
      <Data period="4" value="50.0" unit="EUR" splitting="A"/>
      <Data period="5" value="50.0" unit="EUR" splitting="A"/>
      <Data period="6" value="50.0" unit="EUR" splitting="A"/>
      <Data period="7" value="50.0" unit="EUR" splitting="A"/>
      <Data period="8" value="50.0" unit="EUR" splitting="A"/>
      <Data period="9" value="50.0" unit="EUR" splitting="A"/>
      <Data period="10" value="50.0" unit="EUR" splitting="A"/>
      <Data period="11" value="50.0" unit="EUR" splitting="A"/>
      <Data period="12" value="50.0" unit="EUR" splitting="A"/>
      <Data period="13" value="50.0" unit="EUR" splitting="A"/>
      <Data period="14" value="50.0" unit="EUR" splitting="A"/>
      <Data period="15" value="50.0" unit="EUR" splitting="A"/>
      <Data period="16" value="50.0" unit="EUR" splitting="A"/>
      <Data period="17" value="50.0" unit="EUR" splitting="A"/>
      <Data period="18" value="50.0" unit="EUR" splitting="A"/>
      <Data period="19" value="50.0" unit="EUR" splitting="A"/>
      <Data period="20" value="50.0" unit="EUR" splitting="A"/>
      <Data period="21" value="50.0" unit="EUR" splitting="A"/>
      <Data period="22" value="50.0" unit="EUR" splitting="A"/>
      <Data period="23" value="50.0" unit="EUR" splitting="A"/>
      <Data period="24" value="50.0" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 8** Modification of specific order

*RESPONSE.812*

In accordance with specification of RESPONSE structure, response is given in case of successful or unsuccessful order processing. In this case message-code=812 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 9** Response on success of order modification

*ISOTEDATA.813*

In case of successful order modification, the succesfully modified order will be returned with increased version in comparison to original order. Message-code=813 can be found in the header.

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="813"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade id="1016" trade-day="2009-09-21" version="2" trade-type="P" block-order="N" trade-
stage="P" sett-curr="EUR" market-area="SK" market="DAM" delivery-duration="60">
    <TimeData datetime="2009-07-03T13:46:26Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="10.0" unit="MWH" splitting="A"/>
      <Data period="2" value="10.0" unit="MWH" splitting="A"/>
      <Data period="3" value="10.0" unit="MWH" splitting="A"/>
      <Data period="4" value="10.0" unit="MWH" splitting="A"/>
      <Data period="5" value="20.0" unit="MWH" splitting="A"/>
      <Data period="6" value="20.0" unit="MWH" splitting="A"/>
      <Data period="7" value="20.0" unit="MWH" splitting="A"/>
      <Data period="8" value="10.0" unit="MWH" splitting="A"/>
      <Data period="9" value="10.0" unit="MWH" splitting="A"/>
      <Data period="10" value="10.0" unit="MWH" splitting="A"/>
      <Data period="11" value="10.0" unit="MWH" splitting="A"/>
      <Data period="12" value="10.0" unit="MWH" splitting="A"/>
      <Data period="13" value="10.0" unit="MWH" splitting="A"/>
      <Data period="14" value="10.0" unit="MWH" splitting="A"/>
      <Data period="15" value="20.0" unit="MWH" splitting="A"/>
      <Data period="16" value="20.0" unit="MWH" splitting="A"/>
      <Data period="17" value="20.0" unit="MWH" splitting="A"/>
      <Data period="18" value="20.0" unit="MWH" splitting="A"/>
      <Data period="19" value="20.0" unit="MWH" splitting="A"/>
      <Data period="20" value="20.0" unit="MWH" splitting="A"/>
      <Data period="21" value="10.0" unit="MWH" splitting="A"/>
      <Data period="22" value="10.0" unit="MWH" splitting="A"/>
      <Data period="23" value="10.0" unit="MWH" splitting="A"/>
      <Data period="24" value="10.0" unit="MWH" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="50.0" unit="EUR" splitting="A"/>
      <Data period="2" value="50.0" unit="EUR" splitting="A"/>
      <Data period="3" value="50.0" unit="EUR" splitting="A"/>
      <Data period="4" value="50.0" unit="EUR" splitting="A"/>
      <Data period="5" value="50.0" unit="EUR" splitting="A"/>
      <Data period="6" value="50.0" unit="EUR" splitting="A"/>
      <Data period="7" value="50.0" unit="EUR" splitting="A"/>
      <Data period="8" value="50.0" unit="EUR" splitting="A"/>
      <Data period="9" value="50.0" unit="EUR" splitting="A"/>
      <Data period="10" value="50.0" unit="EUR" splitting="A"/>
      <Data period="11" value="50.0" unit="EUR" splitting="A"/>
      <Data period="12" value="50.0" unit="EUR" splitting="A"/>
      <Data period="13" value="50.0" unit="EUR" splitting="A"/>
      <Data period="14" value="50.0" unit="EUR" splitting="A"/>
      <Data period="15" value="50.0" unit="EUR" splitting="A"/>
      <Data period="16" value="50.0" unit="EUR" splitting="A"/>
      <Data period="17" value="50.0" unit="EUR" splitting="A"/>
      <Data period="18" value="50.0" unit="EUR" splitting="A"/>
      <Data period="19" value="50.0" unit="EUR" splitting="A"/>
      <Data period="20" value="50.0" unit="EUR" splitting="A"/>
      <Data period="21" value="50.0" unit="EUR" splitting="A"/>
      <Data period="22" value="50.0" unit="EUR" splitting="A"/>
      <Data period="23" value="50.0" unit="EUR" splitting="A"/>
      <Data period="24" value="50.0" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 10** Response with description of modified order

### 4.2.5 Provision of orders (E-02_03)

Provision of orders is carried out by request for order retrieval in *CDSREQ.831* structure (message-code=831) and response in *RESPONSE.832* structure (indication of success/failure) and ISOTEDATA.833 *structure* (description of order registered in the system).


**Figure 23** Scheme of market participant order provision

*CDSREQ.831*

Request can be formulated either for specific trading day (trade-day) or for specific order (id and version) and is filled out according to the specification of CDSREQ structure. Order identification has precedence over trading day.

```
<CDSREQ id="4a6s5d45f" message-code="831" date-time="2014-09-19T01:18:33" dtd-version="1"
dtd-release="1" xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="11XKORLEAINVESTY" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2009-09-21"/>
</CDSREQ>
```

**Example 11** Request for retrieval of orders for specific day

*RESPONSE.832*

In case of successful or failed request processing, response is returned in accordance with specification of RESPONSE structure, where message-code=832 can be found in the header.

```
<RESPONSE id="a9e40366-ad70-45ac-8b36-bd8fbce5ef7" message-code="832"
       date-time="2009-07-03T14:02:36Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="11XKORLEAINVESTY" coding-scheme="15" />
  <Reference id="4a6s5d45f"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 12** Response on success of order provision for specific day

*ISOTEDATA.833*

In case of successful request processing, the structure is returned as it was registered in the system and message-code=833 can be found in the header. Identification and version, under which the order is registered in the system, are returned in *Trade/@id* and *Trade/@version* attribute. If there is a single purchase order and a single sale order registered in the system for a given trading day, return structure of request for specific trading day contains two orders (either valid or invalid).

```
<ISOTEDATA id="1" message-code="833" date-time="2009-07-03T14:02:36Z"
       dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="11XKORLEAINVESTY" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Reference id="4a6s5d45f"/>
  <Trade id="977" trade-day="2009-09-21" trade-type="P" block-order="N" trade-stage="P"
       sett-curr="EUR" market-area="SK" market="DAM" delivery-duration="60">
    <TimeData datetime="2009-07-02T09:00:43Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="5.0" unit="MWH" splitting="A"/>
      <Data period="2" value="5.0" unit="MWH" splitting="A"/>
      <Data period="3" value="5.0" unit="MWH" splitting="A"/>
      <Data period="4" value="5.0" unit="MWH" splitting="A"/>
      <Data period="5" value="5.0" unit="MWH" splitting="A"/>
      <Data period="6" value="5.0" unit="MWH" splitting="A"/>
      <Data period="7" value="5.0" unit="MWH" splitting="A"/>
      <Data period="8" value="5.0" unit="MWH" splitting="A"/>
      <Data period="9" value="5.0" unit="MWH" splitting="A"/>
      <Data period="10" value="5.0" unit="MWH" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="24.00" unit="EUR" splitting="A"/>
      <Data period="2" value="24.00" unit="EUR" splitting="A"/>
      <Data period="3" value="24.00" unit="EUR" splitting="A"/>
      <Data period="4" value="24.00" unit="EUR" splitting="A"/>
      <Data period="5" value="24.00" unit="EUR" splitting="A"/>
      <Data period="6" value="24.00" unit="EUR" splitting="A"/>
      <Data period="7" value="24.00" unit="EUR" splitting="A"/>
      <Data period="8" value="24.00" unit="EUR" splitting="A"/>
      <Data period="9" value="24.00" unit="EUR" splitting="A"/>
      <Data period="10" value="24.00" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="11XKORLEAINVESTY" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 13** Response containing orders for specific day

## 4.3 Administration of Intraday Market orders

Administration of orders of market participants is carried out through operations for order reception, order modification and order provision via IdmOrders web service at Intraday Market or interface WEB API and WebSocket.

### 4.3.1 Processing level

Market participants enter their orders into ISOT by the deadline for given trading hour, where orders can be entered into the system in advance, even for more trading time periods if they are open. Opening of trading hour for next day begins at 3 p.m. every day.

In case of invalid order, user is required to remove reasons for its invalidity by replacing it with a new version. Orders non-compliant with the check at submission (invalid) shall be removed and will not enter the matching process.

Depending on date of transition from domestic intraday market to coordinated cross-border intraday market under the SIDC (Single Intraday Coupling) project, products used within the intraday market will be changed according to the following:

- 60-minutes products:
  - simple orders for 1 period,
  - Block orders,
    - predefined (base/peak/off-peak) – submission **will not be** supported,
    - user-defined (own blocks) – zadávanie **will be** supported.
- 15-minutes products:
  - simple orders for 1 period.

In regards to change of products in relation to length of the trading period, messages under ISOTEDATA-VDT will use following definition of trading periods:

| **Length of product period** | | | | | |
|---|---|---|---|---|---|
| **60-minutes** | | | **15-minutes** | | |
| Period | Period from | Period to | Period | Period from | Period to |
| 00:00 – 01:00 | 0 | 1 | 00:00 – 00:15 | 0 | 1 |
| | | | 00:15 – 00:30 | 1 | 2 |
| | | | 00:30 – 00:45 | 2 | 3 |
| | | | 00:45 – 01:00 | 3 | 4 |
| 01:00 – 02:00 | 1 | 2 | 01:00 – 01:15 | 4 | 5 |
| | | | 01:15 – 01:30 | 5 | 6 |
| | | | 01:30 – 01:45 | 6 | 7 |
| | | | 01:45 – 02:00 | 7 | 8 |
| 02:00 – 03:00 | 2 | 3 | 02:00 – 02:15 | 8 | 9 |
| | | | 02:15 – 02:30 | 9 | 10 |
| | | | 02:30 – 02:45 | 10 | 11 |
| | | | 02:45 – 03:00 | 11 | 12 |

| Length of product period | | | | | |
|---|---|---|---|---|---|
| 60-minutes | | | 15-minutes | | |
| Period | Period from | Period to | Period | Period from | Period to |
| 03:00 – 04:00 | 3 | 4 | 03:00 – 03:15 | 12 | 13 |
| | | | 03:15 – 03:30 | 13 | 14 |
| | | | 03:30 – 03:45 | 14 | 15 |
| | | | 03:45 – 04:00 | 15 | 16 |
| 04:00 – 05:00 | 4 | 5 | 04:00 – 04:15 | 16 | 17 |
| | | | 04:15 – 04:30 | 17 | 18 |
| | | | 04:30 – 04:45 | 18 | 19 |
| | | | 04:45 – 05:00 | 19 | 20 |
| 05:00 – 06:00 | 5 | 6 | 05:00 – 05:15 | 20 | 21 |
| | | | 05:15 – 05:30 | 21 | 22 |
| | | | 05:30 – 05:45 | 22 | 23 |
| | | | 05:45 – 06:00 | 23 | 24 |
| 06:00 – 07:00 | 6 | 7 | 06:00 – 06:15 | 24 | 25 |
| | | | 06:15 – 06:30 | 25 | 26 |
| | | | 06:30 – 06:45 | 26 | 27 |
| | | | 06:45 – 07:00 | 27 | 28 |
| 07:00 – 08:00 | 7 | 8 | 07:00 – 07:15 | 28 | 29 |
| | | | 07:15 – 07:30 | 29 | 30 |
| | | | 07:30 – 07:45 | 30 | 31 |
| | | | 07:45 – 08:00 | 31 | 32 |
| 08:00 – 09:00 | 8 | 9 | 08:00 – 08:15 | 32 | 33 |
| | | | 08:15 – 08:30 | 33 | 34 |
| | | | 08:30 – 08:45 | 34 | 35 |
| | | | 08:45 – 09:00 | 35 | 36 |
| 09:00 – 10:00 | 9 | 10 | 09:00 – 09:15 | 36 | 37 |
| | | | 09:15 – 09:30 | 37 | 38 |
| | | | 09:30 – 09:45 | 38 | 39 |
| | | | 09:45 – 10:00 | 39 | 40 |
| 10:00 – 11:00 | 10 | 11 | 10:00 – 10:15 | 40 | 41 |
| | | | 10:15 – 10:30 | 41 | 42 |

| Length of product period | | | | | |
|---|---|---|---|---|---|
| 60-minutes | | | 15-minutes | | |
| Period | Period from | Period to | Period | Period from | Period to |
| | | | 10:30 – 10:45 | 42 | 43 |
| | | | 10:45 – 11:00 | 43 | 44 |
| 11:00 – 12:00 | 11 | 12 | 11:00 – 11:15 | 44 | 45 |
| | | | 11:15 – 11:30 | 45 | 46 |
| | | | 11:30 – 11:45 | 46 | 47 |
| | | | 11:45 – 12:00 | 47 | 48 |
| 12:00 – 13:00 | 12 | 13 | 12:00 – 12:15 | 48 | 49 |
| | | | 12:15 – 12:30 | 49 | 50 |
| | | | 12:30 – 12:45 | 50 | 51 |
| | | | 12:45 – 13:00 | 51 | 52 |
| 13:00 – 14:00 | 13 | 14 | 13:00 – 13:15 | 52 | 53 |
| | | | 13:15 – 13:30 | 53 | 54 |
| | | | 13:30 – 13:45 | 54 | 55 |
| | | | 13:45 – 14:00 | 55 | 56 |
| 14:00 – 15:00 | 14 | 15 | 14:00 – 14:15 | 56 | 57 |
| | | | 14:15 – 14:30 | 57 | 58 |
| | | | 14:30 – 14:45 | 58 | 59 |
| | | | 14:45 – 15:00 | 59 | 60 |
| 15:00 – 16:00 | 15 | 16 | 15:00 – 15:15 | 60 | 61 |
| | | | 15:15 – 15:30 | 61 | 62 |
| | | | 15:30 – 15:45 | 62 | 63 |
| | | | 15:45 – 16:00 | 63 | 64 |
| 16:00 – 17:00 | 16 | 17 | 16:00 – 16:15 | 64 | 65 |
| | | | 16:15 – 16:30 | 65 | 66 |
| | | | 16:30 – 16:45 | 66 | 67 |
| | | | 16:45 – 17:00 | 67 | 68 |
| 17:00 – 18:00 | 17 | 18 | 17:00 – 17:15 | 68 | 69 |
| | | | 17:15 – 17:30 | 69 | 70 |
| | | | 17:30 – 17:45 | 70 | 71 |
| | | | 17:45 – 18:00 | 71 | 72 |

| Length of product period | | | | | |
|---|---|---|---|---|---|
| 60-minutes | | | 15-minutes | | |
| Period | Period from | Period to | Period | Period from | Period to |
| 18:00 – 19:00 | 18 | 19 | 18:00 – 18:15 | 72 | 73 |
| | | | 18:15 – 18:30 | 73 | 74 |
| | | | 18:30 – 18:45 | 74 | 75 |
| | | | 18:45 – 19:00 | 75 | 76 |
| 19:00 – 20:00 | 19 | 20 | 19:00 – 19:15 | 76 | 77 |
| | | | 19:15 – 19:30 | 77 | 78 |
| | | | 19:30 – 19:45 | 78 | 79 |
| | | | 19:45 – 20:00 | 79 | 80 |
| 20:00 – 21:00 | 20 | 21 | 20:00 – 20:15 | 80 | 81 |
| | | | 20:15 – 20:30 | 81 | 82 |
| | | | 20:30 – 20:45 | 82 | 83 |
| | | | 20:45 – 21:00 | 83 | 84 |
| 21:00 – 22:00 | 21 | 22 | 21:00 – 21:15 | 84 | 85 |
| | | | 21:15 – 21:30 | 85 | 86 |
| | | | 21:30 – 21:45 | 86 | 87 |
| | | | 21:45 – 22:00 | 87 | 88 |
| 22:00 – 23:00 | 22 | 23 | 22:00 – 22:15 | 88 | 89 |
| | | | 22:15 – 22:30 | 89 | 90 |
| | | | 22:30 – 22:45 | 90 | 91 |
| | | | 22:45 – 23:00 | 91 | 92 |
| 23:00 – 24:00 | 23 | 24 | 23:00 – 23:15 | 92 | 93 |
| | | | 23:15 – 23:30 | 93 | 94 |
| | | | 23:30 – 23:45 | 94 | 95 |
| | | | 23:45 – 24:00 | 95 | 96 |

**Table 141** Converter for usage of periods within intraday market

### 4.3.2 Reception of orders (E-06_01)

Reception of orders is carried out by request for order submission in *ISOTEDATA-VDT.801* structure (message-code=801) and by response in *RESPONSE-VDT.802* structure (indication of success/failure) and *ISOTEDATA-VDT.803* structure (description of order registered in the system or in JSON format throught the automated WEB API interface or WebSocket). Market participant is not informed about successfully submitted order, but is informed about successful reception of order on ISOT side due to manners of synchronous communication (see chapter 3.2).

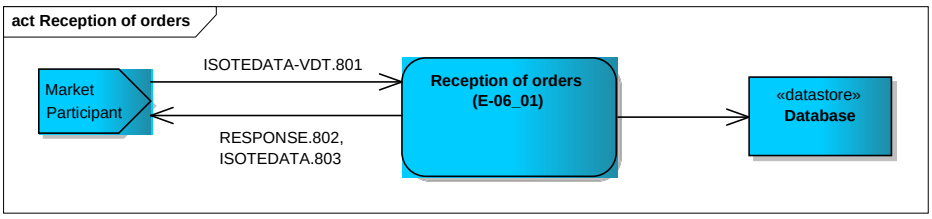


**Figure 24** Schema of market participant order reception

### *ISOTEDATA-VDT.801*

The structure contains *message-code=801* attribute in the header and is filled out in accordance with ISOTEDATA-VDT. Only a single order can be concurrently entered, i.e. submission of multiple orders is carried out through multiple calls.

```
<ISOTEDATA id="1" message-code="801" date-time="2016-06-20T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01">
  <SenderIdentification id="24X--YOUR-EIC--B " coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2016-02-16" order-expiration="2016-02-15T19:30:10" trade-type="N" block-
order="N" indication="N" trade-stage="P" delivery-duration="60" market-area="SK" sett-
curr="EUR" market="VDT" >
    <ProfileData profile-role="BC01">
      <Data period-from="0" period-to="1" value="19.1" unit="MW" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="0" period-to="1" value="75.00" unit="EUR" />
    </ProfileData>
    <Party id="24X--YOUR-EIC--B " role="TO" />
  </Trade>
```

**Example 14** Submission of 60-minutes sale order

### *RESPONSE-VDT.802*

In accordance with specification of RESPONSE-VDT structure, response is given in case of successful or unsuccessful order processing. In this case message-code=802 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
       date-time="2009-07-03T13:46:26Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03" trade-id="1016"/>
</RESPONSE>
```

**Example 15** Response on success of order submission

### *ISOTEDATA-VDT.803*

The structure is returned in the form it was registered in the system in case of successful order processing. In this case message-code=803 can be found in the header.

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="803"
       date-time="2016-02-15T16:30:10Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04/01">
  <SenderIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade trade-day="2016-02-16" order-expiration="2016-02-15T19:30:10" trade-type="N" block-
order="N" indication="N" trade-stage="P" delivery-duration="60" market-area="SK" sett-
curr="EUR" market="VDT">
    <TimeData datetime="2016-02-15T16:30:10Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period-from="0" period-to="1" value="19.1" unit="MW" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="0" period-to="1" value="75.00" unit="EUR" />
    </ProfileData>
    <Party id="24X--YOUR-EIC--B " role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 16** Response with description of entered order in the system

*JSON*

The structure is used in the WEB API and WebSocket interfaces for placing orders on the intraday market.

POST IDM orders-create

WEB API Interface: The structure is in JSON format. A single order or multiple orders can be submitted at once.

Request:

Url: https://{hostname}/api/v1/idm/orders

JSON:

```
{
  "correlationId":"1@2024-11-18T07:59:11.261+01:00",
  "groupIndication": "rejectAll",
  "orders":[
    {
      "direction":"sell",
      "indication":"noIndication",
      "deliveryStart":"2024-11-18T20:00:00Z",
      "deliveryEnd":"2024-11-18T21:00:00Z",
      "expiration":"2024-11-18T11:00:00Z",
      "quantity":1.2,
      "price":10.5,
      "active":true,
      "note":"just my note",
      "type":"simple"
    }
  ]
}
```

**Example 17** Request to place a single 60-minute sell order.

*Response:*

In the case of successful or unsuccessful request processing, a response is returned. For a successful creation, the status code 201 Created or 202 Accepted is returned.

If the request processing fails, the interface may return a response with the following structure:


```
{
   "code": "ValidationProblem",
   "message": "Validation problems occurred.",
   "errors": {
      "item-1": [
         {
            "code": "IntervalIsOpenRule",
            "message": "Invalid order. Reason: IntervalIsOpenRule",
            "messageArgs": [
               "2024-11-18T00:00:00+01:00"
            ]
         }
      ]
   }
}
```


**Example 18** Example of a response for an unsuccessful request

*SEND IDM orders-create*

WebSocket Interfaces: The structure is in JSON format. A single order or multiple orders can be submitted at once.

Message: order-create

- topics=orders: This parameter specifies that the client wants to receive only changes to their own orders. It is possible to extend the existing connection with additional topics as needed. For more detailed information on how to extend the connection, please refer to the chapter [WebSocket connection].


```
connect: wss://{hostname}/api/v1/idm/ws?topics=orders

JSON:

{
  "payload":{
    "correlationId":"test01",
    "groupIndication": "rejectAll",
    "orders":[
      {
        "direction":"sell",
        "indication":"noIndication",
        "deliveryStart":"2024-11-20T15:15:00Z",
        "deliveryEnd":"2024-11-20T15:30:00Z",
        "expiration":null,
        "quantity":1.1,
        "price":1.5,
        "active":true,
        "note":"",
        "type":"simple"
      }
    ]
  },
  "type":"order-create"
}
```

**Example 19** Request to place a single 15-minute order.

### 4.3.3 Modification of orders (E-06_02)

Modification of orders is realized by request for modification of order in structure ISOTEDATA-VDT.804 and by response in structure RESPONSE-VDT.805 (indication of success/failure) or in JSON format throught the automated WEB API interface or WebSocket. Market participant is not informed about successfully modified order, but is informed about successful reception of instruction for order modification on ISOT side due to manners of synchronous communication (see chapter 3.2).

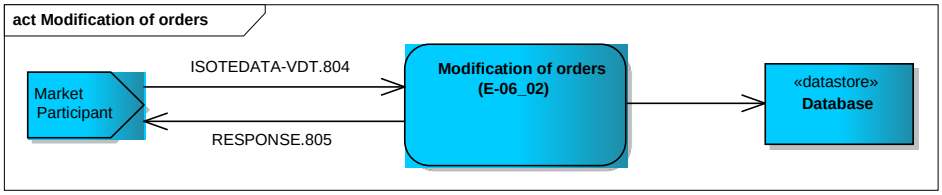


**Figure 25** Reception of order scheme

Market participant can modify status of the order in case of request for modification. Activating, deactivating or canceling already inputted order can be done through external interfaces.

*ISOTEDATA-VDT.804*

The structure contains *message-code=804* attribute in the header and is filled out in accordance with ISOTEDATA-VDT. Only a single order can be concurrently entered, i.e. submission of multiple orders is carried out through multiple calls.

