DH API documentation for guaranteed supplier

Content

1 Documentation version history

The table below provides information on document version history:

Version	Date	Description
1.0.0	2023-12-07	Initial document version.

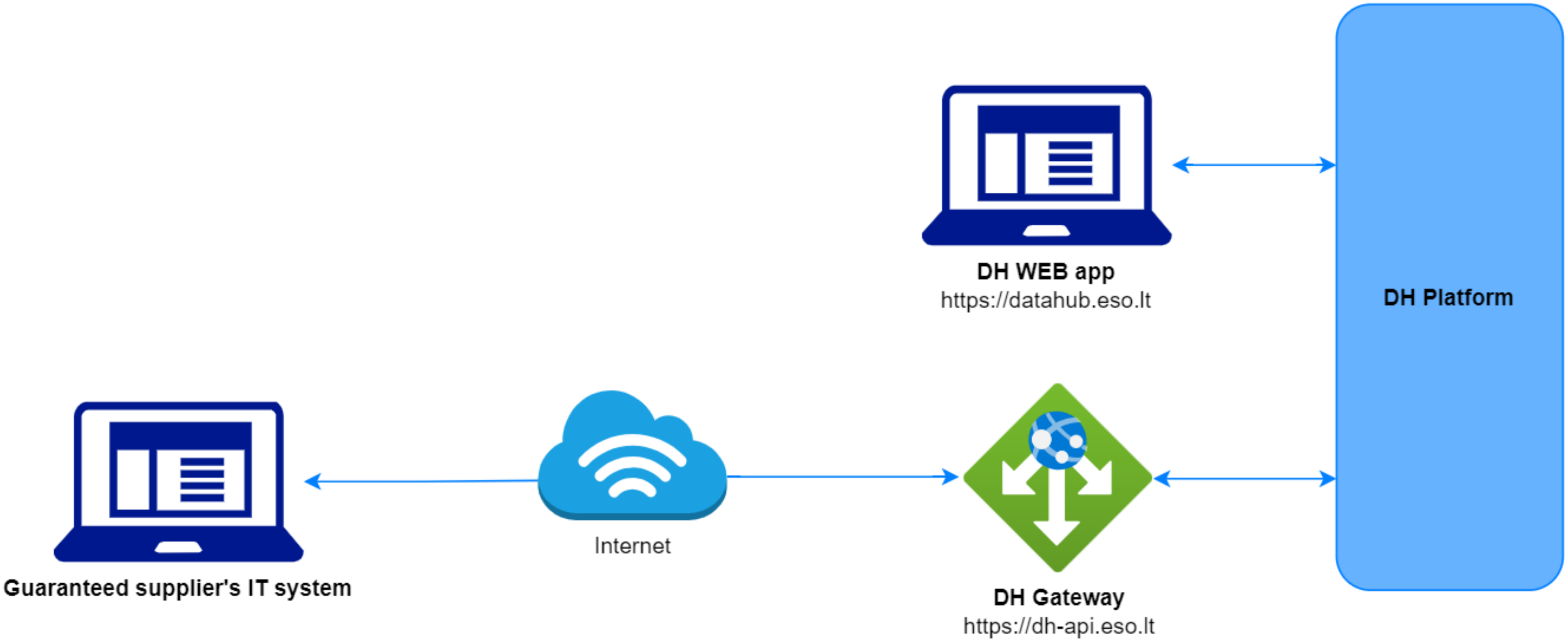
Note: Changes in table marked in white are already deployed, marked in green will be deployed in near future.

2 Preface

The Common Data Exchange Platform (hereinafter referred as DH Platform) Gateway is a component enabling the guaranteed suppliers to directly access DH Platform from within their IT systems and thus helps perform their activities more efficiently.

DH Gateway provides open standards-based interfaces allowing the guaranteed suppliers themselves (or with outside assistance) integrate their IT systems with DH Platform.

This document provides technical information on DH Gateway interfaces which is needed to integrate guaranteed suppliers' information systems with DH Platform.



3 Definitions and abbreviations

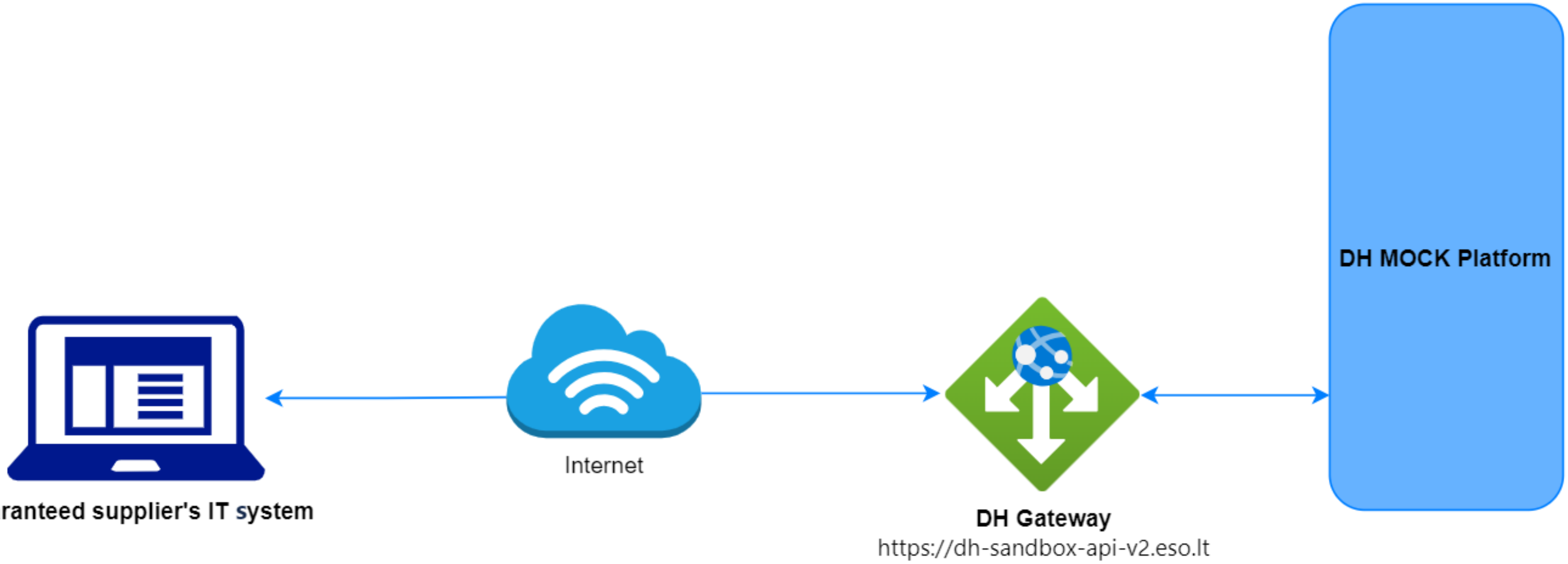
Definition / abbreviation	Description
DH Gateway	DH Platform component enabling guaranteed supplier IT systems to directly access the platform and achieve a higher-level degree of process automation.
DH, DH Platform	Common Data Exchange Platform.
DSO, ESO	Energy distribution system operator – AB „Energijos skirstymo operatorius“.
Object	A site where electricity consumption takes place.
DH WEB app	It is a web application that provides a graphical user interface (GUI) for working with the DataHub system.

4 Environments

There are two DH Gateway environments the guaranteed supplier might access:

- “Sandbox” environment
- Production environment

DH Sandbox environment made of Mock API Gateway with mock requests and responses (scenarios). There is no connection to database or any data source, all possible requests and answers are hard coded into mock API source code and has no any data selection logic or rules. This data is real depersonalized data from DSO customers. Sandbox requests and responses scenarios will be provided in additional document, and it should be used just for preparation to integrate with DH production API environment or testing purposes.


DH MOCK Platform
Guaranteed supplier's IT system
Internet
DH Gateway
https://dh-sandbox-api-v2.eso.lt

DH Platform also has WEB interface, which is connected to DH Production Gateway. All environments are provided in the table:

Environment	Swagger Link	WEB Interface
Production	https://dh-api.eso.lt/swagger-ui.html	https://datahub.eso.lt/
Sandbox	https://dh-sandbox-api-v2.eso.lt/swagger-ui.html#/	-

5 Digital certificates

In both the testing and production environments of the DH Gateway component, the identity of the guaranteed supplier is established using a TOKEN, which the guaranteed supplier’s information system must provide each time the DH Gateway network service is called.

To get started:

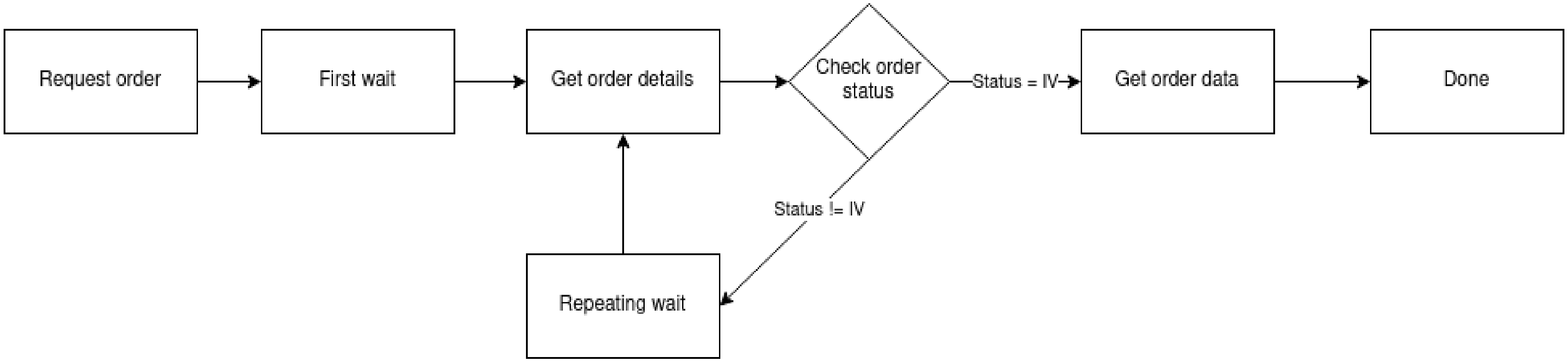
1. The DSO responsible person sends the JWT key (JSON Web Token) to be used with each request to the DH API.
2. To make requests to the DH Gateway API - the TOKEN submission in the case of curl takes place.

6 Recommendation for API client

6.1 ASYNC

Async pattern is mainly used for data orders: https://dh-api.eso.lt/swagger-ui/index.html#/-guaranteed-supplier-order-controller (will be deployed in 2023-12-14)

Client side should implement following process with steps:



Step descriptions

Step name	Description	Endpoint	Request example	Response example

Request order	Submit new data order. Request will return order id which will be used in other steps for getting order details and order data.	POST /gateway/guaranteed-supplier/order/**yyyyyyyyyy**
		where **yyyyyyyyyy** is order type:
		• data-hr-15min-obj-lvl
			POST /gateway/guaranteed-supplier /order/ data-hr-15min-obj-lvl
			Body:
			{
			"consumptionCategories" : [
			"P+"
			],
			"dateFrom": "2023-11-01",
				HTTP status 201
				{
				"orderId": 10000001
				}

			"dateTo": "2023-11-30", "interval": "HOUR", "objectNumbers": ["11111111", "22222222"] }	
First wait	Wait for some period of time after order submission. This step is needed because after order request it takes some time to process it and there is no reason to try get status immediately after order submission. First wait duration depends on order type and parameters. If order collects more data, it can take minutes to prepare data. For duration recommendations look at Recommendations.			

Get order details	Request to get order details. This request is needed to get order latest status which is stored in field "latestStatus".
	Posible values for "latestStatus":
	• P - Submitted order
	• V - Order in progress
	• IV - Order is finished and data are prepared.
	• K - Order has errors
		POST/gateway/guaranteed-supplier/order/list	POST /gateway/guaranteed-supplier /order/list
			{
			"orderId": 10000001
			}
				HTTP status 200
				[
				{
				"orderId": 10000001,
				"orderType": "data-hr-15min-obj-lvl",
				"submittedDate": "2023-12-07T08:49:29.117",
				"dateFrom": "2023-11-01",
				"dateTo": "2023-11-30",
				"orderParameters": "{\"consumptionCategories\“: [\"P+\"],\"objectNumbers\":[\"111111

				11\",\"22222222\"],\"interval\":\"HOUR\"}", "latestStatus": "V", "statusDate": "2023-12-07T08:49:30.446", "expireDate": "null, "auto": false, "userName": "PUBLIC" }]
Check order status	Logic operation to check order "latestStatus" field value. If value equals to "IV" it means that order data is prepared. Otherwise order data is not ready algorithm should go to step "Repeating wait".			
Repeating wait	Wait for some period of time after order status check when status was not equal to "IV". This step is needed because repetitive status check without wait can do unneeded load to DH system. For duration recommendations look at Recommendations.			

Get order data	Get order data.
	Note: If order has too many data, then pagination should be used. Default and max page size is 10 000 records (usually it's objects).
		GET/gateway/guaranteed-supplier/order/**zzzzzzzz/yyyyyyyyyy?**first=**oooooo**&count=**ssssss**
		where:
		• **zzzzzzzz** is order Id
		• **oooooo** is offset position
		• **ssssss** is page size
		yyyyyyyyyy is order type:
		• data-hr-15min-obj-lvl
			GET/gateway/guaranteed-supplier/order/10000001/ data-hr-15min-obj-lvl?first=0&count=10000	HTTP status 200 with order data in JSON format.
				If order content is empty get method will return HTTP status 400 with message.
				{
				"code": 2018,
				"text": "There is no data for the selected search parameters, the response is empty."
				}

6.1.1 DataHub order processing retry policy

If any issues appear during order data processing stage the process stops, and order gets status K. DH uses retry policy for all orders with status K.