```
<ISOTEDATA id="Your_own_ID_for_this_message" message-code="802" date-time="2011-01-
18T09:52:37" dtd-version="1" dtd-release="1" answer-required="false"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
   <SenderIdentification id="24X--YOUR-EIC--B" coding-scheme="15" />
   <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
   <Trade id="1234" trade-stage="N" delivery-duration="60" market-area="SK" market="VDT">
      <Party id="24X--YOUR-EIC--B" role="TO" />
   </Trade>
</ISOTEDATA>
```

**Example 20** Deactivation of order

*RESPONSE-VDT.805*

In accordance with specification of RESPONSE-VDT structure response is given in case of successful or unsuccessful order processing. In this case message-code=805 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="805"
       date-time="2016-02-15T16:30:10Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 21** Successful modification of order response

### *JSON for WEB API and WebSocket*

The structure is used in the WEB API and WebSocket interfaces for modifying orders on the intraday market.

*POST IDM orders-activate*

WEB API Interface: The structure is in JSON format. Changing the status of a single order based on its ID.

Request:

Url: https://{hostname}/api/v1/idm/orders/{orderid}/activate

JSON:

```
{
   "correlationId": "1@2024-11-18T07:59:11.261+01:00"
}
```

**Example 22** Request to activate an order.

*Response:*

In the case of successful or unsuccessful request processing, a response is returned. For a successful creation, the status code 201 Created or 202 Accepted is returned.

If the request processing fails, the interface may return a response with the following structure:

```
{
   "code": "ValidationProblem",
   "message": "Validation problems occurred.",
   "errors": {
      "item-1": [
         {
            "code": "IntervalIsOpenRule",
            "message": "Invalid order. Reason: IntervalIsOpenRule",
            "messageArgs": [
               "2024-11-18T00:00:00+01:00"
            ]
         }
      ]
   }
}
```

**Example 23** Example of a response for an unsuccessful request.

*POST IDM orders-deactivate*

WEB API Interface: The structure is in JSON format. Changing the status of a single order based on its ID.

Request:

Url: https://{hostname}/api/v1/idm/orders/{orderid}/deactivate

JSON:


```
{
   "correlationId": "1@2024-11-18T07:59:11.261+01:00"
}
```


**Example 24** Request to deactivate an order.

*Response:*

In the case of successful or unsuccessful request processing, a response is returned. For a successful creation, the status code 201 Created or 202 Accepted is returned.

If the request processing fails, the interface may return a response with the following structure:


```
{
   "code": "ValidationProblem",
   "message": "Validation problems occurred.",
   "errors": {
      "item-1": [
         {
            "code": "IntervalIsOpenRule",
            "message": "Invalid order. Reason: IntervalIsOpenRule",
            "messageArgs": [
               "2024-11-18T00:00:00+01:00"
            ]
         }
      ]
   }
}
```


**Example 25** Example of a response for an unsuccessful request.

*POST IDM orders-cancel*

WEB API Interface: The structure is in JSON format. Changing the status of a single order based on its ID.

Request:

Url: https://{hostname}/api/v1/idm/orders/{orderid}/cancel

JSON:


```
{
   "correlationId": "1@2024-11-18T07:59:11.261+01:00"
}
```


**Example 26** Request to cancel an order.

*Response:*

In the case of successful or unsuccessful request processing, a response is returned. For a successful creation, the status code 201 Created or 202 Accepted is returned.

*SEND IDM orders-activate*

WebSocket Interface: The structure is in JSON format. Changing the status of a single order based on its ID.

Message order-activate:

topics=orders: This parameter specifies that the client wants to receive only updates about their own orders. It is possible to extend the existing connection with additional topics as needed. For more detailed information on how to extend the connection, please refer to the [WebSocket connection] chapter.

```
connect: wss://{hostname}/api/v1/idm/ws?topics=orders

{
   "type": "order-activate",
   "payload": { "id": 1, "correlationId": "random-text" }
}
```

**Example 27** Request to activate an order.

*SEND IDM orders-deactivate*

WebSocket Interface: The structure is in JSON format. Changing the status of a single order based on its ID.

Message order-deactivate:

topics=orders: This parameter specifies that the client wants to receive only updates about their own orders. It is possible to extend the existing connection with additional topics as needed. For more detailed information on how to extend the connection, please refer to the [WebSocket connection] chapter.

```
connect: wss://{hostname}/api/v1/idm/ws?topics=orders

{
   "type": "order-deactivate",
   "payload": { "id": 1, "correlationId": "random-text" }
}
```

**Example 28** Request to deactivate an order.

*SEND IDM orders-cancel*

WebSocket Interface: The structure is in JSON format. Changing the status of a single order based on its ID.

Message order-cancel:

topics=orders: This parameter specifies that the client wants to receive only updates about their own orders. It is possible to extend the existing connection with additional topics as needed. For more detailed information on how to extend the connection, please refer to the [WebSocket connection] chapter.

```
connect: wss://{hostname}/api/v1/idm/ws?topics=orders

{
   "type": "order-cancel",
   "payload": { "id": 1, "correlationId": "random-text" }
}
```

**Example 29** Request to cancel an order.

### 4.3.4 Provision of orders (E-06_03)

Provision of orders is carried out by request for provision of order in *CDSREQ-VDT.807* structure and response in *RESPONSE-VDT.809* structure (indication of success/fail) and ISOTEDATA-VDT.809 structure (description of modified order) or in JSON format throught the automated WEB API interface or WebSocket.

Market participant can ask for provision of a specific order or every order for specific time frame in a request for modification of an order.

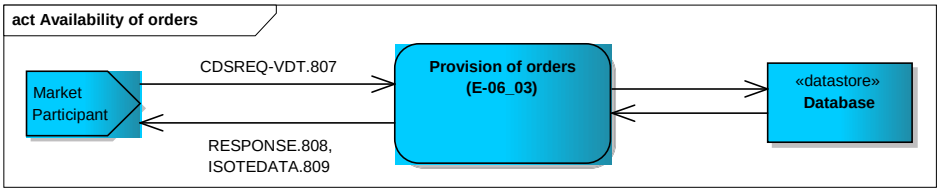


**Figure 26** Provision of orders scheme

*CDSREQ-VDT.807*

The structure contains *message-code=807* attribute in the header and is filled out in accordance with structure specification of order provision. Request for order´s data can be formulated either for specific order with order´s identification in attribute Trade/@id, or via request for every order in given time period with attribute Trade/@trade-day. It is possible to specify time period with Trade/@period-from and Trade/@period-to.

```
<CDSREQ date-time="2017-04-11T07:00:00" dtd-release="1" dtd-version="1"  id="45t" message-
code="807"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
    <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
    <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
    <Trade id=1/>
</CDSREQ>
```

**Example 30** Provision of specific order

```
<CDSREQ date-time="2017-04-11T07:00:00" dtd-release="1" dtd-version="1"  id="45t" message-
code="807"
xmlns:ns=" http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
   <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
   <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
   <Trade trade-day="2017-02-12" period-from="8" period-to="16" delivery-duration = "60"/>
</CDSREQ>
```

**Example 31** Provision of all orders in given time period

### *RESPONSE-VDT.808*

In accordance with specification of RESPONSE-VDT structure response is given in case of successful or unsuccessful order processing. In this case message-code=808 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="808"
       date-time="2016-02-15T16:30:10Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 32** Successful modification of order response

### *ISOTEDATA-VDT.809*

In case of successful order processing, the structure is returned as it was registered in the system with a message-code=809 in the header. Attribute Trade/@trade-stage contains status in which the order currently is. Interval of delivery is specified in attributes Trade/ProfileData/Data/@period-from and Trade/ProfileData/Data/@period-to. These intervals represent time periods order of a given day.

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="809" date-time="2016-02-
15T16:30:10Z" dtd-version="1" dtd-release="1" answer-required="false"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
     <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
     <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
     <Reference id="1"/>
     <Trade id="1" trade-day="2016-02-16" order-expiration="2016-02-15T19:30:10" trade-
type="N" block-order="N" indication="N" trade-stage="P" trader-id="123456" delivery-duration
= "60" market-area="SK" sett-curr="EUR" market="VDT">
        <TimeData datetime="2016-02-15T16:30:10Z" datetime-type="DTC"/>
        <ProfileData profile-role="BC01">
           <Data period-from="10" period-to="11" value="19.1" unit="MW" />
        </ProfileData>
        <ProfileData profile-role="BP01">
            <Data period-from="10" period-to="11" value="75.00" unit="EUR" />
        </ProfileData>
        <Party id="24X--YOUR-EIC--B " role="TO"/>
     </Trade>
     <Trade id="2" trade-day="2016-02-16" order-expiration="2016-02-15T19:30:10" trade-
type="N" block-order="N" indication="N" trade-stage="P" trader-id="123456" delivery-duration
= "60" market-area="SK" sett-curr="EUR" market="VDT">
        <TimeData datetime="2016-02-15T16:35:10Z" datetime-type="DTC"/>
        <ProfileData profile-role="BC01">
           <Data period-from="10" period-to="11" value="2" unit="MW" />
        </ProfileData>
        <ProfileData profile-role="BP01">
            <Data period-from="10" period-to="11" value="60.00" unit="EUR" />
        </ProfileData>
        <Party id="24X--YOUR-EIC--B " role="TO"/>
     </Trade>
</ISOTEDATA>
```

**Example 33** Response with description of two available orders

### *JSON for WEB API and WebSocket*

The structure is used in the WEB API and WebSocket interfaces to access orders on the intraday market.

#### *GET IDM orders-list*

WEB API Interface: The structure is in JSON format. If the request is successfully processed, the structure of the order is returned in the form it exists at the time of access. The status object contains the current status of the order in the system. The delivery interval is specified in the deliveryStart and deliveryEnd objects. These intervals represent the sequence of the period for the given day.

Request:

```
https://{hostname}/api/v1/idm/orders?status=canceled,matched&product=60&createdFrom=2024-11-13T16:20:00Z&createdTo=2024-11-18T17:00:00Z&updatedFrom=2024-11-13T16:20:00Z&updatedTo=2024-11-18T17:00:00Z&deliveryFrom=2024-11-13T08:00:00Z&deliveryTo=2024-11-18T09:00:00Z&offset=0&limit=2&expandTrades=true
```

**Example 34** URL to access two orders for a given interval.

Response:

If the request is successfully processed, a response is returned in JSON format along with a status code **200 OK**.

```
[
  {
    "id":123456,
    "type":“simple“,
    "productType":60,
    "deliveryDay":"2024-11-14",
    "deliveryStart":"2024-11-14T14:00:00Z",
    "deliveryEnd":"2024-11-14T15:00:00Z",
    "direction":"buy",
    "quantity":1.3,
    "price":0.82,
    "status":"matched",
    "isPending":false,
    "realizedQuantity":1.3,
    "realizedPriceWeighted":60.11,
    "remainingQuantity":0,
    "expiration":null,
    "createdAt":"2024-11-13T16:58:10Z",
    "updatedAt":"2024-11-13T16:58:16Z",
    "createdBy":"tester01",
    "trades":[
      {
        "id":112233,
        "time":"2024-11-13T17:58:10",
        "price":60.11,
        "quantity":1.3
      }
    ]
  },
  {
    "id":123457,
    "type":“simple“,
    "productType":60,
    "deliveryDay":"2024-11-14",
    "deliveryStart":"2024-11-14T14:00:00Z",
    "deliveryEnd":"2024-11-14T15:00:00Z",
    "direction":"sell",
    "quantity":1.4,
    "price":0.49,
    "status":"matched",
    "isPending":false,
    "realizedQuantity":1.4,
    "realizedPriceWeighted":60.11,
    "remainingQuantity":0,
    "expiration":null,
    "createdAt":"2024-11-13T16:58:10Z",
    "updatedAt":"2024-11-13T16:58:16Z",
    "createdBy":"tester02",
    "trades":[
      {
        "id":112244,
        "time":"2024-11-13T17:58:10",
        "price":60.11,
        "quantity":1.4
      }
    ]
  }
```


**Example 35** Response with description of two available orders

*GET IDM orders-detail*

WEB API Interface: The structure is in JSON format. If the request is successfully processed, the structure of the order is returned in the form it exists at the time of access. The status object contains the current status of the order in the system. The delivery interval is specified in the deliveryStart and deliveryEnd objects. These intervals represent the sequence of the period for the given day.

.

Request:

```
https://{hostname}/api/v1/idm/orders/{orderid}
```

**Example 36** URL to access the detail of an order by ID.

Response:

If the request is successfully processed, a response is returned in JSON format along with a status code **200 OK**.

```
{
  "id":1,
  "type":“simple“,
  "productType":60,
  "deliveryDay":"2024-11-14",
  "deliveryStart":"2024-11-14T14:00:00Z",
  "deliveryEnd":"2024-11-14T15:00:00Z",
  "direction":"sell",
  "quantity":1.3,
  "price":0.82,
  "status":"matched",
  "isPending":false,
  "realizedQuantity":1.3,
  "realizedPriceWeighted":60.11,
  "remainingQuantity":0,
  "expiration":null,
  "createdAt":"2024-11-13T16:58:10Z",
  "updatedAt":"2024-11-13T16:58:16Z",
  "createdBy":"tester01",
  "participant":"participanteic",
  "trades":[
    {
      "id":2,
      "time":"2024-11-13T17:58:10",
      "price":60.11,
      "quantity":1.3
    }
  ]
}
```

**Example 37** Response detailing a single order.

*GET IDM trades of order*

WEB API Interface: The structure is in JSON format. If the request is successfully processed, the structure of the order is returned in the form it exists at the time of access. The root object contains a list of trades associated with the specific order.

Request:

```
https://{hostname}/api/v1/idm/orders/{orderid}/trades
```

**Example 38** URL to access the details of trades by order ID.

Response:

If the request is successfully processed, a response is returned in JSON format along with a status code **200 OK**.

```
[
   {
      "id": 1,
      "time": "2024-11-13T17:58:10Z",
      "price": 75.9,
      "quantity": 1
   }
]
```

**Example 39** Response detailing a single trade.

*SEND IDM orders-detail*

WebSocket Interface: The structure is in JSON format. If the request is successfully processed, the order structure is returned in the form it exists at the time of access. The status object contains the current state of the order in the system. The delivery interval is specified in the deliveryStart and deliveryEnd objects, representing the sequence of the period for the given day.

Message: order-detail

topics=orders: This parameter indicates that the client wants to receive updates only for changes to their own orders. The existing connection can be extended with additional topics as needed. For more detailed information on how to extend the connection, refer to the chapter [WebSocket Connection].

```
connect: wss://{hostname}/api/v1/idm/ws?topics=orders

{
   "type": "order-detail",
   "payload": { "id": 1 }
}
```

**Example 40** Connection and message to access the order detail by ID.

Received message:

```
{
    "payload": {
        "id": 1,
      "type":“simple“,
        "productType": 60,
        "deliveryDay": "2024-11-20",
        "deliveryStart": "2024-11-20T08:00:00Z",
        "deliveryEnd": "2024-11-20T09:00:00Z",
        "direction": "sell",
        "quantity": 45,
        "price": 11,
        "status": "active",
        "isPending": false,
        "realizedQuantity": 0,
        "remainingQuantity": 45,
        "createdAt": "2024-11-19T20:57:08Z",
        "updatedAt": "2024-11-19T20:57:08Z",
        "createdBy": "tester01",
        "participant": "participant01",
        "trades": []
    },
    "type": "order-detail"
}
```


**Example 41** Order detail by ID.

## 4.4 Administration of intraday order book

Administration of intraday order book is carried out through operation intraday order book provision via IdmOrderBook web service. Within the coordinated cross-border intraday market, the IdmOrderBook web service will also allow market participants to obtain available cross-border capacities in the form of Hub-to-Hub matrix data.

### 4.4.1 Processing level

Market participants have the option to request data about the current order book status. Communication runs via AMQP protocol which automatically informs market participants about changes in the order book. For more information about automated notifications which inform about changes in intraday order book, see Chapters 3.2 and 4.7.

Depending on date of transition from domestic intraday market to coordinated cross-border intraday market under the SIDC (Single Intraday Coupling) project, the structure of orderb ook data will also be extended by the information of period length. Information regarding the order book data will be additionally specified by period length (60 or 15 minutes) relating to the specific order type.

Using web service IdmOrderBook and Download method market participant gets information about all available quantities of offered/demanded limited prices per each period of relevant product in case of simple orders, or for all products, alternatively intervals in case of block orders. Using the download method, the market participant has further possibility to obtain overall traded quantity, last traded price and quantity including the price direction within the relevant period.

Moreover, market participants are able using the web service IdmOrderBook and method DownloadH2HMatrix to download information regarding available transmission cross-border capacities between SK trading area and other trading areas pariticipating within the SIDC market. These data regarding cross-border capacities are of an exclusively informative nature and their availability may be subject to a delay compared to the real state in the central intraday market order matching solution within the SIDC.

### 4.4.2 The order book data (E-08_01)

Provision of order book data is realized by request for provision of order book in CDSREQ-VDT.810 structure and response in RESPONSE-VDT.811 structure (success/failure) and ISOTEDATA-VDT.81 (description of the order book data) or in JSON format throught the automated WEB API interface or WebSocket.

In response, market participant gets all available amounts for a specific time frame in the order book.

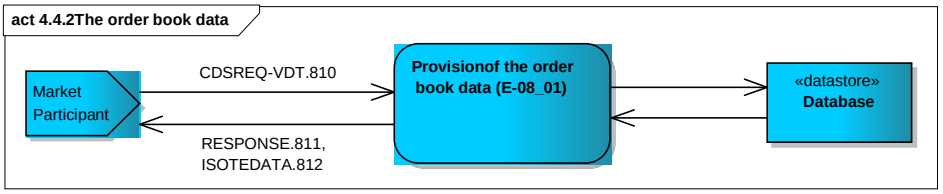


**Figure 27** Provision of order book data scheme

*CDSREQ-VDT.810*

The structure contains message-code=810 attribute in the header and is filled out in accordance with specification of order book provision structure. Request for the order book data is realized with optional possibility to specify period length (60 or 15 minutes) within the Trade/@delivery-duration element. System automatically evaluate current status of the order book after reception of a request and provides status for specified time period. In the case of not defining the information on the length of the period, all data will be made available broken down by the length of the trading period. Market participant receive data for specific time period in response.

```
<CDSREQ date-time="2017-04-11T07:00:00" dtd-release="1" dtd-version="1"  id="45t" message-
code="810"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
   <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
   <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
</CDSREQ>
```

**Example 42** Provision of order book

```
<CDSREQ date-time="2017-04-11T07:00:00" dtd-release="1" dtd-version="1"  id="45t" message-
code="810"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
   <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
   <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
   <Trade delivery-duration="60"/>
</CDSREQ>
```

**Example 43** Provision of order book for specific period length

*RESPONSE-VDT.811*

In accordance with specification of RESPONSE structure, response is given in case of successful or unsuccessful order processing. In this case message-code=811 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="811"
       date-time="2016-02-15T16:30:10Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 44** Response about successful processing of order book data provision request

*ISOTEDATA-VDT.812*

In case of successful order processing, all available order book data are returned for a given time period and period length and message-code=812 can be found in the header. Information about amount of particular types of orders are ordered in limited prices and periods can be found in response. Attribute Trade/ProfileData/Data/@seq-num order different prices of periods according to view of the market participant from the best price (the highest for purchase, the lowest for sale) to the worst price (the lowest for purchase, the lowest for sale). Matching of user-defined block orders is established on different principals as other type of orders, so it is not possible aggregate amount of orders after inputted prices. For this reason, availability of marked user-defined block orders is shown under anonymous ID in attribute Trade/ProfileData/@trade-id. Further it is possible to obtain statistics of the orderbook in extent of overall traded quantity (MW) within the relevant period using the attribute Trade/ProfileData/@profile-role=”TC01”, last traded quantity (MW) within the period using the attribute Trade/ProfileData/@profile-role=”LC01” and last price (EUR/MWh) including the price direction using the attribute Trade/ProfileData/@profile-role=”LP01”.

```
<?xml version="1.0" encoding="utf-8"?>
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="812"
date-time="2016-02-15T16:30:10Z" dtd-version="1" dtd-release="1" answer-required="false"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24X--YOUR-EIC--B" coding-scheme="15"            />
  <Reference id="1"/>
  <!--den D, posledná cena/množstvo-->
  <Trade trade-day="2016-07-13" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
    <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
    <ProfileData profile-role="TC01">
      <Data period-from="10" period-to="11" value="105" unit="MW"/>
      <Data period-from="11" period-to="12" value="246" unit="MW"/>
      <Data period-from="12" period-to="13" value="231.3" unit="MW"/>
      <Data period-from="13" period-to="14" value="101.5" unit="MW"/>
      <Data period-from="14" period-to="15" value="594.3" unit="MW"/>
      <Data period-from="15" period-to="16" value="3232.1" unit="MW"/>
      <Data period-from="16" period-to="17" value="1123.4" unit="MW"/>
      <Data period-from="17" period-to="18" value="340.1" unit="MW"/>
      <Data period-from="18" period-to="19" value="764.5" unit="MW"/>
      <Data period-from="19" period-to="20" value="100.5" unit="MW"/>
      <Data period-from="20" period-to="21" value="126.3" unit="MW"/>
      <Data period-from="21" period-to="22" value="314.2" unit="MW"/>
      <Data period-from="22" period-to="23" value="500" unit="MW"/>
      <Data period-from="23" period-to="24" value="5000" unit="MW"/>
    </ProfileData>
    <ProfileData profile-role="LC01">
      <Data period-from="10" period-to="11" value="5" unit="MW"/>
      <Data period-from="11" period-to="12" value="6" unit="MW"/>
      <Data period-from="12" period-to="13" value="2.1" unit="MW"/>
      <Data period-from="13" period-to="14" value="10" unit="MW"/>
      <Data period-from="14" period-to="15" value="5" unit="MW"/>
      <Data period-from="15" period-to="16" value="3" unit="MW"/>
      <Data period-from="16" period-to="17" value="1" unit="MW"/>
      <Data period-from="17" period-to="18" value="0.1" unit="MW"/>
      <Data period-from="18" period-to="19" value="0.5" unit="MW"/>
      <Data period-from="19" period-to="20" value="10.5" unit="MW"/>
      <Data period-from="20" period-to="21" value="12" unit="MW"/>
      <Data period-from="21" period-to="22" value="11" unit="MW"/>
      <Data period-from="22" period-to="23" value="50" unit="MW"/>
      <Data period-from="23" period-to="24" value="500" unit="MW"/>
    </ProfileData>
    <ProfileData profile-role="LP01">
      <Data period-from="10" period-to="11" value="110" unit="EUR" price-direction="N"/>
      <Data period-from="11" period-to="12" value="121.22" unit="EUR" price-direction="I"/>
      <Data period-from="12" period-to="13" value="114" unit="EUR" price-direction="D"/>
      <Data period-from="13" period-to="14" value="55" unit="EUR" price-direction="D"/>
      <Data period-from="14" period-to="15" value="59.3" unit="EUR" price-direction="I"/>
      <Data period-from="15" period-to="16" value="142" unit="EUR" price-direction="I"/>
      <Data period-from="16" period-to="17" value="150" unit="EUR" price-direction="N"/>
      <Data period-from="17" period-to="18" value="-5" unit="EUR" price-direction="D"/>
      <Data period-from="18" period-to="19" value="0.12" unit="EUR" price-direction="D"/>
      <Data period-from="19" period-to="20" value="123" unit="EUR" price-direction="I"/>
      <Data period-from="20" period-to="21" value="45" unit="EUR" price-direction="D"/>
      <Data period-from="21" period-to="22" value="89" unit="EUR" price-direction="D"/>
      <Data period-from="22" period-to="23" value="90.2" unit="EUR" price-direction="I"/>
      <Data period-from="23" period-to="24" value="15" unit="EUR" price-direction="D"/>
    </ProfileData>
  </Trade>
  <!--den D+1, posledná cena/množstvo-->
  <Trade trade-day="2016-07-14" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
    <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
    <ProfileData profile-role="TC01">
      <Data period-from="0" period-to="1" value="105" unit="MW"/>
      <Data period-from="1" period-to="2" value="246" unit="MW"/>
      <Data period-from="2" period-to="3" value="231.3" unit="MW"/>
      <Data period-from="3" period-to="4" value="101.5" unit="MW"/>
      <Data period-from="4" period-to="5" value="594.3" unit="MW"/>
      <Data period-from="5" period-to="6" value="3232.1" unit="MW"/>
      <Data period-from="6" period-to="7" value="1123.4" unit="MW"/>
      <Data period-from="7" period-to="8" value="340.1" unit="MW"/>
      <Data period-from="8" period-to="9" value="764.5" unit="MW"/>
      <Data period-from="9" period-to="10" value="100.5" unit="MW"/>
      <Data period-from="10" period-to="11" value="126.3" unit="MW"/>
      <Data period-from="11" period-to="12" value="246" unit="MW"/>
      <Data period-from="12" period-to="13" value="231.3" unit="MW"/>
      <Data period-from="13" period-to="14" value="101.5" unit="MW"/>
      <Data period-from="14" period-to="15" value="594.3" unit="MW"/>
      <Data period-from="15" period-to="16" value="3232.1" unit="MW"/>
      <Data period-from="16" period-to="17" value="1123.4" unit="MW"/>
      <Data period-from="17" period-to="18" value="340.1" unit="MW"/>
      <Data period-from="18" period-to="19" value="764.5" unit="MW"/>
      <Data period-from="19" period-to="20" value="100.5" unit="MW"/>
      <Data period-from="20" period-to="21" value="126.3" unit="MW"/>
      <Data period-from="21" period-to="22" value="314.2" unit="MW"/>
      <Data period-from="22" period-to="23" value="500" unit="MW"/>
      <Data period-from="23" period-to="24" value="5000" unit="MW"/>
    </ProfileData>
```

```
  <ProfileData profile-role="LC01">
      <Data period-from="0" period-to="1" value="5" unit="MW"/>
      <Data period-from="1" period-to="2" value="6" unit="MW"/>
      <Data period-from="2" period-to="3" value="2.1" unit="MW"/>
      <Data period-from="3" period-to="4" value="10" unit="MW"/>
      <Data period-from="4" period-to="5" value="5" unit="MW"/>
      <Data period-from="5" period-to="6" value="3" unit="MW"/>
      <Data period-from="6" period-to="7" value="1" unit="MW"/>
      <Data period-from="7" period-to="8" value="0.1" unit="MW"/>
      <Data period-from="8" period-to="9" value="0.5" unit="MW"/>
      <Data period-from="9" period-to="10" value="10.5" unit="MW"/>
      <Data period-from="10" period-to="11" value="5" unit="MW"/>
      <Data period-from="11" period-to="12" value="6" unit="MW"/>
      <Data period-from="12" period-to="13" value="2.1" unit="MW"/>
      <Data period-from="13" period-to="14" value="10" unit="MW"/>
      <Data period-from="14" period-to="15" value="5" unit="MW"/>
      <Data period-from="15" period-to="16" value="3" unit="MW"/>
      <Data period-from="16" period-to="17" value="1" unit="MW"/>
      <Data period-from="17" period-to="18" value="0.1" unit="MW"/>
      <Data period-from="18" period-to="19" value="0.5" unit="MW"/>
      <Data period-from="19" period-to="20" value="10.5" unit="MW"/>
      <Data period-from="20" period-to="21" value="12" unit="MW"/>
      <Data period-from="21" period-to="22" value="11" unit="MW"/>
      <Data period-from="22" period-to="23" value="50" unit="MW"/>
      <Data period-from="23" period-to="24" value="500" unit="MW"/>
    </ProfileData>
    <ProfileData profile-role="LP01">
      <Data period-from="0" period-to="1" value="110" unit="EUR" price-direction="N"/>
      <Data period-from="1" period-to="2" value="121.22" unit="EUR" price-direction="I"/>
      <Data period-from="2" period-to="3" value="114" unit="EUR" price-direction="D"/>
      <Data period-from="3" period-to="4" value="55" unit="EUR" price-direction="D"/>
      <Data period-from="4" period-to="5" value="59.3" unit="EUR" price-direction="I"/>
      <Data period-from="5" period-to="6" value="142" unit="EUR" price-direction="I"/>
      <Data period-from="6" period-to="7" value="150" unit="EUR" price-direction="N"/>
      <Data period-from="7" period-to="8" value="-5" unit="EUR" price-direction="D"/>
      <Data period-from="8" period-to="9" value="0.12" unit="EUR" price-direction="D"/>
      <Data period-from="9" period-to="10" value="123" unit="EUR" price-direction="I"/>
      <Data period-from="10" period-to="11" value="110" unit="EUR" price-direction="N"/>
      <Data period-from="11" period-to="12" value="121.22" unit="EUR" price-direction="I"/>
      <Data period-from="12" period-to="13" value="114" unit="EUR" price-direction="D"/>
      <Data period-from="13" period-to="14" value="55" unit="EUR" price-direction="D"/>
      <Data period-from="14" period-to="15" value="59.3" unit="EUR" price-direction="I"/>
      <Data period-from="15" period-to="16" value="142" unit="EUR" price-direction="I"/>
      <Data period-from="16" period-to="17" value="150" unit="EUR" price-direction="N"/>
      <Data period-from="17" period-to="18" value="-5" unit="EUR" price-direction="D"/>
      <Data period-from="18" period-to="19" value="0.12" unit="EUR" price-direction="D"/>
      <Data period-from="19" period-to="20" value="123" unit="EUR" price-direction="I"/>
      <Data period-from="20" period-to="21" value="45" unit="EUR" price-direction="D"/>
      <Data period-from="21" period-to="22" value="89" unit="EUR" price-direction="D"/>
      <Data period-from="22" period-to="23" value="90.2" unit="EUR" price-direction="I"/>
      <Data period-from="23" period-to="24" value="15" unit="EUR" price-direction="D"/>
    </ProfileData>
  </Trade>
<!--Jednoduche objednavky, nakup, den D-->
  <Trade trade-day="2016-07-13" trade-type="N" block-order="N" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
    <!--datum a cas okamihu casoveho rezu orderbooku-->
    <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
    <ProfileData profile-role="BC01">
      <Data period-from="12" period-to="13" value="5" unit="MW" seq-num="1"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="12" period-to="13" value="31" unit="EUR" seq-num="1"/>
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="12" period-to="13" value="5" unit="MW" seq-num="2"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="12" period-to="13" value="25" unit="EUR" seq-num="2"/>
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="15" period-to="16" value="3" unit="MW" seq-num="1"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="15" period-to="16" value="35" unit="EUR" seq-num="1"/>
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="16" period-to="17" value="10" unit="MW" seq-num="1"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="16" period-to="17" value="30" unit="EUR" seq-num="1"/>
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="16" period-to="17" value="10" unit="MW" seq-num="2"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="16" period-to="17" value="20" unit="EUR" seq-num="2"/>
    </ProfileData>
```

```
  <ProfileData profile-role="BC01">
    <Data period-from="16" period-to="17" value="5" unit="MW" seq-num="3"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="16" period-to="17" value="18.73" unit="EUR" seq-num="3"/>
  </ProfileData>
</Trade>
<!--Jednoduche objednavky, predaj, den D-->
<Trade trade-day="2016-07-13" trade-type="P" block-order="N" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
  <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
  <ProfileData profile-role="BC01">
    <Data period-from="12" period-to="13" value="10" unit="MW" seq-num="1"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="12" period-to="13" value="33" unit="EUR" seq-num="1"/>
  </ProfileData>
  <ProfileData profile-role="BC01">
    <Data period-from="12" period-to="13" value="10" unit="MW" seq-num="2"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="12" period-to="13" value="40" unit="EUR" seq-num="2"/>
  </ProfileData>
  <ProfileData profile-role="BC01">
    <Data period-from="12" period-to="13" value="5" unit="MW" seq-num="3"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="12" period-to="13" value="41" unit="EUR" seq-num="3"/>
  </ProfileData>
  <ProfileData profile-role="BC01">
    <Data period-from="14" period-to="15" value="1" unit="MW" seq-num="1"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="14" period-to="15" value="20" unit="EUR" seq-num="1"/>
  </ProfileData>
  <ProfileData profile-role="BC01">
    <Data period-from="16" period-to="17" value="10" unit="MW" seq-num="1"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="16" period-to="17" value="45" unit="EUR" seq-num="1"/>
  </ProfileData>
  <ProfileData profile-role="BC01">
    <Data period-from="16" period-to="17" value="10" unit="MW" seq-num="2"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="16" period-to="17" value="46" unit="EUR" seq-num="2"/>
  </ProfileData>
  <ProfileData profile-role="BC01">
    <Data period-from="16" period-to="17" value="2" unit="MW" seq-num="3"/>
  </ProfileData>
  <ProfileData profile-role="BP01">
    <Data period-from="16" period-to="17" value="46.15" unit="EUR" seq-num="3"/>
  </ProfileData>
 </Trade>
 !--Jednoduche objednavky, nakup, den D+1-->
 <Trade trade-day="2016-07-14" trade-type="N" block-order="N" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
   <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
   <ProfileData profile-role="BC01">
     <Data period-from="0" period-to="1" value="5" unit="MW" seq-num="1"/>
   </ProfileData>
   <ProfileData profile-role="BP01">
     <Data period-from="0" period-to="1" value="20" unit="EUR" seq-num="1"/>
   </ProfileData>
   </Trade>
   <!--Jednoduche objednavky, predaj, den D+1-->
   <Trade trade-day="2016-07-14" trade-type="P" block-order="N" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
     <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
     <ProfileData profile-role="BC01">
       <Data period-from="0" period-to="1" value="1" unit="MW" seq-num="1"/>
     </ProfileData>
     <ProfileData profile-role="BP01">
       <Data period-from="0" period-to="1" value="21" unit="EUR" seq-num="1"/>
     </ProfileData>
     <ProfileData profile-role="BC01">
       <Data period-from="0" period-to="1" value="1" unit="MW" seq-num="2"/>
     </ProfileData>
     <ProfileData profile-role="BP01">
       <Data period-from="0" period-to="1" value="23" unit="EUR" seq-num="2"/>
     </ProfileData>
   </Trade>
```

```
    <!--Blokove objednavky, nakup, Base load-->
    <Trade trade-day="2016-07-14" trade-type="N" block-order="A" block-type="BL" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
      <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
      <ProfileData profile-role="BC01">
        <Data value="5" unit="MW" seq-num="1"/>
      </ProfileData>
      <ProfileData profile-role="BP01">
        <Data value="20" unit="EUR" seq-num="1"/>
      </ProfileData>
      <ProfileData profile-role="BC01">
        <Data value="1" unit="MW" seq-num="2"/>
      </ProfileData>
      <ProfileData profile-role="BP01">
        <Data value="18" unit="EUR" seq-num="2"/>
      </ProfileData>
    </Trade>
      <!--Blokove objednavky, predaj, Base load-->
      <Trade trade-day="2016-07-14" trade-type="P" block-order="A" block-type="BL" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
        <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
        <ProfileData profile-role="BC01">
            <Data value="2" unit="MW" seq-num="1"/>
        </ProfileData>
        <ProfileData profile-role="BP01">
          <Data value="22" unit="EUR" seq-num="1"/>
        </ProfileData>
        <ProfileData profile-role="BC01">
          <Data value="1" unit="MW" seq-num="2"/>
        </ProfileData>
        <ProfileData profile-role="BP01">
          <Data value="23" unit="EUR" seq-num="2"/>
        </ProfileData>
      </Trade>
      <!--Blokove objednavky, nakup, Peak load-->
      <Trade trade-day="2016-07-14" trade-type="N" block-order="A" block-type="PL" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
        <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
        <ProfileData profile-role="BC01">
            <Data value="4" unit="MW" seq-num="1"/>
        </ProfileData>
       <ProfileData profile-role="BP01">
            <Data value="17" unit="EUR" seq-num="1"/>
        </ProfileData>
        <ProfileData profile-role="BC01">
            <Data value="2" unit="MW" seq-num="2"/>
        </ProfileData>
        <ProfileData profile-role="BP01">
            <Data value="15" unit="EUR" seq-num="2"/>
        </ProfileData>
      </Trade>
      <!--Blokove objednavky, predaj, Peak load-->
      <Trade trade-day="2016-07-14" trade-type="P" block-order="A" block-type="PL" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
        <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
        <ProfileData profile-role="BC01">
            <Data value="3" unit="MW" seq-num="1"/>
        </ProfileData>
        <ProfileData profile-role="BP01">
            <Data value="18" unit="EUR" seq-num="1"/>
        </ProfileData>
      </Trade>
      <!--Blokove objednavky, nakup, Off-peak-->
      <Trade trade-day="2016-07-14" trade-type="N" block-order="A" block-type="OP" delivery-duration="60" market-area="SK" sett-curr="EUR" market="VDT">
        <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
        <ProfileData profile-role="BC01">
            <Data value="1" unit="MW" seq-num="1"/>
        </ProfileData>
        <ProfileData profile-role="BP01">
            <Data value="12" unit="EUR" seq-num="1"/>
        </ProfileData>
      </Trade>
```

```
        <!--Blokove objednavky, predaj, Off-peak-->
        <Trade trade-day="2016-07-14" trade-type="P" block-order="A" block-type="OP" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
            <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
            <ProfileData profile-role="BC01">
                <Data value="3" unit="MW" seq-num="1"/>
            </ProfileData>
            <ProfileData profile-role="BP01">
                <Data value="12.1" unit="EUR" seq-num="1"/>
            </ProfileData>
            <ProfileData profile-role="BC01">
                <Data value="3" unit="MW" seq-num="2"/>
            </ProfileData>
            <ProfileData profile-role="BP01">
                <Data value="12.5" unit="EUR" seq-num="2"/>
            </ProfileData>
        </Trade>
        <!--Blokove objednavky, nakup, vlastne-->
        <Trade trade-day="2016-07-13" trade-type="N" block-order="A" block-type="V" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
            <TimeData datetime="2016-07-13T09:30:10.123Z" datetime-type="DTO"/>
            <ProfileData profile-role="BC01" trade-id="2920CAF91042B1841B32D9E3E63E7C75">
                <Data period-from="16" period-to="19" value="1" unit="MW"/>
            </ProfileData>
            <ProfileData profile-role="BP01" trade-id="2920CAF91042B1841B32D9E3E63E7C75">
                <Data period-from="16" period-to="19" value="12" unit="EUR"/>
            </ProfileData>
        </Trade>
</ISOTEDATA>
```

**Example 45** Response with description of order book snapshot

*JSON for WebSocket*

The structure is used in the WebSocket interface to provide the complete order book.

*Orderbook-snapshot request*

Market participants can request the current snapshot of the order book when needed. Below is an example of connecting to WebSocket along with the corresponding message

Send Message:

```
Connect: wss://{hostname}/api/v1/idm/ws?topics=orderbook

{
   "type": "orderbook-snapshot"
}
```

**Example 46** This message allows the client to request the current state of the order book. (orderbook snapshot).

Received Message:

If the request is successfully processed, all available order book data is returned in JSON structure. The response includes the quantities of different types of orders, broken down by the period length of the respective product, limit prices, and by periods. Various prices within the period are sorted from the market participant's perspective, starting from the best price (highest for buy orders, lowest for sell orders) to the worst price (lowest for buy orders, highest for sell orders).

```
  "payload":{
    "seqNo":500,
    "timeDelta":1,
    "data":[
      {
        "period":{
          "start":"2024-11-19T17:30:00Z",
          "end":"2024-11-19T17:45:00Z",
          "isBlock":false,
          "tradingEnd":"2024-11-19T17:00:00Z"
        },
        "buyList":[
          {
            "price":75.3,
            "quantity":0.1,
            "ownQuantity":0
          },
          {
            "price":74.3,
            "quantity":5,
            "ownQuantity":0
          },
          {
            "price":74.2,
            "quantity":5,
            "ownQuantity":0
          },
          {
            "price":74,
            "quantity":64,
            "ownQuantity":0
          }
        ],
        "sellList":[
          {
            "price":79,
            "quantity":1,
            "ownQuantity":0
          },
          {
            "price":79.1,
            "quantity":2.5,
            "ownQuantity":0
          },
          {
            "price":79.3,
            "quantity":16,
            "ownQuantity":0
          },
          {
            "price":79.32,
            "quantity":0.8,
            "ownQuantity":0
          },
          {
            "price":80,
            "quantity":10,
            "ownQuantity":0
          }
        ],
        "blockOrders":[

        ]
      }
    ]
  },
  "type":"orderbook-snapshot"
}
```


**Example 47** Response with the description of the order book data (sample data for only one period for a 15-minute product, due to the message size).

### 4.4.3 Available cross-border transmission capacities data H2H (E-08_02)

Provision of available cross-border transmission capacities in format of Hub-to-Hub (H2H) is ensured by requests in structure CDSREQ-VDT.840 and responses in structure RESPONSE-VDT.811 (indication of success/unsuccess) and ISOTEDATA-VDT.812 (description of cross-border capacities data) or in JSON format throught the automated WEB API interface or WebSocket.

Response containing cross-border capacities includes all available data related to available cross-border capacities for specified time period.

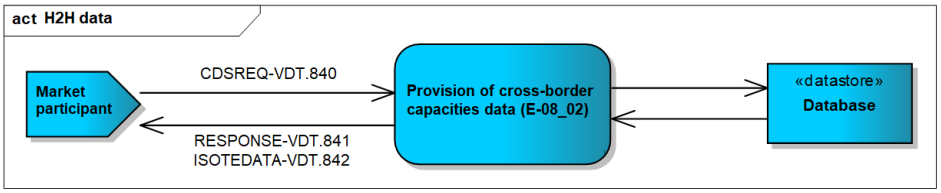


**Figure 28** Provision of cross-border capacities scheme

*CDSREQ-VDT.840*

Structure contains attribute message-code=840 within the header and has to be filled up according to specification of structure for provision of cross-border capacities data. Request for cross-border capacities data may be realized with optional possibility to specify relevant trading period(s) and the system automatically evaluates the current state of cross-border capacities data and provides them for required filter of periods. In case of not defining the information on the length of the period, all data will be made available broken down by business period. The data on the time slice will be made available to the market participant in the response together with the data.

```
<CDSREQ date-time="2019-01-01T07:00:00" dtd-release="1" dtd-version="1" id="45t" message-
code="840" xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
       <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
       <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
       <Trade trade-day="2022-09-27" period-from="12" period-to="13"/>
</CDSREQ>
```

**Example 48** Provision of available cross-border capacities for IDM (H2H)

*RESPONSE-VDT.841*

In accordance with specification of RESPONSE structure, response is given in case of successful or unsuccessful order processing. In this case message-code=841 can be found in the header.

```
<RESPONSE id="2bb436c9a4c84d3eb7c2002b4c8602bc" message-code="841" date-time="2022-09-
27T09:21:44Z" dtd-version="1" dtd-release="1"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
       <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
       <ReceiverIdentification id="24X--YOUR-EIC--B" coding-scheme="15"/>
       <Reference id="45t"/>
       <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 49** Response about successful processing of IDM cross-border capacities data provision request

*ISOTEDATA-VDT.842*

In case of successful order processing, all available cross-border capacities data in the form of H2H matrix are returned for a given time period and message-code=842 can be found in the header. Response includes available capacities broken down by relevant direction and cross-border profiles including the information related to granularity of the value (15/60-min product).

```
<ISOTEDATA id="22158dc52e74481bbf15ea90fcc1c24c" message-code="842" date-time="2022-09-27T09:21:44Z" dtd-version="1" dtd-release="1"
answer-required="false" xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
        <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04"/>
        <ReceiverIdentification id="11XAENSK-------B" coding-scheme="15" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04"/>
        <Reference id="45t" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04"/>
        <Trade trade-day="2022-09-27" market="VDT" delivery-duration="60" market-area="SK" area-from="SK" area-to="AT"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
                <TimeData datetime="2022-09-27T11:21:44.6815426+02:00" datetime-type="DTO"/>
                <ProfileData profile-role="AC01">
                        <Data period-from="0" period-to="1" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="0" period-to="1" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="1" period-to="2" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="1" period-to="2" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <!--...missing periods ...-->
                <ProfileData profile-role="AC01">
                        <Data period-from="21" period-to="22" value="1287" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="21" period-to="22" value="684" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="22" period-to="23" value="1134" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="22" period-to="23" value="764" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="23" period-to="24" value="1168" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="23" period-to="24" value="2098" unit="MW" seq-num="1"/>
                </ProfileData>
        </Trade>
        <Trade trade-day="2022-09-27" market="VDT" delivery-duration="60" market-area="SK" area-from="SK" area-to="BE"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
                <TimeData datetime="2022-09-27T11:21:44.6815426+02:00" datetime-type="DTO"/>
                <ProfileData profile-role="AC01">
                        <Data period-from="0" period-to="1" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="0" period-to="1" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="1" period-to="2" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="1" period-to="2" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="2" period-to="3" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="2" period-to="3" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <!--... missing periods ..-->
                <ProfileData profile-role="AC01">
                        <Data period-from="21" period-to="22" value="418" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="21" period-to="22" value="684" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="22" period-to="23" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="22" period-to="23" value="764" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC01">
                        <Data period-from="23" period-to="24" value="0" unit="MW" seq-num="1"/>
                </ProfileData>
                <ProfileData profile-role="AC02">
                        <Data period-from="23" period-to="24" value="845" unit="MW" seq-num="1"/>
                </ProfileData>
        </Trade>
</ISOTEDATA>
```

**Example 50** Response containing available cross-border capacities data (data sample)

*JSON for WEB API*

The structure is used in the Web API interface to provide data on cross-border capacities.

*API GET IDM hub-to-hub*

Requesting cross-border capacity data is performed with an optional specification of a specific period. Upon receiving this request, the system automatically evaluates the current state of cross-border capacity and provides it for the selected time slice of the period filter. If no period duration is defined, all data will be made available categorized by trading period. The time slice information will be provided to the market participant in the response along with the data. Data access is realized through the following request.

Request:

```
https://{hostname}/api/v1/idm/hub-to-hub?countryCodes=CZ&deliveryFrom=2024-11-18T13:00:00Z&deliveryTo=2024-11-18T23:00:00
```

**Example 51** URL for Accessing Cross-Border Capacities for the Intraday Market (IDM).

Response:

In case of successful processing of the request, the response will be returned according to the JSON structure with a status code of 200 OK..

```
[
  {
    "eic": "10YCZ-CEPS-----N",
    "areaName": "CEPS",
    "countryCode": "CZ",
    "deliveryDay": "2024-11-18",
    "deliveryStart": "2024-11-18T22:00:00Z",
    "deliveryEnd": "2024-11-18T23:00:00Z",
    "availableCapacityIn": 3066,
    "availableCapacityOut": 2798
  }
]
```

**Example 52** Response with a description of Cross-Border Capacity data (sample Data).

## 4.5 Administration of IDA orders

Market participant DAM order administration is carried out through operations of order reception and availability registered in ISOT via IdaOrders web service.

### 4.5.1 Processing level

Market participants enter their orders into ISOT by the deadline for order registration for specific auction at the latest. Orders can be entered into the system in advance, even several days in advance.

Market participant can register unlimited number of sale orders and unlimited number of purchase orders for a single trading day. Trading period within intraday auctions is specified to 15 minutes. Order can contain following characteristics:

- Standard order (order containing maximum of 25 blocks without the possibility to define total acceptance of block no. 1). This order type is defined by element *block-order="N".*
- Block order with one of the following types:

- simple profile block order with possibility to fill up one block containing the quantity for maximum of 96 (92/100 in case of clock change day) periods with definition of one weighted-average price, defined by *block-order="A" a block-type="SB"*.

In the case that market participant, using the modification of existing order, replaces already existing order, or removes the order from the system, it is not allowed to return to the previous version of the order i.e. not even in case that the original order was valid and the new order is invalid.

In case of invalid order, user is required to remove reasons for its invalidity by replacement with a new version or by deleting the existing order and creation of new one respectively. Orders non-compliant with the check at submission (invalid) shall be removed and will not enter the matching process.

Removal of already entered and accepted orders is carried out through submission of a new order with null values in the first block (paired values of amount/price).

### 4.5.2 Reception of orders (E-09_01)

Reception of orders is carried out by request for order submission in *ISOTEDATA.851* structure (message-code=851) and by response in *RESPONSE.812* structure (indication of success/fail) and *ISOTEDATA.813* structure (description of order registered in the system).

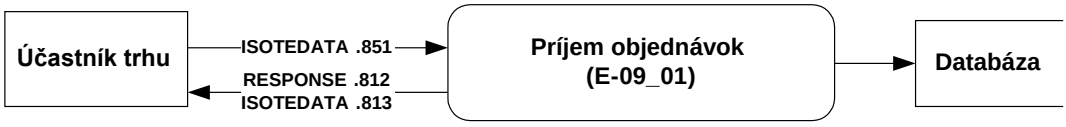


**Figure 29** Schéma príjmu objednávky VDA

*ISOTEDATA.811*

The structure contains message-code=811 attribute in the header and is filled out in accordance with order structure. Only a single order can be concurrently entered, i.e. submission of multiple orders is carried out through multiple calls.

```
<ISOTEDATA id="1" message-code="851" date-time="2023-12-01T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2023-12-01" auction-id="IDA1" trade-type="P" block-order="N" market-
area="SK" market="IDA" delivery-duration="15" sett-curr="EUR">
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MW" splitting="A" />
      <Data period="2" value="100.0" unit="MW" splitting="A" />
      <Data period="3" value="100.0" unit="MW" splitting="A" />
      <Data period="4" value="100.0" unit="MW" splitting="A" />
      <Data period="5" value="100.0" unit="MW" splitting="A" />
      <Data period="6" value="100.0" unit="MW" splitting="N" />
      <Data period="7" value="100.0" unit="MW" splitting="N" />
      <Data period="8" value="100.0" unit="MW" splitting="N" />
      <Data period="9" value="100.0" unit="MW" splitting="N" />
      <Data period="10" value="100.0" unit="MW" splitting="N" />
      <Data period="11" value="100.0" unit="MW" splitting="N" />
      <Data period="12" value="100.0" unit="MW" splitting="N" />
      <Data period="13" value="100.0" unit="MW" splitting="N" />
      <Data period="14" value="100.0" unit="MW" splitting="N" />
      <Data period="15" value="100.0" unit="MW" splitting="N" />
      <Data period="16" value="100.0" unit="MW" splitting="N" />
      <Data period="17" value="100.0" unit="MW" splitting="N" />
      <Data period="18" value="100.0" unit="MW" splitting="N" />
      <Data period="19" value="100.0" unit="MW" splitting="N" />
      <Data period="20" value="100.0" unit="MW" splitting="N" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A" />
      <Data period="2" value="15.00" unit="EUR" splitting="A" />
      <Data period="3" value="15.00" unit="EUR" splitting="A" />
      <Data period="4" value="15.00" unit="EUR" splitting="A" />
      <Data period="5" value="15.00" unit="EUR" splitting="A" />
      <Data period="6" value="15.00" unit="EUR" splitting="N" />
      <Data period="7" value="15.00" unit="EUR" splitting="N" />
      <Data period="8" value="15.00" unit="EUR" splitting="N" />
      <Data period="9" value="15.00" unit="EUR" splitting="N" />
      <Data period="10" value="15.00" unit="EUR" splitting="N" />
      <Data period="11" value="15.00" unit="EUR" splitting="N" />
      <Data period="12" value="15.00" unit="EUR" splitting="N" />
      <Data period="13" value="15.00" unit="EUR" splitting="N" />
      <Data period="14" value="15.00" unit="EUR" splitting="N" />
      <Data period="15" value="15.00" unit="EUR" splitting="N" />
      <Data period="16" value="15.00" unit="EUR" splitting="N" />
      <Data period="17" value="15.00" unit="EUR" splitting="N" />
      <Data period="18" value="15.00" unit="EUR" splitting="N" />
      <Data period="19" value="15.00" unit="EUR" splitting="N" />
      <Data period="20" value="15.00" unit="EUR" splitting="N" />
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 53** Submission of standard IDA order for sale

```
<ISOTEDATA id="1" message-code="851" date-time="2023-12-01T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2023-12-01" auction-id="IDA1" trade-type="P" block-order="A" block-
type="SB" market-area="SK" market="IDA" delivery-duration="15" sett-curr="EUR">
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MW" splitting="A" />
      <Data period="2" value="100.0" unit="MW" splitting="A" />
      <Data period="3" value="100.0" unit="MW" splitting="A" />
      <Data period="4" value="100.0" unit="MW" splitting="A" />
      <Data period="5" value="100.0" unit="MW" splitting="A" />
      <Data period="6" value="100.0" unit="MW" splitting="N" />
      <Data period="7" value="100.0" unit="MW" splitting="N" />
      <Data period="8" value="100.0" unit="MW" splitting="N" />
      <Data period="9" value="100.0" unit="MW" splitting="N" />
      <Data period="10" value="100.0" unit="MW" splitting="N" />
      <Data period="11" value="100.0" unit="MW" splitting="N" />
      <Data period="12" value="100.0" unit="MW" splitting="N" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A" />
      <Data period="2" value="15.00" unit="EUR" splitting="A" />
      <Data period="3" value="15.00" unit="EUR" splitting="A" />
      <Data period="4" value="15.00" unit="EUR" splitting="A" />
      <Data period="5" value="15.00" unit="EUR" splitting="A" />
      <Data period="6" value="15.00" unit="EUR" splitting="N" />
      <Data period="7" value="15.00" unit="EUR" splitting="N" />
      <Data period="8" value="15.00" unit="EUR" splitting="N" />
      <Data period="9" value="15.00" unit="EUR" splitting="N" />
      <Data period="10" value="15.00" unit="EUR" splitting="N" />
      <Data period="11" value="15.00" unit="EUR" splitting="N" />
      <Data period="12" value="15.00" unit="EUR" splitting="N" />
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 54** Submission of simple block IDA order for sale

*RESPONSE.812*

In case of successful or failed order processing, response is returned in accordance with specification of RESPONSE structure, with *message-code=812* in the header. Identification, under which the order is registered in the system, is returned in *Reason/@trade-id* attribute.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
       date-time="2023-12-01T13:46:26Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03" trade-id="1016"/>
</RESPONSE>
```

**Example 55** Response on success of IDA order submission

*ISOTEDATA.813*

In case of successful order processing, the structure is returned as it was registered in the system, where *message-code=813* can be found in the header. Identification and version, under which the order is registered in the system, are returned in *Trade/@id* and *Trade/@version* attribute. Own order data can be retrieved back based on this identification (see E-09_02).

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="813"
       date-time="2023-12-01T13:46:26Z" dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade id="1016" trade-day="2023-12-01" auction-id="IDA1" version="1" trade-type="P" block-
order="N" trade-stage="P" sett-curr="EUR" market-area="SK" market="IDA" delivery-duration="15">
    <TimeData datetime="2023-12-01T13:46:26Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MW" splitting="A"/>
      <Data period="2" value="100.0" unit="MW" splitting="A"/>
      <Data period="3" value="100.0" unit="MW" splitting="A"/>
      <Data period="4" value="100.0" unit="MW" splitting="A"/>
      <Data period="5" value="100.0" unit="MW" splitting="A"/>
      <Data period="6" value="100.0" unit="MW" splitting="N"/>
      <Data period="7" value="100.0" unit="MW" splitting="N"/>
      <Data period="8" value="100.0" unit="MW" splitting="N"/>
      <Data period="9" value="100.0" unit="MW" splitting="N"/>
      <Data period="10" value="100.0" unit="MW" splitting="N"/>
      <Data period="11" value="100.0" unit="MW" splitting="N" />
      <Data period="12" value="100.0" unit="MW" splitting="N" />
      <Data period="13" value="100.0" unit="MW" splitting="N" />
      <Data period="14" value="100.0" unit="MW" splitting="N" />
      <Data period="15" value="100.0" unit="MW" splitting="N" />
      <Data period="16" value="100.0" unit="MW" splitting="N" />
      <Data period="17" value="100.0" unit="MW" splitting="N" />
      <Data period="18" value="100.0" unit="MW" splitting="N" />
      <Data period="19" value="100.0" unit="MW" splitting="N" />
      <Data period="20" value="100.0" unit="MW" splitting="N" />

    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A"/>
      <Data period="2" value="15.00" unit="EUR" splitting="A"/>
      <Data period="3" value="15.00" unit="EUR" splitting="A"/>
      <Data period="4" value="15.00" unit="EUR" splitting="A"/>
      <Data period="5" value="15.00" unit="EUR" splitting="A"/>
      <Data period="6" value="15.00" unit="EUR" splitting="N"/>
      <Data period="7" value="15.00" unit="EUR" splitting="N"/>
      <Data period="8" value="15.00" unit="EUR" splitting="N"/>
      <Data period="9" value="15.00" unit="EUR" splitting="N"/>
      <Data period="10" value="15.00" unit="EUR" splitting="N"/>
      <Data period="11" value="15.00" unit="EUR" splitting="N" />
      <Data period="12" value="15.00" unit="EUR" splitting="N" />
      <Data period="13" value="15.00" unit="EUR" splitting="N" />
      <Data period="14" value="15.00" unit="EUR" splitting="N" />
      <Data period="15" value="15.00" unit="EUR" splitting="N" />
      <Data period="16" value="15.00" unit="EUR" splitting="N" />
      <Data period="17" value="15.00" unit="EUR" splitting="N" />
      <Data period="18" value="15.00" unit="EUR" splitting="N" />
      <Data period="19" value="15.00" unit="EUR" splitting="N" />
      <Data period="20" value="15.00" unit="EUR" splitting="N" />
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 56** Response with IDA order description

### 4.5.3 Removal of orders (E-09_01)

Removal of order is carried out by submission of specific order in *ISOTEDATA.851* structure (message-code=851) containing null values for amount and price in the first block for all periods of a given trading day. Response to order removal is returned in *RESPONSE.812* structure (indication of success/failure) and *ISOTEDATA.813* structure (description of order that was removed from the system).

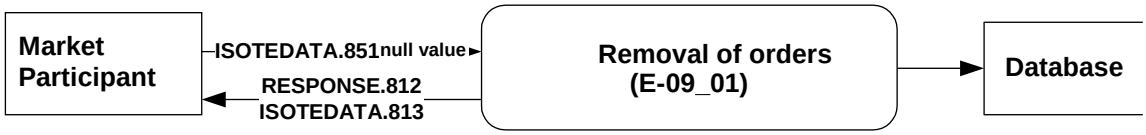


**Figure 30** Schema for IDA order removal

*ISOTEDATA.851*

The structure contains *message-code=851* attribute in the header and is filled out in accordance with order structure specification, where only the 1. block is entered containing null values for amount and price. Request can be entered for removal of orders for specific trading day and auction (*trade-day* and *auction-id* attribute), removal of specific auction and order type (*trade-day* and *trade-type* attributes) or removal of specific order via order identification of relevant order (*id* attribute returned in responses at order submission).

```
<ISOTEDATA id="1" message-code="851" date-time="2023-12-01T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade id="1016" trade-day="2023-12-01" auction-id="IDA1" trade-type="P" market-area="SK"
market="IDA" delivery-duration="15" sett-curr="EUR">
    <ProfileData profile-role="BC01">
      <Data period="1" value="0.0" unit="MW" splitting="A"/>
      <Data period="2" value="0.0" unit="MW" splitting="A"/>
      <Data period="3" value="0.0" unit="MW" splitting="A"/>
      <Data period="4" value="0.0" unit="MW" splitting="A"/>
      <Data period="5" value="0.0" unit="MW" splitting="A"/>
      <Data period="6" value="0.0" unit="MW" splitting="A"/>
      <Data period="7" value="0.0" unit="MW" splitting="A"/>
      <Data period="8" value="0.0" unit="MW" splitting="A"/>
      <Data period="9" value="0.0" unit="MW" splitting="A"/>
      <Data period="10" value="0.0" unit="MW" splitting="A"/>
      <Data period="11" value="0.0" unit="MW" splitting="A"/>
      <Data period="12" value="0.0" unit="MW" splitting="A"/>
      <Data period="13" value="0.0" unit="MW" splitting="A"/>
      <Data period="14" value="0.0" unit="MW" splitting="A"/>
      <Data period="15" value="0.0" unit="MW" splitting="A"/>
      <Data period="16" value="0.0" unit="MW" splitting="A"/>
      <Data period="17" value="0.0" unit="MW" splitting="A"/>
      <Data period="18" value="0.0" unit="MW" splitting="A"/>
      <Data period="19" value="0.0" unit="MW" splitting="A"/>
      <Data period="20" value="0.0" unit="MW" splitting="A"/>
      <Data period="21" value="0.0" unit="MW" splitting="A"/>
      <Data period="22" value="0.0" unit="MW" splitting="A"/>
      <Data period="23" value="0.0" unit="MW" splitting="A"/>
      <Data period="24" value="0.0" unit="MW" splitting="A"/>
      <Data period="25" value="0.0" unit="MW" splitting="A"/>
       ......
      <Data period="94" value="0.0" unit="MW" splitting="A"/>
      <Data period="95" value="0.0" unit="MW" splitting="A"/>
      <Data period="96" value="0.0" unit="MW" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="0.0" unit="EUR" splitting="A"/>
      <Data period="2" value="0.0" unit="EUR" splitting="A"/>
      <Data period="3" value="0.0" unit="EUR" splitting="A"/>
      <Data period="4" value="0.0" unit="EUR" splitting="A"/>
      <Data period="5" value="0.0" unit="EUR" splitting="A"/>
      <Data period="6" value="0.0" unit="EUR" splitting="A"/>
      <Data period="7" value="0.0" unit="EUR" splitting="A"/>
      <Data period="8" value="0.0" unit="EUR" splitting="A"/>
      <Data period="9" value="0.0" unit="EUR" splitting="A"/>
      <Data period="10" value="0.0" unit="EUR" splitting="A"/>
      <Data period="11" value="0.0" unit="EUR" splitting="A"/>
      <Data period="12" value="0.0" unit="EUR" splitting="A"/>
      <Data period="13" value="0.0" unit="EUR" splitting="A"/>
      <Data period="14" value="0.0" unit="EUR" splitting="A"/>
      <Data period="15" value="0.0" unit="EUR" splitting="A"/>
      <Data period="16" value="0.0" unit="EUR" splitting="A"/>
      <Data period="17" value="0.0" unit="EUR" splitting="A"/>
      <Data period="18" value="0.0" unit="EUR" splitting="A"/>
      <Data period="19" value="0.0" unit="EUR" splitting="A"/>
      <Data period="20" value="0.0" unit="EUR" splitting="A"/>
      <Data period="21" value="0.0" unit="EUR" splitting="A"/>
      <Data period="22" value="0.0" unit="EUR" splitting="A"/>
      <Data period="23" value="0.0" unit="EUR" splitting="A"/>
      <Data period="24" value="0.0" unit="EUR" splitting="A"/>
      <Data period="25" value="0.0" unit="EUR" splitting="A"/>
       ......
      <Data period="94" value="0.0" unit="EUR" splitting="A"/>
      <Data period="95" value="0.0" unit="EUR" splitting="A"/>
      <Data period="96" value="0.0" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 57** Removal of specific IDA order for sale

*RESPONSE.812*

In accordance with specification of RESPONSE structure, response is given in case of successful or unsuccessful order processing. In this case message-code=812 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
        date-time="2023-12-01T13:46:26Z" dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 58** Response on success of IDA order removal

*ISOTEDATA.813*

In case of successful order removal, the original order that was removed from the system will be returned. Message-code=813 can be found in the header.

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="813"
        date-time="2023-12-01T13:46:26Z" dtd-version="1" dtd-release="1" answer-required="false"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade id="1016" trade-day="2023-12-01" auction-id="IDA1" version="1" trade-type="P" block-
order="N" trade-stage="P" sett-curr="EUR" market-area="SK" market="IDA" delivery-duration="15">
    <TimeData datetime="2023-12-01T13:46:26Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="100.0" unit="MWH" splitting="A"/>
      <Data period="2" value="100.0" unit="MWH" splitting="A"/>
      <Data period="3" value="100.0" unit="MWH" splitting="A"/>
      <Data period="4" value="100.0" unit="MWH" splitting="A"/>
      <Data period="5" value="100.0" unit="MWH" splitting="A"/>
      <Data period="6" value="100.0" unit="MWH" splitting="N"/>
      <Data period="7" value="100.0" unit="MWH" splitting="N"/>
      <Data period="8" value="100.0" unit="MWH" splitting="N"/>
      <Data period="9" value="100.0" unit="MWH" splitting="N"/>
      <Data period="10" value="100.0" unit="MWH" splitting="N"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="15.00" unit="EUR" splitting="A"/>
      <Data period="2" value="15.00" unit="EUR" splitting="A"/>
      <Data period="3" value="15.00" unit="EUR" splitting="A"/>
      <Data period="4" value="15.00" unit="EUR" splitting="A"/>
      <Data period="5" value="15.00" unit="EUR" splitting="A"/>
      <Data period="6" value="15.00" unit="EUR" splitting="N"/>
      <Data period="7" value="15.00" unit="EUR" splitting="N"/>
      <Data period="8" value="15.00" unit="EUR" splitting="N"/>
      <Data period="9" value="15.00" unit="EUR" splitting="N"/>
      <Data period="10" value="15.00" unit="EUR" splitting="N"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 59** Response with description of removed IDA order

### 4.5.4 Order modification (E-09_01)

Modification of order is carried out by entry of specific order that is to be modified in *ISOTEDATA.811* structure (message-code=851) containing updated values for amount and price for desired trading periods of a given trading day. Response to order removal is returned in *RESPONSE.812* structure (indication of success/failure) and *ISOTEDATA.813* structure (description of modified order saved in the system). Order header modification is not possible. In order to modify the order header (i.e. type or direction), delete the existing order and submit a new one with required parameters.

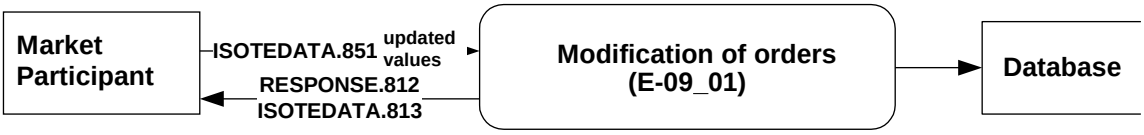


**Figure 31** Schéma modifikácie objednávky ÚT

*ISOTEDATA.851*

The structure contains *message-code=851* attribute in the header and is filled out in accordance with order structure specification. The order that is to be modified must be identified by specification of its ID in the attribute *Trade/id*. Only a single order can be concurrently modified, i.e. modification of multiple orders is carried out through multiple calls.

```
<ISOTEDATA id="1" message-code="851" date-time="2023-12-01T00:00:00" dtd-version="1" dtd-
release="1" answer-required="false"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade id="1016" trade-day="2023-12-01" auction-id="IDA1" trade-type="P" block-order="N"
market-area="SK" market="IDA" delivery-duration="15" sett-curr="EUR">
    <ProfileData profile-role="BC01">
      <Data period="1" value="10.0" unit="MW" splitting="A"/>
      <Data period="2" value="10.0" unit="MW" splitting="A"/>
      <Data period="3" value="10.0" unit="MW" splitting="A"/>
      <Data period="4" value="10.0" unit="MW" splitting="A"/>
      <Data period="5" value="20.0" unit="MW" splitting="A"/>
      <Data period="6" value="20.0" unit="MW" splitting="A"/>
      <Data period="7" value="20.0" unit="MW" splitting="A"/>
      <Data period="8" value="10.0" unit="MW" splitting="A"/>
      <Data period="9" value="10.0" unit="MW" splitting="A"/>
      <Data period="10" value="10.0" unit="MW" splitting="A"/>
      <Data period="11" value="10.0" unit="MW" splitting="A"/>
      <Data period="12" value="10.0" unit="MW" splitting="A"/>
      <Data period="13" value="10.0" unit="MW" splitting="A"/>
      <Data period="14" value="10.0" unit="MW" splitting="A"/>
      <Data period="15" value="20.0" unit="MW" splitting="A"/>
      <Data period="16" value="20.0" unit="MW" splitting="A"/>
      <Data period="17" value="20.0" unit="MW" splitting="A"/>
      <Data period="18" value="20.0" unit="MW" splitting="A"/>
      <Data period="19" value="20.0" unit="MW" splitting="A"/>
      <Data period="20" value="20.0" unit="MW" splitting="A"/>
      <Data period="21" value="10.0" unit="MW" splitting="A"/>
      <Data period="22" value="10.0" unit="MW" splitting="A"/>
      <Data period="23" value="10.0" unit="MW" splitting="A"/>
      <Data period="24" value="10.0" unit="MW" splitting="A"/>
      <Data period="25" value="10.0" unit="MW" splitting="A"/>
       ...
      <Data period="95" value="12.0" unit="MW" splitting="A"/>
      <Data period="96" value="12.0" unit="MW" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="50.0" unit="EUR" splitting="A"/>
      <Data period="2" value="50.0" unit="EUR" splitting="A"/>
      <Data period="3" value="50.0" unit="EUR" splitting="A"/>
      <Data period="4" value="50.0" unit="EUR" splitting="A"/>
      <Data period="5" value="50.0" unit="EUR" splitting="A"/>
      <Data period="6" value="50.0" unit="EUR" splitting="A"/>
      <Data period="7" value="50.0" unit="EUR" splitting="A"/>
      <Data period="8" value="50.0" unit="EUR" splitting="A"/>
      <Data period="9" value="50.0" unit="EUR" splitting="A"/>
      <Data period="10" value="50.0" unit="EUR" splitting="A"/>
      <Data period="11" value="50.0" unit="EUR" splitting="A"/>
      <Data period="12" value="50.0" unit="EUR" splitting="A"/>
      <Data period="13" value="50.0" unit="EUR" splitting="A"/>
      <Data period="14" value="50.0" unit="EUR" splitting="A"/>
      <Data period="15" value="50.0" unit="EUR" splitting="A"/>
      <Data period="16" value="50.0" unit="EUR" splitting="A"/>
      <Data period="17" value="50.0" unit="EUR" splitting="A"/>
      <Data period="18" value="50.0" unit="EUR" splitting="A"/>
      <Data period="19" value="50.0" unit="EUR" splitting="A"/>
      <Data period="20" value="50.0" unit="EUR" splitting="A"/>
      <Data period="21" value="50.0" unit="EUR" splitting="A"/>
      <Data period="22" value="50.0" unit="EUR" splitting="A"/>
      <Data period="23" value="50.0" unit="EUR" splitting="A"/>
      <Data period="24" value="50.0" unit="EUR" splitting="A"/>
      <Data period="25" value="10.0" unit=" EUR " splitting="A"/>
       ...
      <Data period="95" value="30.0" unit="EUR" splitting="A"/>
      <Data period="96" value="30.0" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO" />
  </Trade>
</ISOTEDATA>
```

**Example 60** IDA order modification

*RESPONSE.812*

In accordance with specification of RESPONSE structure, response is given in case of successful or unsuccessful order processing. In this case message-code=812 can be found in the header.

```
<RESPONSE id="cb4d980f-2f9a-4be7-96ef-850be04b214" message-code="812"
       date-time="2023-12-01T13:46:26Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="1"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 61** Response on success of IDA order modification

*ISOTEDATA.813*

In case of successful order modification, the succesfully modified order will be returned with increased version in comparison to original order. Message-code=813 can be found in the header. Within the Trade/@id and Trade/@version attributes are returned ID and version under which the order is saved. Based on these data it is possible to request the own order data (see E-09_02).

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="813"
       date-time="2023-12-01T13:46:26Z" dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Reference id="1"/>
  <Trade id="1016" trade-day="2023-12-01" auction-id="IDA1"  version="2" trade-type="P" block-
order="N" trade-stage="P" sett-curr="EUR" market-area="SK" market="IDA" delivery-duration="15" >
    <TimeData datetime="2023-12-01T13:46:26Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="10.0" unit="MW" splitting="A"/>
      <Data period="2" value="10.0" unit="MW" splitting="A"/>
      <Data period="3" value="10.0" unit="MW" splitting="A"/>
      <Data period="4" value="10.0" unit="MW" splitting="A"/>
      <Data period="5" value="20.0" unit="MW" splitting="A"/>
      <Data period="6" value="20.0" unit="MW" splitting="A"/>
      <Data period="7" value="20.0" unit="MW" splitting="A"/>
      <Data period="8" value="10.0" unit="MW" splitting="A"/>
      <Data period="9" value="10.0" unit="MW" splitting="A"/>
      <Data period="10" value="10.0" unit="MW" splitting="A"/>
      <Data period="11" value="10.0" unit="MW" splitting="A"/>
      <Data period="12" value="10.0" unit="MW" splitting="A"/>
      <Data period="13" value="10.0" unit="MW" splitting="A"/>
      <Data period="14" value="10.0" unit="MW" splitting="A"/>
      <Data period="15" value="20.0" unit="MW" splitting="A"/>
      <Data period="16" value="20.0" unit="MW" splitting="A"/>
      <Data period="17" value="20.0" unit="MW" splitting="A"/>
      <Data period="18" value="20.0" unit="MW" splitting="A"/>
      <Data period="19" value="20.0" unit="MW" splitting="A"/>
      <Data period="20" value="20.0" unit="MW" splitting="A"/>
      <Data period="21" value="10.0" unit="MW" splitting="A"/>
      <Data period="22" value="10.0" unit="MW" splitting="A"/>
      <Data period="23" value="10.0" unit="MW" splitting="A"/>
      <Data period="24" value="10.0" unit="MW" splitting="A"/>
      <Data period="25" value="10.0" unit="MW" splitting="A"/>
       ...
      <Data period="95" value="12.0" unit="MW" splitting="A"/>
      <Data period="96" value="12.0" unit="MW" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="50.0" unit="EUR" splitting="A"/>
      <Data period="2" value="50.0" unit="EUR" splitting="A"/>
      <Data period="3" value="50.0" unit="EUR" splitting="A"/>
      <Data period="4" value="50.0" unit="EUR" splitting="A"/>
      <Data period="5" value="50.0" unit="EUR" splitting="A"/>
      <Data period="6" value="50.0" unit="EUR" splitting="A"/>
      <Data period="7" value="50.0" unit="EUR" splitting="A"/>
      <Data period="8" value="50.0" unit="EUR" splitting="A"/>
      <Data period="9" value="50.0" unit="EUR" splitting="A"/>
      <Data period="10" value="50.0" unit="EUR" splitting="A"/>
      <Data period="11" value="50.0" unit="EUR" splitting="A"/>
      <Data period="12" value="50.0" unit="EUR" splitting="A"/>
      <Data period="13" value="50.0" unit="EUR" splitting="A"/>
      <Data period="14" value="50.0" unit="EUR" splitting="A"/>
      <Data period="15" value="50.0" unit="EUR" splitting="A"/>
      <Data period="16" value="50.0" unit="EUR" splitting="A"/>
      <Data period="17" value="50.0" unit="EUR" splitting="A"/>
      <Data period="18" value="50.0" unit="EUR" splitting="A"/>
      <Data period="19" value="50.0" unit="EUR" splitting="A"/>
      <Data period="20" value="50.0" unit="EUR" splitting="A"/>
      <Data period="21" value="50.0" unit="EUR" splitting="A"/>
      <Data period="22" value="50.0" unit="EUR" splitting="A"/>
      <Data period="23" value="50.0" unit="EUR" splitting="A"/>
      <Data period="24" value="50.0" unit="EUR" splitting="A"/>
      <Data period="25" value="10.0" unit=" EUR " splitting="A"/>
       ...
      <Data period="95" value="30.0" unit="EUR" splitting="A"/>
      <Data period="96" value="30.0" unit="EUR" splitting="A"/>

    </ProfileData>
    <Party id="24X-ENTRADE-SK-9" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 62** Response with description of modified IDA order

### 4.5.5 Provision of orders (E-09_02)

Provision of orders is carried out by request for order retrieval in *CDSREQ.831* structure (message-code=831) and response in *RESPONSE.832* structure (indication of success/failure) and ISOTEDATA.833 *structure* (description of order registered in the system).


**Figure 32** IDA order provision schema

*CDSREQ.831*

Request can be formulated either for specific auction (trade-day and auction-id) or for specific order (id and version) and is filled out according to the specification of CDSREQ structure. Order identification has precedence over trading day and auction.

```
<CDSREQ id="4a6s5d45f" message-code="831" date-time="2023-12-01T01:18:33" dtd-version="1" dtd-
release="1" xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2023-12-01" auction-id="IDA1"/>
</CDSREQ>
```

**Example 63** Request for provision of IDA orders for specific auction

*RESPONSE.832*

In case of successful or failed request processing, response is returned in accordance with specification of RESPONSE structure, where message-code=832 can be found in the header.

```
<RESPONSE id="a9e40366-ad70-45ac-8b36-bd8fbce5ef7" message-code="832"
       date-time="2023-12-01T14:02:36Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15" />
  <Reference id="4a6s5d45f"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 64** Response on success of IDA order data provision for specific auction

*ISOTEDATA.833*

In case of successful request processing, the structure is returned as it was registered in the system and message-code=833 can be found in the header. Identification and version, under which the order is registered in the system, are returned in *Trade/@id* and *Trade/@version* attribute. If there is a single purchase order and a single sale order registered in the system for a given trading day and auction, return structure of request for specific trading day contains two orders (either valid or invalid)

```
<ISOTEDATA id="1" message-code="833" date-time="2009-07-03T14:02:36Z"
       dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/orders/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V 11XKORLEAINVESTY" coding-scheme="15"/>
  <ReceiverIdentification id="24X-ENTRADE-SK-9" coding-scheme="15"/>
  <Reference id="4a6s5d45f"/>
  <Trade id="123" trade-day="2023-12-01" auction-id="IDA1" trade-type="P" block-order="N" trade-
stage="P" sett-curr="EUR" market-area="SK" market="IDA" delivery-duration="15">
    <TimeData datetime="2023-12-01T09:00:43Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period="1" value="5.0" unit="MW" splitting="A"/>
      <Data period="2" value="5.0" unit="MW" splitting="A"/>
      <Data period="3" value="5.0" unit="MW" splitting="A"/>
      <Data period="4" value="5.0" unit="MW" splitting="A"/>
      <Data period="5" value="5.0" unit="MW" splitting="A"/>
      <Data period="6" value="5.0" unit="MW" splitting="A"/>
      <Data period="7" value="5.0" unit="MW" splitting="A"/>
      <Data period="8" value="5.0" unit="MW" splitting="A"/>
      <Data period="9" value="5.0" unit="MW" splitting="A"/>
      <Data period="10" value="5.0" unit="MW" splitting="A"/>
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period="1" value="24.00" unit="EUR" splitting="A"/>
      <Data period="2" value="24.00" unit="EUR" splitting="A"/>
      <Data period="3" value="24.00" unit="EUR" splitting="A"/>
      <Data period="4" value="24.00" unit="EUR" splitting="A"/>
      <Data period="5" value="24.00" unit="EUR" splitting="A"/>
      <Data period="6" value="24.00" unit="EUR" splitting="A"/>
      <Data period="7" value="24.00" unit="EUR" splitting="A"/>
      <Data period="8" value="24.00" unit="EUR" splitting="A"/>
      <Data period="9" value="24.00" unit="EUR" splitting="A"/>
      <Data period="10" value="24.00" unit="EUR" splitting="A"/>
    </ProfileData>
    <Party id="" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 65** Response containing IDA orders for specific auction

## 4.6 DM results and evaluations

Results and evaluations of day-ahead market are available for market participants via operations for DM results and evaluation retrieval for every periods in 15-minute or 60-minute granularity. or a whole day.

### 4.6.1 Processing level

Results of day-ahead market are available immediately after order matching and contain accepted amount and final marginal price (system or area price). Market participant is notified about results availability via ISOT system.

Evaluations of day-ahead market are available immediately after clearing of day-ahead market in the form of summary day-ahead evaluation as well as detailed evaluation per periods. Evaluations contain market organizer obligations and receivables towards a market participant. (receivables are stated with a negative sign). Market participant is notified about evaluations availability via ISOT system. Results of day-ahead market are made available in ISOT.

### 4.6.2 Notification of results for market participants (E-03_02)

Notification of results for entities is carried out by request in *CDSREQ.941* structure (message-code=941) and response with data in *RESPONSE.942* and *ISOTEDATA.943* structure.

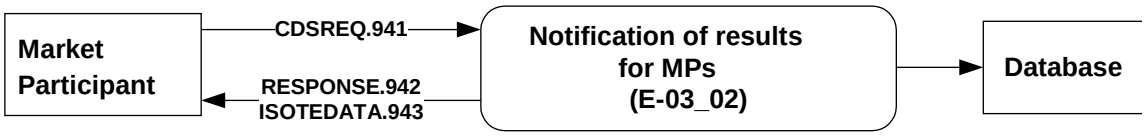


**Figure 33** Notification schema of DM results for market participant

### *CDSREQ.941*

Request is formulated for specific trading day (trade-day) and is filled out in accordance with the specification of CDSREQ structure.

```
<CDSREQ id="45t" message-code="941" date-time="2014-09-19T01:18:33"
        dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2009-09-21"/>
</CDSREQ>
```

**Example 66** Request for retrieval of results for specific day

### *RESPONSE.942*

In case of successful or failed request processing, response is returned in accordance with specification of RESPONSE structure, where message-code=942 can be found in the header.

```
<RESPONSE id="bd12362f-361b-4085-ade0-9ed678efff1" message-code="942"
        date-time="2009-07-03T14:11:43Z" dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 67** Response on success of results retrieval for a specific day

### *ISOTEDATA.943*

In case of successful request processing, the structure of DM results is returned as it was registered in the system, where message-code=943 can be found in the header.

Within the ProfileData/@profile-role attribute are used following types of results:

- SC19 – quantity of purchased electricity for positive prices,
- SC20 – quantity of sold electricity for positive prices,
- SC92 – quantity of purchased electricity for negative prices,
- SC93 – quantity of sold electricity for negative prices,
- SP20 – marginal price.

```
<ISOTEDATA id="ec1b50c0-afe1-4f5e-b6a1-d94c365099e" message-code="943"
       date-time="2009-07-03T14:11:43Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2009-09-21" delivery-duration="60">
    <ProfileData profile-role="SC19">
      <Data period="1" value="50" unit="MWH"/>
      <Data period="2" value="23" unit="MWH"/>
      <Data period="3" value="65" unit="MWH"/>
      <Data period="4" value="45" unit="MWH"/>
      <Data period="5" value="12.6" unit="MWH"/>
      <Data period="6" value="65" unit="MWH"/>
      <Data period="7" value="98" unit="MWH"/>
      <Data period="8" value="78" unit="MWH"/>
      <Data period="9" value="45" unit="MWH"/>
      <Data period="10" value="41" unit="MWH"/>
      <Data period="11" value="42" unit="MWH"/>
      <Data period="12" value="12" unit="MWH"/>
      <Data period="13" value="65" unit="MWH"/>
      <Data period="14" value="31.1" unit="MWH"/>
      <Data period="15" value="32.5" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SC20">
      <Data period="1" value="0" unit="MWH"/>
      <Data period="2" value="64.3" unit="MWH"/>
      <Data period="3" value="0" unit="MWH"/>
      <Data period="4" value="0" unit="MWH"/>
      <Data period="5" value="0" unit="MWH"/>
      <Data period="6" value="23.4" unit="MWH"/>
      <Data period="7" value="78.9" unit="MWH"/>
      <Data period="8" value="0" unit="MWH"/>
      <Data period="9" value="0" unit="MWH"/>
      <Data period="10" value="30.1" unit="MWH"/>
      <Data period="11" value="0" unit="MWH"/>
      <Data period="12" value="0" unit="MWH"/>
      <Data period="13" value="0" unit="MWH"/>
      <Data period="14" value="50" unit="MWH"/>
      <Data period="15" value="40" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP20">
      <Data period="1" value="20.45" unit="EUR"/>
      <Data period="2" value="32.45" unit="EUR"/>
      <Data period="3" value="20" unit="EUR"/>
      <Data period="4" value="20" unit="EUR"/>
      <Data period="5" value="21.65" unit="EUR"/>
      <Data period="6" value="24.95" unit="EUR"/>
      <Data period="7" value="24.35" unit="EUR"/>
      <Data period="8" value="32.65" unit="EUR"/>
      <Data period="9" value="17.65" unit="EUR"/>
      <Data period="10" value="24.87" unit="EUR"/>
      <Data period="11" value="23.98" unit="EUR"/>
      <Data period="12" value="15.45" unit="EUR"/>
      <Data period="13" value="19.87" unit="EUR"/>
      <Data period="14" value="33.54" unit="EUR"/>
      <Data period="15" value="17.65" unit="EUR"/>
    </ProfileData>
    <Party id="24XDSO---------Q" role="TO"/>
    <ResultStatus status="F"/>
  </Trade>
</ISOTEDATA>
```

**Example 68** Response containing requested results

### 4.6.3 Notification of evaluations per periods (E-05_01)

Notification of evaluations per hours is carried out by sending a request in *CDSREQ.951* structure (message-code=951) and response with data in *RESPONSE.952* and *ISOTEDATA.953* structures.

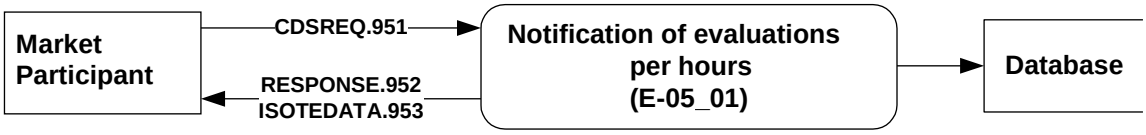


**Figure 34** Notification schema of evaluations per hours for market participant

*CDSREQ.951*

To be filled out in accordance with specification of CDSREQ structure. Request is formulated for specific trading day (trade-day).

```
<CDSREQ id="45t" message-code="951" date-time="2014-09-19T01:18:33"
        dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2009-09-21"/>
</CDSREQ>
```

**Example 69** Request for retrieval of evaluations per periods for specific day

*RESPONSE.952*

In case of successful or failed request processing, response is returned in accordance with specification of RESPONSE structure, where message-code=952 can be found in the header.

```
<RESPONSE id="7cdd21c0-e21f-4e70-a617-2d55db510e8" message-code="952"
        date-time="2009-07-03T14:16:54Z" dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 70** Response on success of evaluations retrieval per hours for a specific day

*ISOTEDATA.953*

In the event of successful request processing, the structure of trading results is returned as it was registered in the system, where message-code=953 can be found in the header.

The following result types are used in *ProfileData/@profile-role* attribute:

- SP02 – clearing/payment for purchased electricity,
- SC02 – amount of purchased electricity,
- SP03 – clearing/payment for sold electricity,
- SC03 – amount of sold electricity,
- SP05 – fee for traded electricity,
- SC05 – amount of traded electricity (the sum of sold and purchased electricity),
- SP90 – fee for trading transactions relating to data manipulation,
- SC90 – amount of trading transactions relating to data manipulation,
- SP91 – fee for trading transactions relating to use of automated interfaces,
- SC91 – amount of trading transactions relating to use of automated interfaces,
- SP92 – clearing/payment for purchased electricity for negative prices (positive number),
- SC92 – quantity of purchased electricity for negative prices (positive number),
- SP93 – clearing/payment for sold electricity for negative prices (positive number),
- SC93 – quantity of sold electricity for negative prices (positive number).

```
<ISOTEDATA id="9d1bd4cd-5c92-4f51-adde-6253a08cfbb" message-code="953"
       date-time="2009-07-03T14:16:54Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2009-09-21" delivery-duration="60">
    <ProfileData profile-role="SP02">
      <Data period="1" value="1022.5" unit="EUR"/>
      <Data period="2" value="746.35" unit="EUR"/>
      <Data period="3" value="1300" unit="EUR"/>
      <Data period="4" value="900" unit="EUR"/>
      <Data period="5" value="272.79" unit="EUR"/>
      <Data period="6" value="1621.75" unit="EUR"/>
      <Data period="7" value="2386.3" unit="EUR"/>
      <Data period="8" value="2546.7" unit="EUR"/>
      <Data period="9" value="794.25" unit="EUR"/>
      <Data period="10" value="1019.67" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC02">
      <Data period="1" value="50" unit="MWH"/>
      <Data period="2" value="23" unit="MWH"/>
      <Data period="3" value="65" unit="MWH"/>
      <Data period="4" value="45" unit="MWH"/>
      <Data period="5" value="12.6" unit="MWH"/>
      <Data period="6" value="65" unit="MWH"/>
      <Data period="7" value="98" unit="MWH"/>
      <Data period="8" value="78" unit="MWH"/>
      <Data period="9" value="45" unit="MWH"/>
      <Data period="10" value="41" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP03">
      <Data period="1" value="0" unit="EUR"/>
      <Data period="2" value="2086.535" unit="EUR"/>
      <Data period="3" value="0" unit="EUR"/>
      <Data period="4" value="0" unit="EUR"/>
      <Data period="5" value="0" unit="EUR"/>
      <Data period="6" value="583.83" unit="EUR"/>
      <Data period="7" value="1921.215" unit="EUR"/>
      <Data period="8" value="0" unit="EUR"/>
      <Data period="9" value="0" unit="EUR"/>
      <Data period="10" value="748.587" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC03">
      <Data period="1" value="0" unit="MWH"/>
      <Data period="2" value="64.3" unit="MWH"/>
      <Data period="3" value="0" unit="MWH"/>
      <Data period="4" value="0" unit="MWH"/>
      <Data period="5" value="0" unit="MWH"/>
      <Data period="6" value="23.4" unit="MWH"/>
      <Data period="7" value="78.9" unit="MWH"/>
      <Data period="8" value="0" unit="MWH"/>
      <Data period="9" value="0" unit="MWH"/>
      <Data period="10" value="30.1" unit="MWH"/>
    </ProfileData>
    <Party id="24XDSO---------Q" role="TO"/>
    <ResultStatus status="F"/>
  </Trade>
</ISOTEDATA>
```

**Example 71** Response containing request evaluations per periods

### 4.6.4 Notification of evaluation for a day (E-05_02)

Notification of evaluations for day is carried out by sending a request in *CDSREQ.961* structure (message-code=961) and response with data in *RESPONSE.962* and *ISOTEDATA.963* structure.

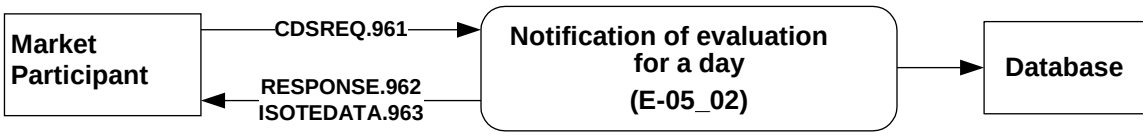


**Figure 35** Notification schema of evaluations for day for market participant

*CDSREQ.961*

Request is formulated for specific trading day (trade-day) and is filled out according to the specification of CDSREQ structure.

```
<CDSREQ id="45t" message-code="961" date-time="2014-09-19T01:18:33"
        dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2009-09-21"/>
</CDSREQ>
```

**Example 72** Request for retrieval of evaluations cumulatively for trading day

*RESPONSE.962*

In case of successful or failed request processing, response is returned according to specification of RESPONSE structure, where message-code=962 can be found in the header.

```
<RESPONSE id="8a848bad-46c2-4e3d-ab78-2c7c4545a21" message-code="962"
        date-time="2009-07-03T14:20:40Z" dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 73** Response on success of evaluations retrieval cumulatively for trading day

*ISOTEDATA.963*

In case of successful request processing, the structure of trading results is returned as it was registered in the system, where message-code=963 can be found in the header.

The following result types are used in *ProfileData /@profile-role* attribute:

- SP02 – clearing/payment for purchased electricity,
- SC02 – amount of purchased electricity,
- SP03 – clearing/payment for sold electricity,
- SC03 – amount of sold electricity,
- SP05 – fee for traded electricity,
- SC05 – amount of traded electricity (the sum of sold and purchased electricity),
- ST16 – monthly fee for access on day-ahead market; the last day of month is specified,
- SP90 – fee for trading transactions relating to data manipulation,
- SC90 – amount of trading transactions relating to data manipulation,
- SP91 – fee for trading transactions relating to use of automated interfaces,
- SC91 – amount of trading transactions relating to use of automated interfaces,
- SP92 – clearing/payment for purchased electricity for negative prices (positive number),
- SC92 – quantity of purchased electricity for negative prices (positive number),
- SP93 – clearing/payment for sold electricity for negative prices (positive number),
- SC93 – quantity of sold electricity for negative prices (positive number).

```
<ISOTEDATA id="526539ee-9bb7-465b-8e5c-0b660674f0f" message-code="963"
       date-time="2009-07-03T14:20:40Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2009-09-21">
    <ProfileData profile-role="SP02">
      <Data period="0" value="27875.987" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC02">
      <Data period="0" value="1146.7" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP03">
      <Data period="0" value="17278.838" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC03">
      <Data period="0" value="639.3" unit="MWH"/>
    </ProfileData>
    <Party id="24XDSO---------Q" role="TO"/>
    <ResultStatus status="F"/>
  </Trade>
</ISOTEDATA>
```

**Example 74** Response containing requested results cumulatively for trading day

## 4.7 Evaluation of IDM

Evaluations of intraday market are available for market participants via operations evaluation for a day or for a whole month retrieval.

### 4.7.1 Processing level

Results of intraday market are available anytime during a day. Summary results for a day or a month are completed when all periods are closed. Information about traded amount, invoices and debts of market organizer to market participant (debts with minus sign) are in summary result.

### 4.7.2 Notice of an evaluation for a day (E-07_01)

Notice of an evaluation for a day is carried out by request CDSREQ-VDT.961 structure (message-code=961) and by a response RESPONSE-VDT.962 structure and ISOTEDATA-VDT.963 structure with data.

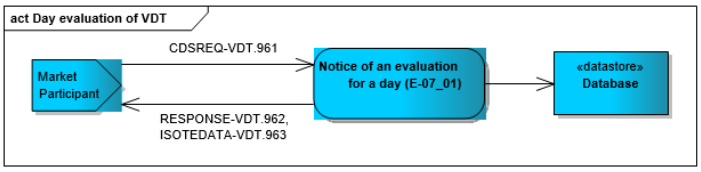


**Figure 36** Notice of an evaluation for a day scheme

*CDSREQ-VDT.961*

The structure contains message-code=961 attribute in the header and is filled out in accordance with CDSREQ-VDT.961 structure. Request is formulated for a specific trade day.

```
<CDSREQ id="45t" message-code="961" date-time="2016-09-19T01:18:33"
       dtd-version="1" dtd-release="1"
xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2016/04/01">
<SenderIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
<ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
<Trade trade-day="2016-09-21" />
</CDSREQ>
```

**Example 75** Request for a summary evaluation of a day

*RESPONSE-VDT.962*

In accordance with specification of RESPONSE-VDT structure response is given in case of successful or unsuccessful order processing. In this case message-code=802 can be found in the header.

```
<RESPONSE id="8a848bad-46c2-4e3d-ab78-2c7c4545a21" message-code="962"
       date-time="2016-07-03T14:20:40Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2016/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 76** Response about successful retrieval of day evaluation

*ISOTEDATA-VDT.963*

In case of successful order processing, the trading results structure is returned with a message-code=963 in the header.

In attribute ProfileData/@profile-role following results can be found:

- SP08 – reckoning/payment for purchased electricity for a positive price,
- SC08 – amount of purchased electricity for a positive price,
- SP58 – reckoning/payment for purchased electricity for a negative price,
- SC58 – amount of purchased electricity for a negative price,
- SP09 – reckoning/payment for a sold electricity for a positive price,
- SC09 – amount of sold electricity for a positive price,
- SP59 – reckoning/payment for a sold electricity for a negative price,
- SC59 – amount of a sold electricity for a negative price.

```
<ISOTEDATA id="526539ee-9bb7-465b-8e5c-0b660674f0f" message-code="963"
       date-time="2016-07-03T14:20:40Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2016/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2009-09-21">
    <ProfileData profile-role="SP08">
      <Data value="27875.987" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC08">
      <Data value="1146.7" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP09">
      <Data value="17278.838" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC09">
      <Data value="639.3" unit="MWH"/>
    </ProfileData>
    <Party id="24X--YOUR-EIC--B " role="TO"/>
  </Trade>
```

**Example 77** Response with day evaluation results

### 4.7.3 Notice of an evaluation for a month (E-07_02)

Notice of an evaluation for a month is carried out by request CDSREQ-VDT.571 structure (message-code=571) and by a response RESPONSE-VDT.572 structure and ISOTEDATA-VDT.573 structure with data.

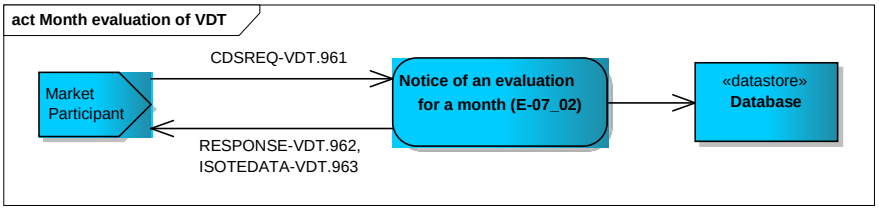


**Figure 37** Notice of an evaluation for a month scheme

### *CDSREQ-VDT.571*

The structure contains message-code=571 attribute in the header and is filled out in accordance with CDSREQ-VDT structure. Request is formulated for a concrete trade month.

```
<CDSREQ id="45t" message-code="571" date-time="2016-09-19T01:18:33"
       dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2016/04/01">
  <SenderIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-month="2016-09" />
</CDSREQ>
```

**Example 78** Request for a summary evaluation of a month

### *RESPONSE-VDT.572*

According to specification of RESPONSE-VDT structure response is given in case of successful or unsuccessful order processing. In this case message-code=572 can be found in the header.

```
<RESPONSE id="8a848bad-46c2-4e3d-ab78-2c7c4545a21" message-code="572"
       date-time="2016-07-03T14:20:40Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2016/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 79** Response about successful retrieval of month evaluation

### *ISOTEDATA-VDT.573*

In case of successful order processing, the trading results structure is returned with a message-code=573 in the header.

In attribute ProfileData/@profile-role following results can be found:

- SP08 – reckoning/payment for purchased electricity for a positive price,
- SC08 – amount of purchased electricity for a positive price,
- SP58 – reckoning/payment for purchased electricity for a negative price,
- SC58 – amount of purchased electricity for a negative price,
- SP09 – reckoning/payment for a sold electricity for a positive price,
- SC09 – amount of sold electricity for a positive price,
- SP59 – reckoning/payment for a sold electricity for a negative price,
- SC59 – amount of a sold electricity for a negative price,
- SP10 – charge for traded electricity,
- SC10 – amount of traded electricity (price of sold and purchased).

```
<ISOTEDATA id="526539ee-9bb7-465b-8e5c-0b660674f0f" message-code="573"
       date-time="2016-07-03T14:20:40Z" dtd-version="1" dtd-release="1" answer-
required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2016/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-month="2016-09">
    <ProfileData profile-role="SP08">
      <Data value="27875.987" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC08">
      <Data value="1146.7" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP09">
      <Data value="17278.838" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC09">
      <Data value="639.3" unit="MWH"/>
    <ProfileData profile-role="SP10">
      <Data value="727.8" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SP10">
      <Data value="88839.3" unit="MWH"/>
    </ProfileData>
    <Party id="24X--YOUR-EIC--B " role="TO"/>
  </Trade>
```

**Example 80** Response with month evaluation results

### 4.7.4 Notice of evaluation for time periods (E-07_03)

Notice of an evaluation for time periods is carried out by request CDSREQ-VDT.951 structure (message-code=951) and by a response RESPONSE-VDT.952 structure and ISOTEDATA-VDT.953 structure with data.

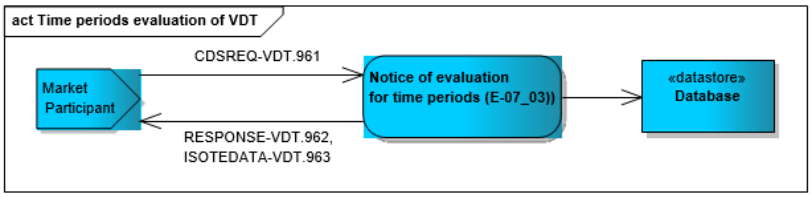


**Figure 38** Notice of an evaluation for time periods scheme

*CDSREQ-VDT.951*

The structure contains message-code=951 attribute in the header and is filled out in accordance with CDSREQ-VDT structure. Request is formulated for a concrete trade day or a concrete day can be specified with period-from, period-to.

```
<CDSREQ date-time="2017-04-11T07:00:00" dtd-release="1" dtd-version="1"  id="45t" message-
code="951" xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
   <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
   <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
   <Trade trade-day="2017-02-12"/>
</CDSREQ>
```

**Example 81** Request for an evaluation of time periods for whole day

```
<CDSREQ date-time="2017-04-11T07:00:00" dtd-release="1" dtd-version="1"  id="45t" message-
code="951" xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
   <SenderIdentification coding-scheme="15" id="24X--YOUR-EIC--B"/>
   <ReceiverIdentification coding-scheme="15" id="24X-OT-SK------V"/>
   <Trade trade-day="2017-02-12" period-from="8" period-to="20" delivery-duration="60"/>
</CDSREQ>
```

**Example 82** Request for an evaluation of time periods for specific period

### *RESPONSE-VDT.952*

In accordance with RESPONSE-VDT structure response is given in case of successful or unsuccessful order processing. In this case message-code=952 can be found in the header.

```
<RESPONSE id="8a848bad-46c2-4e3d-ab78-2c7c4545a21" message-code="952"
       date-time="2016-07-03T14:20:40Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2016/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 83** Response about successful retrieval of time periods evaluation

### *ISOTEDATA-VDT.953*

In case of successful order processing, the trading results structure is returned with a message-code=953 in the header. In attributes Trade/ProfileData/Data/@period-from and Trade/ProfileData/Data/@period-to can be found specified interval of delivery. These intervals represent order of time periods in a specific day.

In attribute ProfileData/@profile-role following results can be found:

- SP08 – reckoning/payment for purchased electricity for a positive price,
- SC08 – amount of purchased electricity for a positive price,
- SP58 – reckoning/payment for purchased electricity for a negative price,
- SC58 – amount of purchased electricity for a negative price,
- SP09 – reckoning/payment for a sold electricity for a positive price,
- SC09 – amount of sold electricity for a positive price,
- SP59 – reckoning/payment for a sold electricity for a negative price,
- SC59 – amount of a sold electricity for a negative price.

```
<ISOTEDATA id="917de340469d45ab9cc14ec18797c31f" message-code="963" date-time="2017-04-
11T12:17:50Z" dtd-version="1" dtd-release="1" answer-required="false"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
   <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
   <ReceiverIdentification id="24X--YOUR-EIC--B " coding-scheme="15"/>
   <Reference id="45t"/>
   <Trade trade-day="2017-02-12" delivery-duration="60">
      <ProfileData profile-role="SP08">
         <Data period-from="0" period-to="1" value="10.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC08">
         <Data period-from="0" period-to="1" value="1.0" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP58">
         <Data period-from="0" period-to="1" value="0.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC58">
         <Data period-from="0" period-to="1" value="0.0" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP09">
         <Data period-from="0" period-to="1" value="50.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC09">
         <Data period-from="0" period-to="1" value="2.0" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP59">
         <Data period-from="0" period-to="1" value="0.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC59">
         <Data period-from="0" period-to="1" value="0.0" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP08">
         <Data period-from="1" period-to="2" value="10.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC08">
         <Data period-from="1" period-to="2" value="0.5" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP58">
         <Data period-from="1" period-to="2" value="0.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC58">
         <Data period-from="1" period-to="2" value="0.0" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP09">
         <Data period-from="1" period-to="2" value="100.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC09">
         <Data period-from="1" period-to="2" value="5.0" unit="MWH"/>
      </ProfileData>
      <ProfileData profile-role="SP59">
         <Data period-from="1" period-to="2" value="0.0" unit="EUR"/>
      </ProfileData>
      <ProfileData profile-role="SC59">
         <Data period-from="1" period-to="2" value="0.0" unit="MWH"/>
      </ProfileData>
      <Party id="24X--YOUR-EIC--B " role="TO"/>
   </Trade>
</ISOTEDATA>
```

**Example 84** Response with time periods evaluation results (2 periods)

## 4.8 AMQP notifications

Notification via AMQP protocol automatically informs market participants about changes in their own orders, or changes in the order book.

### 4.8.1 Processing level

The market participant connected to AMQP interface has the opportunity to get automatic notifications. Queues are automatically created, and allows users to get notification about following event:

- Successfully created order – dataflow (E-10_01),
- Change of the order (status change) – dataflow E-10_01,
- Change in the order book (change of available amount and last price/quantity for periods where the change occured) – dataflow E-10_02.

### 4.8.2 Status change/Creation of own order (E-10_01)

Notification via AMQP protocol is send within dataflow and inform market participant about status changes of own order or about creation of new own order. Notification is sent automatically in accordance with ISOTEDATA-VDT.820 structure (message-code=820).

Notification is a reaction on change, which was called by:

- Market participant (creation or modification of an order),
- Another market participant (whole order traded or part of the order traded),
- System (expiration of the order, or more precisely time period in which was the order inputted and changed afterwards).

Following pictures show transmission of messages which are send when the order is created by market participant connected to AMQP interface via web services. Notification about created order is send to market participant connected to AMQP interface even in case when another subject create order for the market participant via web page XMtrade®/ISOT.

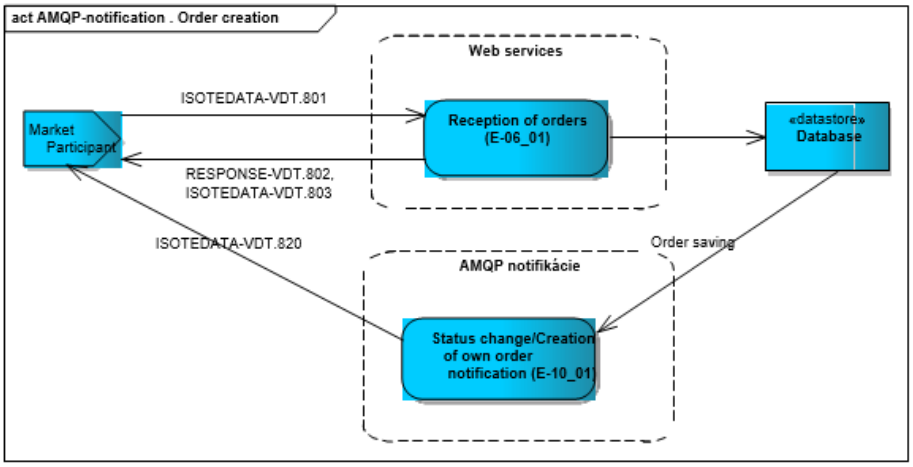


**Figure 39** Sending messages scheme – order creation (AMQP notification)

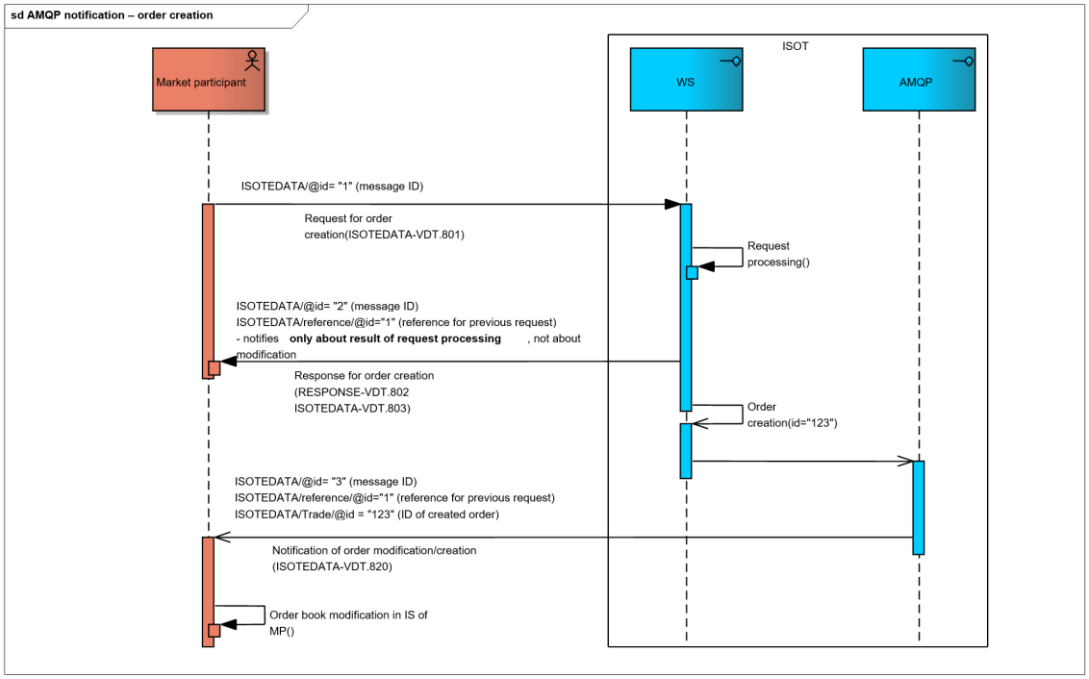


**Figure 40** Order of sended messages scheme – order creation (AMQP notification)

Following pictures show transmission of messages which are sent when the order is modified by market participant connected to AMQP interface via web services. Notification about modified order is send to market participant connected to AMQP interface even in case when another subject modified order for the market participant via web page XMtrade®/ISOT.

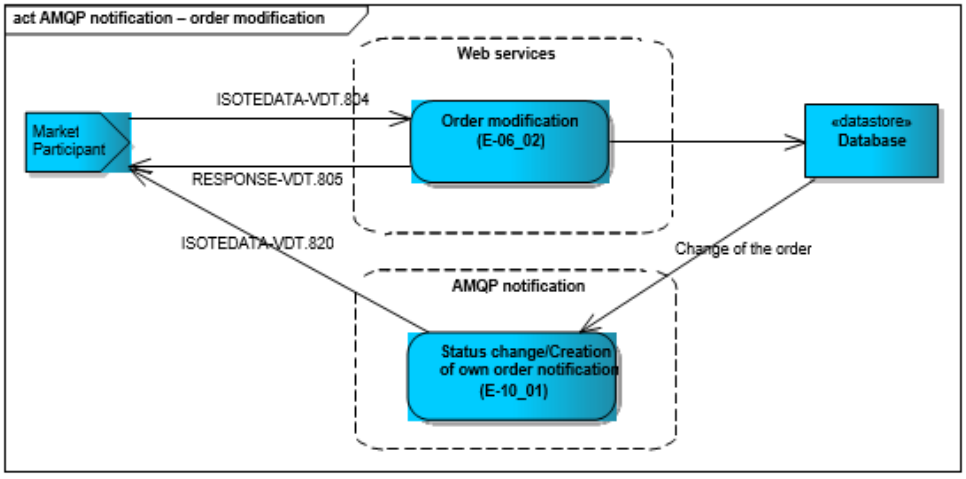


**Figure 41** Sending messages scheme – order modification (AMQP notification)

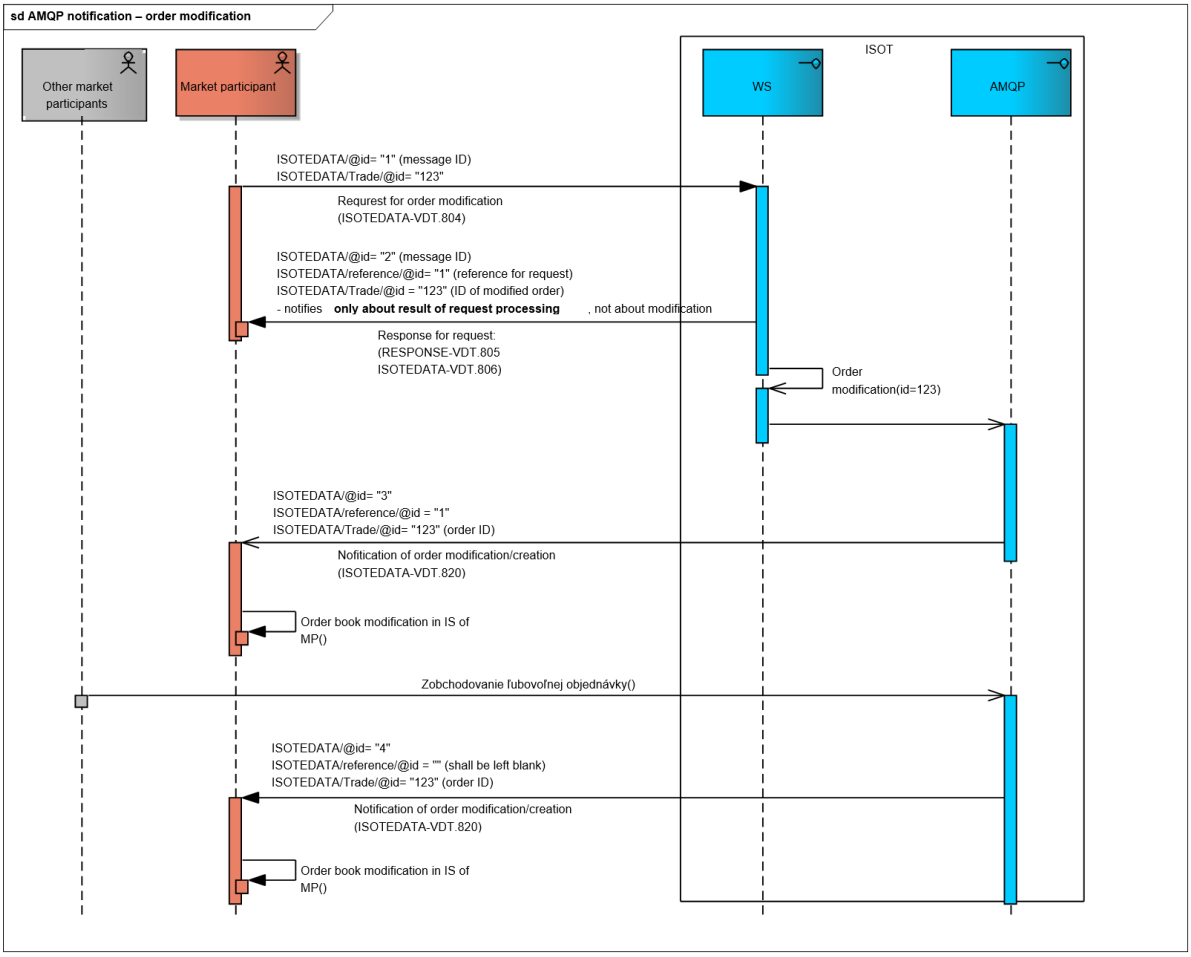


**Figure 42** Order of sent messages scheme – order modification (AMQP notification)

## *ISOTEDATA-VDT.820*

In the case of events shown above, the structure of the order data is sent through the AMQP protocol, with a message-code=820 in the header.

In attribute Reference/@id is reference to a message, which called out order creation. ID of created order is returned in attribute Trade/@id, which can be saved or paired in market participant order system via web service.

```
<ISOTEDATA id="ac5e799q-2qtr-75e7-9bef-8aabc02b7f4" message-code="820" date-time="2016-02-
15T16:30:10Z" dtd-version="1" dtd-release="1" answer-required="false"
xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24X--YOUR-EIC--B" coding-scheme="15" />
  <!--reference to message 801 - request for order submission-->
  <Reference id="1"/>
  <Trade id="1016" trade-day="2016-02-16" order-expiration="2016-02-15T19:30:10" trade-
type="N" block-order="N" indication="N" trade-stage="P" trader-id="123456" delivery-
duration="60" market-area="SK" sett-curr="EUR" market="VDT">
    <TimeData datetime="2016-02-15T16:30:10Z" datetime-type="DTC"/>
    <ProfileData profile-role="BC01">
      <Data period-from="0" period-to="1" value="10.1" unit="MW" />
    </ProfileData>
    <ProfileData profile-role="BP01">
      <Data period-from="0" period-to="1" value="40.00" unit="EUR" />
    </ProfileData>
    <Party id="24X--YOUR-EIC--B" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 85** Example of change in market participant order notification

### 4.8.3 Change of the order book data (E-10_02)

Notification via AMQP protocol is sent within dataflow and inform market participant about order book status changes. Notification is sent automatically in accordance with ISOTEDATA-VDT.830 structure (message-code=830). Same message is sent to all users connected to AMQP interface, but corresponding notification is not at the same level as the header addressed to a specific market participant (element ReceiverIdentification is not used).

Notification is a reaction to change in the order book, which can be caused by following cases that have consequences to amounts in one or more trading time periods:

- Market participant (adding, trading, modification of an order),
- Change caused by other market participant (adding, trading, modification of an order),
- System (expiration of and order, more precisely time period which can cause annulment of available amount in concrete time period).

Following pictures show messages transmission and sequence of those messages, which are sent to gather actual status of the order book. These messages are sent in case market participant is connected to AMQP interface and do not gather overall data from the order book via web services.

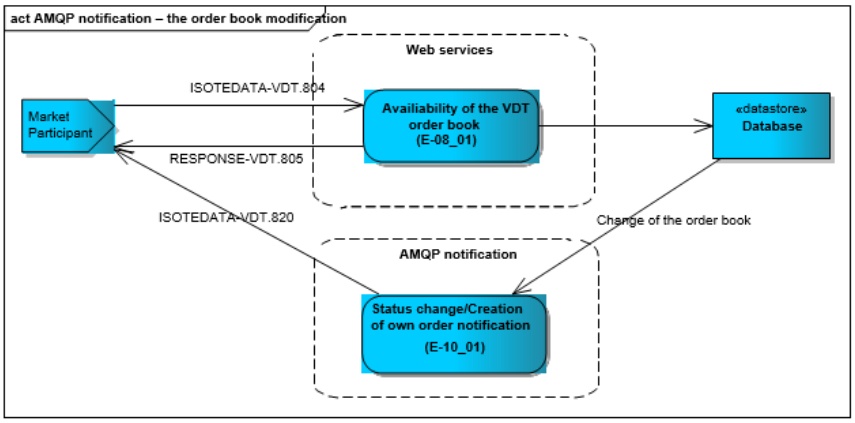


**Figure 43** Sent messages scheme – order creation (AMQP notification)

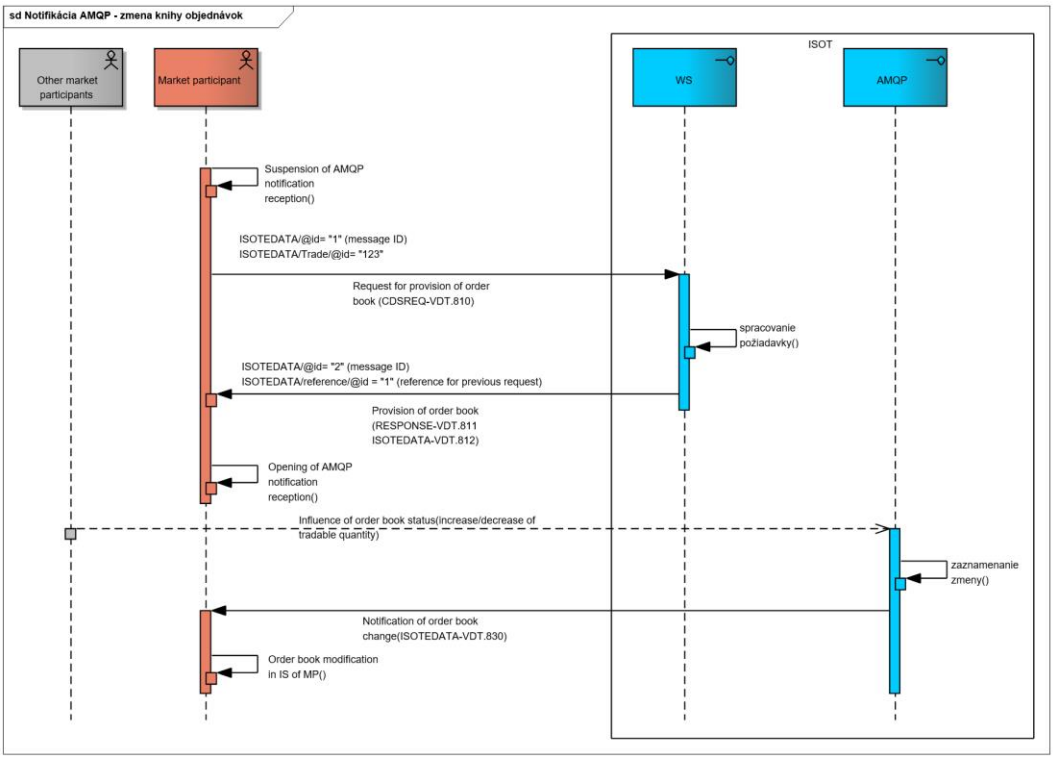


**Figure 44** Sequentiality of sent messages scheme – modification of order book (AMQP notification)

Method of gathering and keeping up-to-date status of the order book is following:

1. Connect to AMQP interface
2. Intermit queue processing for income of automatized notifications.
3. Download the data of the order book via web services.
    a. During downloading data of the order book via web services it is necessary for market participant to not receive notification about changes in the order book.
4. Start receiving notification about changes in the order book
    a. This step ensures actualization of the order book – changes which were made during the downloading the order book via web services will be included
    b. The market participant will be informed about following changes automatically
5. In case market participant needs to download the order book again via web services, it is necessary to start with step number 2.
6. More information about connection failure can be found in the chapter Communication security 3.2.3.

Information necessary for correct notification processing and keeping up-to-date order book:

- Every notification regarded to the order book change, rewrite original amount of the order book for the specific type of order, direction(purchase/sale) and limited price.
- Every change represents one notification in a single XML document.
- In case of block order change, (base-load, peak-load and off-peak) notifications are always sent for side of orders and at the same time for side of sale. In case of change in user-defined block order, notifications about change are sent for all own orders in specific trading time periods.

### *ISOTEDATA-VDT.830*

In case of events shown above, structure of the order book is sent through the AMQP protocol, with a message-code=830 in the header.

This type of message is not intended for a specific user, or market participant, but is intended for all users connected to AMQP interface. Element ReceiverIdentification is not filled out in the message. Amount for a specific price in specific time periods is shown in the message.

Following examples show sending of notification principle:

*Example 1*

1. Period 12-13 has the following amounts on trading day 7/13/2016:

| Trading day: 13.7.2016 | Purchase | | Sale | |
|---|---|---|---|---|
| | **Amount [MWh]** | **Price [EUR/MW]** | **Amount [MWh]** | **Price [EUR/MW]** |
| Period:12-13 | 10,0 | 31,00 | | |

2. Simple order for purchase (without indications) is booked for this period with amount of 11MW for 31 EUR/MW.
3. Period 12-13 will contain following amounts after matching:

| Trading day: 13.7.2016 | Purchase | | Sale | |
|---|---|---|---|---|
| | **Amount [MWh]** | **Price [EUR/MW]** | **Amount [MWh]** | **Price [EUR/MW]** |
| Period:12-13 | 0,0 | 31,00 | 1,0 | 31,00 |

4. Notification no.1, showed below, represents change in amount at the side of purchase at 12-13 period to 0 MW for price 31 EUR/MW. Notification no.2 represents raise of amount at the side of sale to 1 MW for price 31 EURPMW. Both notifications include information of last price/quantity and overall traded quantity of the period 12-13 within sections TC01, LC01, LP01.

```
<!--notification no.1-->
<?xml version="1.0" encoding="utf-8"?>
<ISOTEDATA xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="56f4aab9fd36497497b38fd0ef3a0223"
message-code="830" date-time="2017-08-07T11:04:02Z" dtd-version="1" dtd-release="1" answer-
required="false" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <!--den D, posledná cena/množstvo-->
  <Trade trade-day="2016-07-13" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
      <TimeData datetime="2016-07-13T11:04:02.123Z" datetime-type="DTO"/>
      <ProfileData profile-role="TC01">
         <Data period-from="12" period-to="13" value="20.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LC01">
         <Data period-from="12" period-to="13" value="10.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LP01">
         <Data period-from="12" period-to="13" value="31.00" unit="EUR" price-
direction="N"/>
      </ProfileData>
  </Trade>
  <Trade trade-day="2017-07-13" trade-type="N" block-order="N" delivery-duration="60"
market="VDT" sett-curr="EUR" market-area="SK">
    <TimeData datetime="2017-07-13T11:04:02.808951Z" datetime-type="DTO" />
    <ProfileData profile-role="BP01">
      <Data period-from="12" period-to="13" value="31" unit="EUR" />
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="12" period-to="13" value="0" unit="MW" />
    </ProfileData>
  </Trade>
</ISOTEDATA>

<!--notification no.2-->
<?xml version="1.0" encoding="utf-8"?>
<ISOTEDATA xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="56f4aab9fd36497497b38fd0ef3a0223"
message-code="830" date-time="2017-08-07T11:04:02Z" dtd-version="1" dtd-release="1" answer-
required="false" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2016-07-13" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
      <TimeData datetime="2016-07-13T11:04:02.123Z" datetime-type="DTO"/>
      <ProfileData profile-role="TC01">
         <Data period-from="12" period-to="13" value="20.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LC01">
         <Data period-from="12" period-to="13" value="10.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LP01">
         <Data period-from="12" period-to="13" value="31.00" unit="EUR" price-
direction="N"/>
      </ProfileData>
   </Trade>
   <Trade trade-day="2017-07-13" trade-type="P" block-order="N" delivery-duration="60"
market="VDT" sett-curr="EUR" market-area="SK">
    <TimeData datetime="2017-07-13T11:04:02.808951Z" datetime-type="DTO" />
    <ProfileData profile-role="BP01">
      <Data period-from="12" period-to="13" value="31" unit="EUR" />
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="12" period-to="13" value="1" unit="MW" />
    </ProfileData>
  </Trade>
</ISOTEDATA>
```

**Example 86** Example notification of change in the order book (change at the of purchase and sale and last price/quantity in the period)

*Example 2*

1. Period 16-17 has the following amounts on trading day 7/13/2016:

| Trading day: | Purchase | Sale |
|---|---|---|

| 13.7.2016 | Amount [MWh] | Price [EUR/MW] | Amount [MWh] | Price [EUR/MW] |
|---|---|---|---|---|
| Period:16-17 | | | 5,0 | 45,00 |
| | | | 3,0 | 46,00 |
| | | | 2,0 | 46,15 |

2. Simple order for purchase (without indications) is booked for this period with amount of 10MW for 47 EUR/MW.
3. Period 16-17 will contain following amounts after matching:

| Trading day: | Purchase | | Sale | |
|---|---|---|---|---|
| 13.7.2016 | Amount [MWh] | Price [EUR/MW] | Amount [MWh] | Price [EUR/MW] |
| Period:16-17 | | | 0,0 | 45,00 |
| | | | 0,0 | 46,00 |
| | | | 0,0 | 46,15 |

4. Notification no.1 represents change in amount at the side of purchase at 16-17 period to 0 MW for price 45 EUR/MW. Notification no.2 represents change to 0 MW for price 46,15 EURPMW and notification no.3 represents change to 0 MW for price 46 EUR/MW. Both notifications include information of last price/quantity and overall traded quantity of the period 16-17 within sections TC01, LC01, LP01.

```
<!--notifikacia c.1-->
<?xml version="1.0" encoding="utf-8"?>
<ISOTEDATA xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="8ddc0826d77847aea837c057865adffd"
message-code="830" date-time="2017-08-07T11:11:26Z" dtd-version="1" dtd-release="1" answer-
required="false" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2016-07-13" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
      <TimeData datetime="2016-07-13T11:04:02.123Z" datetime-type="DTO"/>
      <ProfileData profile-role="TC01">
         <Data period-from="16" period-to="17" value="20.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LC01">
         <Data period-from="16" period-to="17" value="2.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LP01">
         <Data period-from="16" period-to="17" value="46.15" unit="EUR" price-direction="I"/>
      </ProfileData>
   </Trade>
   <Trade trade-day="2017-07-13" trade-type="P" block-order="N" delivery-duration="60"
market="VDT" sett-curr="EUR" market-area="SK">
    <TimeData datetime="2017-07-13T11:11:26.4356724Z" datetime-type="DTO" />
    <ProfileData profile-role="BP01">
      <Data period-from="16" period-to="17" value="45" unit="EUR" />
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="16" period-to="17" value="0" unit="MW" />
    </ProfileData>
  </Trade>
</ISOTEDATA>

<!--notifikacia c.2-->
<?xml version="1.0" encoding="utf-8"?>
<ISOTEDATA xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="8ddc0826d77847aea837c057865adffd"
message-code="830" date-time="2017-08-07T11:11:26Z" dtd-version="1" dtd-release="1" answer-
required="false" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2016-07-13" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
      <TimeData datetime="2016-07-13T11:04:02.123Z" datetime-type="DTO"/>
      <ProfileData profile-role="TC01">
         <Data period-from="16" period-to="17" value="20.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LC01">
         <Data period-from="16" period-to="17" value="2.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LP01">
         <Data period-from="16" period-to="17" value="46.15" unit="EUR" price-direction="I"/>
      </ProfileData>
   </Trade>
   <Trade trade-day="2017-07-13" trade-type="P" block-order="N" delivery-duration="60"
market="VDT" sett-curr="EUR" market-area="SK">
    <TimeData datetime="2017-07-13T11:11:26.4356724Z" datetime-type="DTO" />
    <ProfileData profile-role="BP01">
      <Data period-from="16" period-to="17" value="46.15" unit="EUR" />
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="16" period-to="17" value="0" unit="MW" />
    </ProfileData>
  </Trade>
</ISOTEDATA>
```

```
<!--notifikacia c.3-->
<?xml version="1.0" encoding="utf-8"?>
<ISOTEDATA xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="8ddc0826d77847aea837c057865adffd"
message-code="830" date-time="2017-08-07T11:11:26Z" dtd-version="1" dtd-release="1" answer-
required="false" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade trade-day="2016-07-13" delivery-duration="60" market-area="SK" sett-curr="EUR"
market="VDT">
      <TimeData datetime="2016-07-13T11:04:02.123Z" datetime-type="DTO"/>
      <ProfileData profile-role="TC01">
         <Data period-from="16" period-to="17" value="20.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LC01">
         <Data period-from="16" period-to="17" value="2.0" unit="MW"/>
      </ProfileData>
      <ProfileData profile-role="LP01">
         <Data period-from="16" period-to="17" value="46.15" unit="EUR" price-direction="I"/>
      </ProfileData>
   </Trade>
   <Trade trade-day="2017-07-13" trade-type="P" block-order="N" delivery-duration="60"
market="VDT" sett-curr="EUR" market-area="SK">
    <TimeData datetime="2017-07-13T11:11:26.4356724Z" datetime-type="DTO" />
    <ProfileData profile-role="BP01">
      <Data period-from="16" period-to="17" value="46" unit="EUR" />
    </ProfileData>
    <ProfileData profile-role="BC01">
      <Data period-from="16" period-to="17" value="0" unit="MW" />
    </ProfileData>
  </Trade>
</ISOTEDATA>
```

**Example 87** Example notification of change in the order book (the side of sale, different prices and last price/quantity in the period)

### 4.8.4 Change in cross-border capacities data H2H (E-10_03)

Notification informing the market participant about a change in available cross-border capacities (H2H) is sent using AMQP protocol in case of operation of coordinated cross-border intraday market. Notification is generated by the central matching mechanism within the SIDC, while ISOT system is automatically processing this information and forwarding it to market participants connected via AMQP interfaces. Notification is automatically sent in the structure ISOTEDATA-VDT.843 (message-code=843). Form of XML file shares the logic with the message ISOTEDATA-VDT.842 sent as a part of response within the data flow E-08_02. Notification about a change sent within AMQP interfaces includes only those periods which are subject to the change.

Notification is a response to change within the shared order book, which may be caused by change withing the order book status within coupled cross-border intraday market with regard to cross-border capacities or by a change within the cross-border capacities values defined by transmission system operators.

*ISOTEDATA-VDT.843*

In case of occurence of previously mentioned events, message containing change in available cross-border capacities data with message-code=843 is sent via AMQP protocol.

```
<ISOTEDATA xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" id="d48ee3c16f7e4c24b3a9a3c9ad741ec2"
message-code="843" date-time="2022-09-27T08:41:10Z" dtd-version="1" dtd-release="1" answer-
required="false" xmlns="http://sfera.sk/xmtrade/isot/types/IDM/2016/04">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <Trade market="VDT" delivery-duration="60" market-area="SK" area-from="10YSK-SEPS-----K" area-
to="10YCZ-CEPS-----N">
    <TimeData datetime="2022-09-27T08:41:10.1720189Z" datetime-type="DTO" />
    <ProfileData profile-role="AC01">
      <Data period-from="14" period-to="15" value="1923.4" unit="MW" />
    </ProfileData>
    <ProfileData profile-role="AC02">
      <Data period-from="14" period-to="15" value="1425.8" unit="MW" />
    </ProfileData>
  </Trade>
</ISOTEDATA>
```

**Example 88** Example of notification about change in available cross-border capacities data (H2H)

## 4.9 IDA results and evaluations

Results and evaluations of intraday auctions are available for market participants via operations for IDA results and evaluation retrieval for every period or a whole day.

### 4.9.1 Processing level

Results of intraday auction are available immediately after order matching and contain accepted amount and final marginal price (system or area price). Market participant is notified about results availability via ISOT system.

Evaluations of intraday auction are available immediately after clearing of specific auction in the form of summary daily evaluation as well as detailed evaluation per periods. Evaluations contain market organizer obligations and receivables towards a market participant. (receivables are stated with a negative sign). Market participant is notified about evaluations availability via ISOT system. Results of day-ahead market are made available in ISOT.

Evaluations within the intraday auctions are recalculated to MWh quantity unit according to the usage of 15-minute trading periods.

### 4.9.2 Notification of results for market participants (E-11_01)

Notification of IDA results for market participants is carried out by request in *CDSREQ.941* structure (message-code=941) and response with data in *RESPONSE.942* and *ISOTEDATA.943* structure.

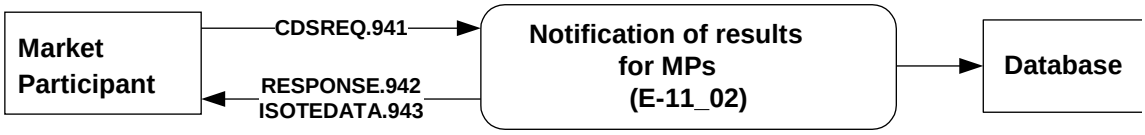


**Figure 45** Notification of IDA results for MP scheme

*CDSREQ.941*

Request is formulated for specific auction (trade-day and auction-id) and is filled out in accordance with the specification of CDSREQ structure.

```
<CDSREQ id="45t" message-code="941" date-time="2023-12-01T01:18:33"
        dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2023-12-01" auction-id="IDA1"/>
</CDSREQ>
```

**Example 89** Request for downloading IDA results for specific auction

*RESPONSE.942*

In case of successful or failed request processing, response is returned in accordance with specification of RESPONSE structure, where message-code=942 can be found in the header.

```
<RESPONSE id="bd12362f-361b-4085-ade0-9ed678efff1" message-code="942"
        date-time="2023-12-01T14:11:43Z" dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 90** Response on success of downloading IDA results for specific auction

*ISOTEDATA.943*

In case of successful request processing, the structure of IDA results is returned as it was registered in the system, where message-code=943 can be found in the header.

Within the ProfileData/@profile-role attribute are used following types of results:

- SC19 – quantity of purchased electricity for positive prices,
- SC20 – quantity of sold electricity for positive prices,
- SC92 – quantity of purchased electricity for negative prices,
- SC93 – quantity of sold electricity for negative prices,
- SP20 – marginal price.

```
<ISOTEDATA id="ec1b50c0-afe1-4f5e-b6a1-d94c365099e" message-code="943"
       date-time="2023-12-01T14:11:43Z" dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2023-12-01" auction-id="IDA1" delivery-duration="15" >
    <ProfileData profile-role="SC19">
      <Data period="1" value="50" unit="MWH"/>
      <Data period="2" value="23" unit="MWH"/>
      <Data period="3" value="65" unit="MWH"/>
      <Data period="4" value="45" unit="MWH"/>
      <Data period="5" value="12.6" unit="MWH"/>
      <Data period="6" value="65" unit="MWH"/>
      <Data period="7" value="98" unit="MWH"/>
      <Data period="8" value="78" unit="MWH"/>
      <Data period="9" value="45" unit="MWH"/>
      <Data period="10" value="41" unit="MWH"/>
      <Data period="11" value="42" unit="MWH"/>
      <Data period="12" value="12" unit="MWH"/>
      <Data period="13" value="65" unit="MWH"/>
      <Data period="14" value="31.1" unit="MWH"/>
      <Data period="15" value="32.5" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SC20">
      <Data period="1" value="0" unit="MWH"/>
      <Data period="2" value="64.3" unit="MWH"/>
      <Data period="3" value="0" unit="MWH"/>
      <Data period="4" value="0" unit="MWH"/>
      <Data period="5" value="0" unit="MWH"/>
      <Data period="6" value="23.4" unit="MWH"/>
      <Data period="7" value="78.9" unit="MWH"/>
      <Data period="8" value="0" unit="MWH"/>
      <Data period="9" value="0" unit="MWH"/>
      <Data period="10" value="30.1" unit="MWH"/>
      <Data period="11" value="0" unit="MWH"/>
      <Data period="12" value="0" unit="MWH"/>
      <Data period="13" value="0" unit="MWH"/>
      <Data period="14" value="50" unit="MWH"/>
      <Data period="15" value="40" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP20">
      <Data period="1" value="20.45" unit="EUR"/>
      <Data period="2" value="32.45" unit="EUR"/>
      <Data period="3" value="20" unit="EUR"/>
      <Data period="4" value="20" unit="EUR"/>
      <Data period="5" value="21.65" unit="EUR"/>
      <Data period="6" value="24.95" unit="EUR"/>
      <Data period="7" value="24.35" unit="EUR"/>
      <Data period="8" value="32.65" unit="EUR"/>
      <Data period="9" value="17.65" unit="EUR"/>
      <Data period="10" value="24.87" unit="EUR"/>
      <Data period="11" value="23.98" unit="EUR"/>
      <Data period="12" value="15.45" unit="EUR"/>
      <Data period="13" value="19.87" unit="EUR"/>
      <Data period="14" value="33.54" unit="EUR"/>
      <Data period="15" value="17.65" unit="EUR"/>
    </ProfileData>
    <Party id="24XDSO---------Q" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 91** Response containing requested IDA results

### 4.9.3 Notification of evaluations per periods (E-11_01)

Notification of evaluations per periods is carried out by sending a request in *CDSREQ.951* structure (message-code=951) and response with data in *RESPONSE.952* and *ISOTEDATA.953* structures.

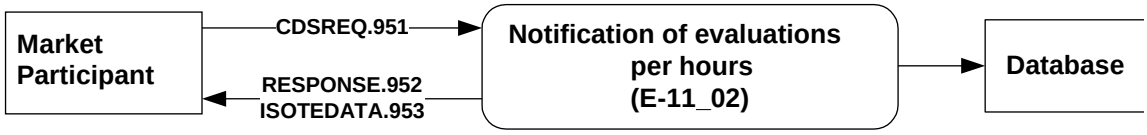


**Figure 46** Scheme for IDA evaluation per periods for MPs

*CDSREQ.951*

To be filled out in accordance with specification of CDSREQ structure. Request is formulated for specific trading day (trade-day) and auction (auction-id).

```
<CDSREQ id="45t" message-code="951" date-time="2023-23-01T01:18:33"
       dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2023-12-01" auction-id="IDA1" />
</CDSREQ>
```

**Example 92** Request for IDA evaluation per periods for specific auction

*RESPONSE.952*

In case of successful or failed request processing, response is returned in accordance with specification of RESPONSE structure, where message-code=952 can be found in the header.

```
<RESPONSE id="7cdd21c0-e21f-4e70-a617-2d55db510e8" message-code="952"
date-time="2023-12-01T14:16:54Z" dtd-version="1" dtd-release="1"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 93** Response on success of requesting IDA evaluation per periods

*ISOTEDATA.953*

In the event of successful request processing, the structure of trading results is returned as it was registered in the system, where message-code=953 can be found in the header.

The following result types are used in *ProfileData/@profile-role* attribute:

- SP02 – clearing/payment for purchased electricity,
- SC02 – amount of purchased electricity,
- SP03 – clearing/payment for sold electricity,
- SC03 – amount of sold electricity,
- SP05 – fee for traded electricity,
- SC05 – amount of traded electricity (the sum of sold and purchased electricity),
- SP90 – fee for trading transactions relating to data manipulation,
- SC90 – amount of trading transactions relating to data manipulation,
- SP91 – fee for trading transactions relating to use of automated interfaces,
- SC91 – amount of trading transactions relating to use of automated interfaces,
- SP92 – clearing/payment for purchased electricity for negative prices (positive number),
- SC92 – quantity of purchased electricity for negative prices (positive number),
- SP93 – clearing/payment for sold electricity for negative prices (positive number),
- SC93 – quantity of sold electricity for negative prices (positive number).

```
<ISOTEDATA id="9d1bd4cd-5c92-4f51-adde-6253a08cfbb" message-code="953"
       date-time="2023-12-01T14:16:54Z" dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2023-12-01" auction-id="IDA1" delivery-duration="15" >
    <ProfileData profile-role="SP02">
      <Data period="1" value="1022.5" unit="EUR"/>
      <Data period="2" value="746.35" unit="EUR"/>
      <Data period="3" value="1300" unit="EUR"/>
      <Data period="4" value="900" unit="EUR"/>
      <Data period="5" value="272.79" unit="EUR"/>
      <Data period="6" value="1621.75" unit="EUR"/>
      <Data period="7" value="2386.3" unit="EUR"/>
      <Data period="8" value="2546.7" unit="EUR"/>
      <Data period="9" value="794.25" unit="EUR"/>
      <Data period="10" value="1019.67" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC02">
      <Data period="1" value="50" unit="MWH"/>
      <Data period="2" value="23" unit="MWH"/>
      <Data period="3" value="65" unit="MWH"/>
      <Data period="4" value="45" unit="MWH"/>
      <Data period="5" value="12.6" unit="MWH"/>
      <Data period="6" value="65" unit="MWH"/>
      <Data period="7" value="98" unit="MWH"/>
      <Data period="8" value="78" unit="MWH"/>
      <Data period="9" value="45" unit="MWH"/>
      <Data period="10" value="41" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP03">
      <Data period="1" value="0" unit="EUR"/>
      <Data period="2" value="2086.535" unit="EUR"/>
      <Data period="3" value="0" unit="EUR"/>
      <Data period="4" value="0" unit="EUR"/>
      <Data period="5" value="0" unit="EUR"/>
      <Data period="6" value="583.83" unit="EUR"/>
      <Data period="7" value="1921.215" unit="EUR"/>
      <Data period="8" value="0" unit="EUR"/>
      <Data period="9" value="0" unit="EUR"/>
      <Data period="10" value="748.587" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC03">
      <Data period="1" value="0" unit="MWH"/>
      <Data period="2" value="64.3" unit="MWH"/>
      <Data period="3" value="0" unit="MWH"/>
      <Data period="4" value="0" unit="MWH"/>
      <Data period="5" value="0" unit="MWH"/>
      <Data period="6" value="23.4" unit="MWH"/>
      <Data period="7" value="78.9" unit="MWH"/>
      <Data period="8" value="0" unit="MWH"/>
      <Data period="9" value="0" unit="MWH"/>
      <Data period="10" value="30.1" unit="MWH"/>
    </ProfileData>
    <Party id="24XDSO---------Q" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 94** Response containing IDA evaluation per periods

### 4.9.4 Notification of evaluations per day (E-05_02)

Notification of evaluations for day is carried out by sending a request in *CDSREQ.961* structure (message-code=961) and response with data in *RESPONSE.962* and *ISOTEDATA.963* structure.

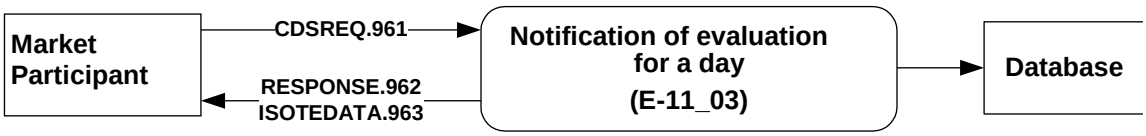


**Figure 47** Schéma oznámenia vyhodnotení VDA za deň pre ÚT

*CDSREQ.961*

Request is formulated for specific trading day (trade-day) and optional possibility to fill specific auction (auction-id) is filled out according to the specification of CDSREQ structure

```
<CDSREQ id="45t" message-code="951" date-time="2023-12-01T01:18:33"
        dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <ReceiverIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <Trade trade-day="2023-12-01" />
</CDSREQ>
```

**Example 95** Request for downloading summary IDA evaluations per day

*RESPONSE.962*

In case of successful or failed request processing, response is returned according to specification of RESPONSE structure, where message-code=962 can be found in the header.

```
<RESPONSE id="8a848bad-46c2-4e3d-ab78-2c7c4545a21" message-code="962"
        date-time="2023-12-01T14:20:40Z" dtd-version="1" dtd-release="1"
        xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/ut/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15"/>
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Reason code="0" type="A03"/>
</RESPONSE>
```

**Example 96** Response on success of requesting summary daily IDA evaluation

*ISOTEDATA.963*

In case of successful request processing, the structure of trading results is returned as it was registered in the system, where message-code=963 can be found in the header.

The following result types are used in *ProfileData /@profile-role* attribute:

- SP02 – clearing/payment for purchased electricity,
- SC02 – amount of purchased electricity,
- SP03 – clearing/payment for sold electricity,
- SC03 – amount of sold electricity,
- SP05 – fee for traded electricity,
- SC05 – amount of traded electricity (the sum of sold and purchased electricity),
- ST16 – monthly fee for access on day-ahead market; the last day of month is specified,
- SP90 – fee for trading transactions relating to data manipulation,
- SC90 – amount of trading transactions relating to data manipulation,
- SP91 – fee for trading transactions relating to use of automated interfaces,
- SC91 – amount of trading transactions relating to use of automated interfaces,
- SP92 – clearing/payment for purchased electricity for negative prices (positive number),
- SC92 – quantity of purchased electricity for negative prices (positive number),
- SP93 – clearing/payment for sold electricity for negative prices (positive number),

- SC93 – quantity of sold electricity for negative prices (positive number).

```
<ISOTEDATA id="526539ee-9bb7-465b-8e5c-0b660674f0f" message-code="963"
       date-time="2023-12-01T14:20:40Z" dtd-version="1" dtd-release="1" answer-required="false"
       xmlns="http://sfera.sk/ws/xmtrade/isot/interfaces/evaluations/types/2009/04/01">
  <SenderIdentification id="24X-OT-SK------V" coding-scheme="15" />
  <ReceiverIdentification id="24XDSO---------Q" coding-scheme="15"/>
  <Reference id="45t"/>
  <Trade trade-day="2023-12-01">
    <ProfileData profile-role="SP02">
      <Data period="0" value="27875.987" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC02">
      <Data period="0" value="1146.7" unit="MWH"/>
    </ProfileData>
    <ProfileData profile-role="SP03">
      <Data period="0" value="17278.838" unit="EUR"/>
    </ProfileData>
    <ProfileData profile-role="SC03">
      <Data period="0" value="639.3" unit="MWH"/>
    </ProfileData>
    <Party id="24XDSO---------Q" role="TO"/>
  </Trade>
</ISOTEDATA>
```

**Example 97** Response containing summary IDA evaluation per day

## 4.10 WebSocket protocol

WebSocket messages automatically inform market participants about changes that have occurred with their own orders, changes in the order book, or changes in available cross-border transmission capacities.

### 4.10.1 Processing level

A user connected to the WebSocket interface has the ability to receive information through which they are notified of the following events:

- Successful creation of their own order – data stream E-12_01,
- Change of their own order (status change) – data stream E-12_01,
- Changes in the order book (changes in available quantity and the last price/quantity for periods for which the change is sent) – data stream E-12_02,
- Changes in available cross-border transmission capacities – data stream E-12_03.

### 4.10.2 Status change/Creation of own order (E-12_01)

Within this data stream, a message is sent via the WebSocket protocol that notifies the market participant about changes in the status of their existing order or the creation of a new order. This mechanism allows for instant notifications of relevant changes related to the participant's own orders in real time.

For more detailed information on how messages are transmitted via a different protocol, we recommend referring to the chapter (Notifications AMQP 4.8.2 Status change/Creation of Own Order (E10-01)). This chapter provides a detailed description of the process of sending and receiving notifications via the AMQP protocol.

### *Order-change*

In the case of the creation of a new own order or a change of status, the order's data structure is sent via the WebSocket protocol in JSON format.

The CorrelationId field contains a reference to the message that initiated the creation or change of the order, enabling easy tracking and backtracking of specific actions. The id field provides a unique identifier for the created order, which the user can save and use to match it with the order in their system, whether created via WebSocket, web services, or Web API. The order processing status is indicated by the isPending field.

Message 1:


```
{
   "payload": {
      "id": 3,
      "type": "simple",
      "productType": 15,
      "deliveryDay": "2024-11-20",
      "deliveryStart": "2024-11-20T22:15:00Z",
      "deliveryEnd": "2024-11-20T22:30:00Z",
      "direction": "sell",
      "quantity": 1,
      "price": 1,
      "status": "inactive",
      "note": "just my note",
      "clientOrderId": "00001",
      "isPending": true,
      "remainingQuantity": 1,
      "createdAt": "2024-11-20T10:16:10",
      "updatedAt": "2024-11-20T10:16:10",
      "createdBy": "tester01",
      "action": "added",
      "correlationId": "test01"
   },
   "type": "order-change"
}
```


Message 2:


```
{
   "payload": {
      "id": 3,
      "type": "simple",
      "productType": 15,
      "deliveryDay": "2024-11-20",
      "deliveryStart": "2024-11-20T22:15:00Z",
      "deliveryEnd": "2024-11-20T22:30:00Z",
      "direction": "sell",
      "quantity": 1,
      "price": 1,
      "status": "active",
      "note": "just my note",
      "clientOrderId": "00001",
      "isPending": false,
      "realizedQuantity": 0,
      "remainingQuantity": 1,
      "createdAt": "2024-11-20T10:16:10",
      "updatedAt": "2024-11-20T10:16:10",
      "createdBy": "test01",
      "action": "activated",
      "correlationId": "test01"
   },
   "type": "order-change"
}
```


**Example 98** Example of an order-change message after adding a new order.

```
{
    "payload": {
        "id": 1,
        "type": "simple",
        "productType": 60,
        "deliveryDay": "2024-11-19",
        "deliveryStart": "2024-11-19T15:00:00Z",
        "deliveryEnd": "2024-11-19T16:00:00Z",
        "direction": "buy",
        "quantity": 1,
        "price": 1,
        "status": "inactive",
        "note": "just my note",
        "clientOrderId": "00001",
        "isPending": false,
        "realizedQuantity": 0,
        "remainingQuantity": 1,
        "createdAt": "2024-11-19T15:03:08",
        "updatedAt": "2024-11-19T15:06:17",
        "createdBy": "test01",
        "action": "deactivated",
        "correlationId": "test01"
    },
    "type": "order-change"
}
```


**Example 99** Example of an order-change message after deactivating a own order.


```
{
    "payload": {
        "id": 2,
        "type": "simple",
        "productType": 15,
        "deliveryDay": "2024-11-20",
        "deliveryStart": "2024-11-20T11:30:00Z",
        "deliveryEnd": "2024-11-20T11:45:00Z",
        "direction": "sell",
        "quantity": 10,
        "price": 10,
        "status": "matched",
        "note": "just my note",
        "clientOrderId": "00001",
        "isPending": false,
        "realizedQuantity": 10,
        "remainingQuantity": 0,
        "createdAt": "2024-11-20T10:32:00",
        "updatedAt": "2024-11-20T10:32:00",
        "createdBy": "testet01",
        "action": "matched",
        "correlationId": "test01"
    },
    "type": "order-change"
}
```


**Example 100** Example of order-change message after trading own order.

***Message reception***

The message is received via the WebSocket protocol. The message includes:

- Message type: order-change
- Payload field, which contains the order details.

***Message pairing***

- Based on the correlationId value, the message can be paired with the previous request made via the WEB API, WebSocket, or Web Service.
- The order ID can then be stored for future reference.

### 4.10.3 Change of the order book data (E-12_02)

Notification via WebSocket protocol is sent within dataflow and inform market participant about order book status changes The message is a response to a change in the order book, which may be triggered by one of the following reasons, resulting in any change in the quantity in one or more trading periods:

- Market participant (adding, trading, modification of an order),
- Change caused by other market participant (adding, trading, modification of an order),
- System (expiration of and order, more precisely time period which can cause annulment of available amount in concrete time period).

For more detailed information on the method of message transmission using another protocol, we recommend referring to the chapter **AMQP Notifications 4.8.3 Change of the Order Book Data (E10-02)**. This chapter provides a detailed description of the process of sending and receiving notifications within the AMQP protocol.

*OrderBook-change*

This type of message is not directed to a specific user or market participant but is distributed to all users connected to the WebSocket interface. The content of the message includes information about a change in quantity at a specific price within a defined time period, on both the buy and sell sides.

```
{
   "payload": {
      "seqNo": 3278,
      "timeDelta": 3002,
      "data": [
         {
            "period": {
               "start": "2024-11-20T19:00:00Z",
               "end": "2024-11-20T20:00:00Z",
               "isBlock": false,
               "tradingEnd": "2024-11-20T18:30:00Z"
            },
            "statistics": {
               "lastTradeTime": "2024-11-19T15:41:28.885Z",
               "lastPrice": 291.27,
               "maxPrice": 291.27,
               "minPrice": 291.26,
               "totalVolume": 30.0,
               "lastQuantity": 10.0,
               "priceDirection": 1
            },
            "ownStatistics": {
               "buy": {},
               "sell": {
                  "quantity": 0,
                  "remainingQuantity": 291.27,
                  "weigtedAveragePrice": 0,
               }
            },
            "buyChanges": [
               {
                  "index": 0,
                  "action": "add",
                  "price": -94.84,
                  "quantity": 5,
                  "quantityDelta": 5,
                  "ownQuantity": 0,
                  "ownQuantityDelta": 0
               }
            ],
            "sellChanges": [
               {
                  "index": 0,
                  "action": "add",
                  "price": 291.27,
                  "quantity": 0.7,
                  "quantityDelta": 0.7,
                  "ownQuantity": 0,
                  "ownQuantityDelta": 0
               },
               {
                  "index": 0,
                  "action": "add",
                  "price": 291.26,
                  "quantity": 5,
                  "quantityDelta": 5,
                  "ownQuantity": 0,
                  "ownQuantityDelta": 0
               }
            ],
            "blockOrderChanges": [],
            "action": "update"
         }
      ]
   },
   "type": "orderbook-change"
}
```


**Example 101** Example of an *orderbook-change* message about a change in quantity during a period.

***Message reception***

The message is received via WebSocket protocol. The message includes:

- Message type: `orderbook-change`.
- Payload field: contains details about changes in the order books.

***Data consistency verification***

- **Sequence (seqNo):** The client checks if the seqNo (sequence number) follows the previous message.
  - If seqNo is inconsistent, the client should request a new order book snapshot via the orderbook-snapshot request.
- **Time Difference (timeDelta)**: Provides the client with the time delay between messages in milliseconds.

***Processing changes***

The message contains changes in various parts of the order book:

1. **Period**:
   Includes details about the period affected by the changes, such as:
   - Start and end time of the period.
   - Whether the order is simple or block.
   - Trading end time (tradingEnd).
2. **Statistics:**
   - Includes the last price (`lastPrice`), maximum and minimum prices, total traded quantity, and price direction (`priceDirection`). If there are no changes in statistics, the property is null.
3. **Buy-side changes:**
   - Each change includes**:**
     - index: Position in the order list (market depth).
     - action: Type of action (add, update, remove).
     - price: Order price.
     - quantity: Current quantity after the change.
     - quantityDelta: Change in quantity compared to the previous state.
     - ownQuantity: Client's own quantity at the given price.
     - ownQuantityDelta: Change in the client's own quantity compared to the previous state.
4. **Sell-side changes:**
   - Structure is identical to that of buyChanges.
5. **Block order changes:**
   - Information about block orders, if any exist.
6. **Action**:
   - Indicates that the message represents an update.

### 4.10.4 Change in cross-border capacities data H2H (E-12_03)

Message informing a market participant about changes in available cross-border transmission capacities (H2H) is sent via the WebSocket protocol during the operation of the cross-border intraday market.

For more detailed information on the method of message transmission using another protocol, we recommend referring to the chapter **Change in cross-border capacities data H2H (E-10_03)**. This chapter provides a detailed description of the process for sending and receiving notifications within the AMQP protocol.

```
{
   "payload":{
      "data":[
         {
            "eic":"10YCZ-CEPS-----N",
            "areaName":"CEPS",
            "countryCode":"CZ",
            "deliveryDay":"2024-11-20",
            "deliveryStart":"2024-11-20T20:00:00Z",
            "deliveryEnd":"2024-11-20T21:00:00Z",
            "availableCapacityIn":2319,
            "availableCapacityOut":0
         },
         {
            "eic":"10YCZ-CEPS-----N",
            "areaName":"CEPS",
            "countryCode":"CZ",
            "deliveryDay":"2024-11-20",
            "deliveryStart":"2024-11-20T21:00:00Z",
            "deliveryEnd":"2024-11-20T22:00:00Z",
            "availableCapacityIn":4104,
            "availableCapacityOut":0
         },
         {
            "eic":"10YCZ-CEPS-----N",
            "areaName":"CEPS",
            "countryCode":"CZ",
            "deliveryDay":"2024-11-20",
            "deliveryStart":"2024-11-20T22:00:00Z",
            "deliveryEnd":"2024-11-20T23:00:00Z",
            "availableCapacityIn":2195,
            "availableCapacityOut":2327
         }
      ]
   },
   "type":"hubtohub-change"
}
```


**Example 102** Example message H2H IDM.

### 4.10.5 Error messges

Through the WebSocket interface, the client can be notified about:

- validation errors when creating or modifying orders (message type: order-error).
- general errors, such as an incorrect message format (message type: error).

Messages include:

- **Error Code**: unique identifier specifying the type of error.
- **Readable Error Message**: description of the error in English.
- **Additional Details** (optional): Further information providing more context about the issue.

```
{
    "payload": {
        "correlationId": "tst3",
        "code": "ValidationProblem",
        "message": "Validation problems occurred.",
        "errors": {
           "00001": [
           {
            "code":"ExpTimeEndRule",
            "message": "Order expiration time cannot be later than period trading end.",
            "messageArgs": []
           }]
        }
    },
    "type": "order-error"
}
```


**Example 103** Example of an error message when requesting the creation of an order for a closed period.


```
{
    "payload": {
        "correlationId": "tst4",
        "code": "OrderNotFound",
        "message": "Order with ID 2172507 not found"
    },
    "type": "error"
`
```


**Example 104** Example of an error message when a client requests a non-existent order.


```
{
    "payload": {
        "reset": "5",
        "policy": "50;w=10",
        "limit": "50",
        "code": "RateLimitExceeded",
        "message": "Rate limit exceeded. Try again later."
    },
    "type": "ratelimit-error"
}
```


**Example 105** Example of an error message when a client exceeds rate limit.

## 4.11 Retrieval of MCC values

MCC values are available for a market participant via operations for retrieval of these values per hours for a specific trading day.

### 4.11.1 Processing level

Transmission systems operators report capacities assigned for the purpose of interconnection of the CR and SR day-ahead markets for the following day in the form of market coupling (MCC - Market

Coupling Capacity). These capacities are stated for each hour of the following trading day for cross-border trading areas which are connected via Market Coupling. Capacities are mentioned for every profile in both directions.

Daily deadline for reception of MCC from transmission system operators is set to 9:00 am. In exceptional cases, transmission system operator is entitled to update these values by the launch time for order coupling. Market participant is notified of eventual update of MCC values after the standard deadline via ISOT system.

Functionality for downloading MCC values is not available within the web service *StatusRequest* and method *DownloadMCC* in case of requesting data for the trading day which is under the CORE regime. During the CORE regime, the cross-border capacities are calculated by a new method of so called *flow-based* matrix. Hyperlink for access to relevant *flow-based* matrix of cross-border capacities is available one the OKTE's website.

### 4.11.2 Notification of MCC values (E-01_02)

Notification of MCC values is carried out by request in *ESR.StatusRequest* structure and response with data in *ECAN.CapacityDocument* and *EAD.Acknowledgement* structures.


**Figure 48** Notification schema of MCC values

In case of failed processing of *StatusRequest* request, only Acknowledgement containing reason for the fail is returned. In case of successful processing, confirming Acknowledgement and CapacityDocument are returned provided that MCC values exist for a given trading day.

### 4.11.3 ESR.StatusRequest

*StatusRequest* structure is used in general for retrieval of process status or information on process in accordance with *ENTSO-E* standards. The structure is in accordance with *ESR V1R1* standard and consists a single part:

- *StatusRequest* – contains details related to the entire message.

#### *StatusRequest*

Header of request for retrieval of status information contains values of particular attributes according to the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| Message Identification | Identification | Unique message identification<br>Maximum 35 characters. | Required |
| Message Type | A13 | Type of message, which status is required.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Process Type | A07 | Type of process, which the message relates to. | Required |

| Element | Value | Description | Use |
|---|---|---|---|
| | | According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | |
| Sender Identification | Sender EIC | Message sender identification.<br><br>Entity EIC is used.<br><br>Maximum 16 characters. | Required |
| Sender Role | A01 | Message sender role.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Receiver Identification | 24X-OT-SK------V | Message receiver identification.<br><br>EIC = 24X-OT-SK------V is used.<br><br>Maximum 16 characters. | Required |
| Receiver Role | A07 | Message receiver role.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Message Date And Time | YYYY-MMDDTHH:MM:SSZ | Date of message sending.<br><br>Date and time format:<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• MM – minute.<br><br>Date and time items are inserted into the message in UTC (Universal Time). | Required |

| Element | Value | Description | Use |
|---|---|---|---|
| Requested Time Interval | YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | Time interval = trading day, for which information is requested.<br><br>Date and time format:<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• MM – minute.<br><br>Example for trading day 20.08.2009: 2009-08-19T22:00Z/2009-08-20T22:00Z<br><br>Date and time items are inserted into the message in UTC (Universal Time). | Required |

```
<ns:StatusRequest DtdVersion="1" DtdRelease="1"
xmlns:ns="http://sfera.sk/ws/xmtrade/isot/statusrequest/services/2009/04/01"
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/esrv1r1/2009/04/01">
  <MessageIdentification v="ce75631a99b045e98322d8912b0090b1"/>
  <MessageType v="A13"/>
  <ProcessType v="A07"/>
  <SenderIdentification v="11XSEBRATISLAVA4" codingScheme="A01"/>
  <SenderRole v="A01"/>
  <ReceiverIdentification v="24X-OT-SK------V" codingScheme="A01"/>
  <ReceiverRole v="A07"/>
  <MessageDateTime v="2009-08-18T14:35:07Z"/>
  <RequestedTimeInterval v="2009-08-15T22:00Z/2009-08-16T22:00Z"/>
</ns:StatusRequest>
```

**Example 106** Request for retrieval of MCC values

### 4.11.4 EAD.Acknowledgement

*Acknowledgement* structure (ACK, Acknowledgement Document) is used for confirmation or rejection of request reception within the communication in *ENTSO-E* structures with the ISOT system. The structure is in accordance with *EAD V5R0* standard and consists of the following parts:

- *Acknowledgement document* – contains details related to the entire message.
- *Time Series Rejection* – contains identification details of respective time series (not used in this context).
- *Time Interval Error* – contains identification details of respective value in time series (not used in this context).
- *Reason* – contains information on acceptance or rejection of the request.

Within notification of MCC values, only *Reason* part of the structure is used and the remaining parts are not relevant in this communication.

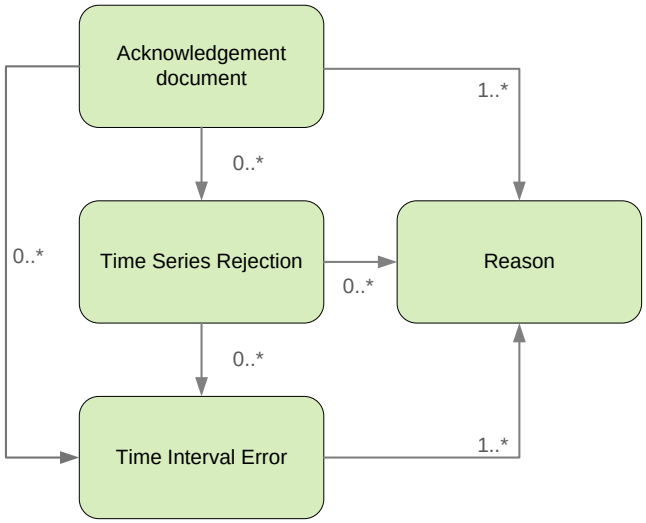


**Figure 49** Scheme of Acknowledgement structure

*Acknowledgement document*

Header of acknowledgement document contains values of particular attributes in accordance with the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| Document Identification | Identification | Document identification.<br><br>Maximum 35 characters. | Required |
| Document Date And Time | YYYY-MMDDTHH:MM:SSZ | Date of document sending.<br><br>Date and time format:<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• MM – minute.<br><br>Date and time items are inserted into the document in UTC (Universal Time). | Required |
| Sender Identification | 24X-OT-SK------V | Document sender identification.<br><br>EIC = 24X-OT-SK------V is used.<br>Maximum 16 characters. | Required |
| Sender Role | A07 | Document sender role.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Receiver Identification | Entity EIC | Message receiver identification.<br><br>Entity EIC is used.<br><br>Maximum 16 characters. | Required |

| Element | Value | Description | Use |
|---|---|---|---|
| Receiver Role | A01 | Message receiver role.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Receiving Document Identification | Identification | Identification of the original message, which the acknowledgement document refers to. | Required |
| Date Time Receiving Document | YYYY-MMDDTHH:MM:SSZ | Reception date of the original message in UTC (Universal Time), which the acknowledgement document refers to. | Optional |

*Reason*

Information about acceptance or rejection of the message contains values of particular attributes in accordance with the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| Reason Code | AXY | Information on acceptance or rejection of the original message and reasons for its rejection.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>At the message level:<br>• A01 - Message fully accepted,<br>• A02 - Message fully rejected,<br>• A04 - Time interval incorrect,<br>• A51 - Message identification or version conflict,<br>• A53 - Receiving party incorrect,<br>• A78 - Sender identification and/or role invalid,<br>• A79 - Process type invalid,<br>• A94 - Document cannot be processed by receiving system. | Required |
| Reason Text | open text | Additional text justification. | Optional |

```
<Acknowledgement DtdVersion="5" DtdRelease="0">
  <DocumentIdentification v="3e26b8eb34d84eec85de683bdf4ceee9" xmlns=""/>
  <DocumentDateTime v="2009-08-18T14:35:07Z" xmlns=""/>
  <SenderIdentification v="24X-OT-SK------V" codingScheme="A01" xmlns=""/>
  <SenderRole v="A07" xmlns=""/>
  <ReceiverIdentification v="11XSEBRATISLAVA4" codingScheme="A01" xmlns=""/>
  <ReceiverRole v="A01" xmlns=""/>
  <ReceivingDocumentIdentification v="ce75631a99b045e98322d8912b0090b1" xmlns=""/>
  <DateTimeReceivingDocument v="2009-08-18T14:35:08Z " xmlns=""/>
  <Reason xmlns="">
    <ReasonCode v="A01"/>
    <ReasonText v=""/>
  </Reason>
</Acknowledgement>
```

**Example 107** ACK, acknowledgement of successful reception for retrieval of MCC values

### 4.11.5 ECAN.CapacityDocument

*CapacityDocument* structure is used for retrieval of MCC values and consists of the following parts in accordance with *ECAN V4R0* standard:

- *CapacityDocument* – contains details related to the entire message (header),
- *CapacityTimeSeries* – contains details about time series,
- *Period* – contains details about time periods,
- *Interval* – contains specific MCC details in particular time units,
- *Reason* – contains details about reasons for invalidity of document or specific interval (not used).

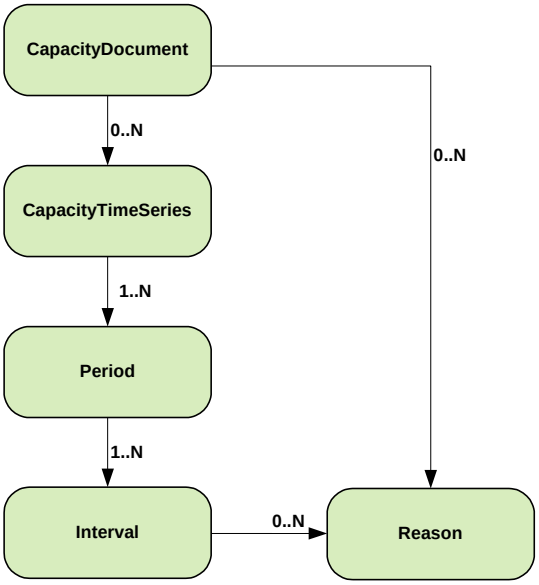


**Figure 50** Schema of CapacityDocument structure

*CapacityDocument*

Header contains values of particular attributes in accordance with the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| Document Identification | Identification | Message identification.<br><br>Maximum 35 characters. | Required |
| Document Version | non-negative number | Version of the registered document. | Required |
| Document Type | A13 | Document type (Interconnection Capacity).<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |

| Element | Value | Description | Use |
|---|---|---|---|
| ProcessType | A07 | Process type (Capacity Allocation).<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Sender Identification | 24X-OT-SK------V | Message sender identification.<br><br>EIC = 24X-OT-SK------V is used<br>Maximum 16 characters. | Required |
| Sender Role | A07 | Message sender role.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Receiver Identification | Entity EIC | Message receiver identification.<br><br>Entity EIC is used.<br><br>Maximum 16 characters. | Required |
| Receiver Role | A01 | Message receiver role.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Creation Date Time | YYYY-MMDDTHH:MM:SSZ | Date of document creation in the system.<br><br>Date and time format:<br>• YYYY – year,<br>• MM – month,<br>• DD – day,<br>• HH – hour,<br>• MM – minute.<br><br>Date and time items are inserted into the message in UTC (Universal Time). | Required |

| Element | Value | Description | Use |
|---|---|---|---|
| Capacity Time Interval | YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | Time interval = trading day, for which information is returned.<br><br>Date and time format:<br>• YYYY – year,<br>• MM – month,<br>• DD - day,<br>• HH – hour,<br>• MM – minute.<br><br>Example of trading day 20.08.2009: 2009-08-19T22:00Z/2009-08-20T22:00Z<br><br>Date and time items are inserted into the message in UTC (Universal Time). | Required |
| Domain | 10YDOM-CZ-DE-SKK | Domain | Required |

*CapacityTimeSeries*

It covers details for cross-border profiles. Two structures are returned for profiles: *CapacityTimeSeries* or two time series. Values of particular attributes are in accordance with the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| TimeSeries Identification | Identification | Identification of time series.<br><br>Maximum 35 characters. | Required |
| Business Type | A31 | Business type (Offered Capacity).<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| Product | 8716867000016 | Product identification.<br><br>According to ENTSO-E General Code List for Data Interchange.<br><br>Maximum 3 characters. | Required |
| InArea | 10YSK-SEPS-----K / 10YCZ-CEPS-----N | EIC code of area, to which energy is coming. | Required |
| OutArea | 10YCZ-CEPS-----N / 10YSK-SEPS-----K | EIC code of area, out of which energy is coming. | Required |
| MeasureUnit | MAW | Unit of values.<br>MAW = MW | Required |

| Element | Value | Description | Use |
|---|---|---|---|
| AuctionIdentification | DAY-AHEAD_IMPLICIT | Identification of auction = implicit auction. | Required |

*Period*

It covers details on particular time period (trading day). Values of particular attributes are in accordance with the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| TimeInterval | YYYY-MM-DDTHH:MMZ/ YYYY-MM-DDTHH:MMZ | Time interval = trading day, for which information are retrieved.<br><br>Date and time format:<br>• YYYY - year,<br>• MM - month,<br>• DD – day,<br>• HH – hour,<br>• MM – minute.<br><br>Example for trading day 20.08.2009: 2009-08-19T22:00Z/2009-08-20T22:00Z<br><br>Date and time items are inserted into the message in UTC (Universal Time). | Required |
| Resolution | PT60M | Resolution of period = hour. | Required |

*Interval*

It contains details on particular capacities in respective hours. Values of particular attributes are in accordance with the following table.

| Element | Value | Description | Use |
|---|---|---|---|
| Pos | positive number | Position = hour within a given trading day.<br><br>It starts from 1. | Required |
| Qty | number | Quantity | Required |

```
<CapacityDocument DtdVersion="4" DtdRelease="0">
  <DocumentIdentification v="7a376855c4644ab990d190e9ccdfbe46"
     xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <DocumentVersion v="1"
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <DocumentType v="A13"
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <ProcessType v="A07" xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <SenderIdentification v="24X-OT-SK------V" codingScheme="A01"
     xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <SenderRole v="A07" xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <ReceiverIdentification v="11XSEBRATISLAVA4" codingScheme="A01"
     xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <ReceiverRole v="A01"
xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <CreationDateTime v="2009-08-18T14:35:07Z"
     xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <CapacityTimeInterval v="2009-08-15T22:00Z/2009-08-16T22:00Z"
     xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
  <Domain v="10YDOM-CZ-DE-SKK" codingScheme="A01"
     xmlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01"/>
```

**Example 108** Retrieved MCC values - header

```
    <CapacityTimeSeries mlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01">
       <TimeSeriesIdentification v="103"/>
       <BusinessType v="A31"/>
       <Product v="8716867000016"/>
       <InArea v="10YSK-SEPS-----K" codingScheme="A01"/>
       <OutArea v="10YCZ-CEPS-----N" codingScheme="A01"/>
       <MeasureUnit v="MAW"/>
       <AuctionIdentification v="DAY-AHEAD_IMPLICIT"/>
       <Period>
          <TimeInterval v="2009-08-15T22:00Z/2009-08-16T22:00Z" xmlns=""/>
          <Resolution v="PT60M" xmlns=""/>
          <Interval xmlns=""><Pos v="1"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="2"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="3"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="4"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="5"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="6"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="7"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="8"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="9"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="10"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="11"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="12"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="13"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="14"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="15"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="16"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="17"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="18"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="19"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="20"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="21"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="22"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="23"/><Qty v="80"/></Interval>
          <Interval xmlns=""><Pos v="24"/><Qty v="80"/></Interval>
       </Period>
    </CapacityTimeSeries>
    <CapacityTimeSeries mlns="http://sfera.sk/ws/xmtrade/isot/common/types/ecan/2009/04/01">
       <TimeSeriesIdentification v="101"/>
       <BusinessType v="A31"/>
       <Product v="8716867000016"/>
       <InArea v="10YCZ-CEPS-----N" codingScheme="A01"/>
       <OutArea v="10YSK-SEPS-----K" codingScheme="A01"/>
       <MeasureUnit v="MAW"/>
       <AuctionIdentification v="DAY-AHEAD_IMPLICIT"/>
       <Period>
          <TimeInterval v="2009-08-15T22:00Z/2009-08-16T22:00Z" xmlns=""/>
          <Resolution v="PT60M" xmlns=""/>
          <Interval xmlns=""><Pos v="1"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="2"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="3"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="4"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="5"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="6"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="7"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="8"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="9"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="10"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="11"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="12"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="13"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="14"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="15"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="16"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="17"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="18"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="19"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="20"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="21"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="22"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="23"/><Qty v="100"/></Interval>
          <Interval xmlns=""><Pos v="24"/><Qty v="100"/></Interval>
       </Period>
    </CapacityTimeSeries>
 </CapacityDocument>
```

**Example 109** Retrieved MCC values – time series

# 5 LIST OF FIGURES

# 6 LIST OF TABLES

# 7 LIST OF EXAMPLES