- Retries order process after 5 minutes.
- Retries order process 300 times.
- For failed orders retry policy will be working in total 25 hours (5 min * 300).
- Retry policy will stop work after 25 hours and order will be left with status K.

This is needed because issues can appear in data preparation stage of couple reasons:

- DH technical problem - for example one of DH integrations was down or contract was changed, data integrity violations and etc.
- Incompatible business logic - for example order got into not defined use case and use case should be adopted to order.

In most cases order processing retry will solve problem. But there are cases like "Incompatible business logic" when additional human interaction is needed to finish order job. We are tracking such an order and fixing them, but fixing might take some hours or even days. So, some orders might not be completed and left in status K.

6.1.2 Order status flows

There are three possible order status flows:

Flow	Description
P → V → IV	This is normal status flow.
P → V → K → IV	This is flow when issues appear during data preparation, but later problem was fixed.
P → V → K	This is flow when issues appear during data preparation and problem was not fixed during DH retry policy time.

Order execution duration depends on multiply factors:

- Order type - different order types use different integration services some of them are faster some of them are slower.
- Order parameters - order parameters describe how much data will be generated. Bigger order periods and bigger object quantity will be generated longer.
- Order quantity in queue. If guaranteed supplier creates too many orders, they will be generated parallelly and will take more time to finish them all.
- Failures - Errors during order data preparation will trigger retry policy so order generation will take more time as usually. Sometimes it will be not generated at all.

6.1.3 Recommendations

1. For better performance "Request order" can be implemented as separate process which is able to create multiple orders.
2. For better performance "Get order details" can be implemented as separate process which is able to get details of multiple orders.
3. For better performance "Get order data" can be implemented as separate process which is able to get order data of multiple orders.
4. For better performance process parallelization could be used but with max 3 threads.
5. Any HTTP request which returns 5xx status can be retried.
6. Any HTTP request which returns 4xx status should stop process because where are business error and manual handling should be used. Except for the step "Get order data" and error "code": 2018, "text": "There is no data for the selected search parameters, the response is empty." It means that order data preparation is finished, and order is empty.
7. Step "Request order" and other steps should have separate retries. Get order data on failure should not trigger Request order one more time.
8. It's up to client to decide how long the "First wait" duration can be but it shouldn't be less than 1 second.
9. It's up to client to decide how long the "Repeating wait" duration can be but it shouldn't be less than 1 second.
10. Use fixed number of times for status check. After 25 hours DH order retry policy will stop working and order will be left in status K. So, it reasonable to have number of times equal ((25 hours) / ("Repeating wait" duration in hours)).
11. Do not recreate orders when orders got status K. Datahub retry policy will try to generate it later or DH team member interaction is needed to finish order. Client-side solutions will not help to solve status K.

6.2 JSON request logic

JSON field usage in requests by type:

Type	Example	Is value provided	Request result
Integer	orderId: null	No	All orders.
Integer	orderId: 4587125	Yes	Order with ID 4587125.
DateTime	submittedDateFrom: null	No	All orders.
DateTime	submittedDateFrom: ""	Yes	Framework validation error because provided value is not matching date format.
DateTime	submittedDateFrom: "2023-01-01"	Yes	All orders with were submitted date greater than 2023-01-01.
List	latestStatuses: null	No	All orders with all statuses.

Type	Example	Is value provided	Request result
List	latestStatuses: []	Yes	Empty list because provided latestStatuses list is not matching any latest status.
List	latestStatuses: [""] or latestStatuses: ["", ""]	Yes	Framework validation error because provided value is not matching statuses.
List	latestStatuses: [null] or latestStatuses: [null, null]	Yes	Empty list because provided latestStatuses list is not matching any latest status.
List	latestStatuses: ["IV"] or latestStatuses: ["IV", "V"]	Yes	Orders with statuses IV or V.
Boolean	auto: null	No	All orders because criteria were not given.
Boolean	auto: ""	No	Validation error because invalid Boolean value was given.
Boolean	auto: "NOT BOOLEAN"	No	Validation error because invalid Boolean value was given.
Boolean	auto: "false"	Yes	Orders which were ordered not automatically.

If field value is not provided, then field criteria shouldn’t be added to query and all lists should be returned.

7 DataHub Gatewy API documentation

7.1 Order controller

7.1.1 POST/gateway/guaranteed-supplier/order/list

URL	POST/gateway/guaranteed-supplier/order/list?first={integer}&count={integer}&sortKey={string}&sortOrder={ASC/DSC}
Description	Method will return list of the orders.
Parameter	**URL parameters:** • **first -** the index of the report line, which must be the first in the return list (starting from 0). Optional. The default value is 0. • **count** - the number of order’s rows in the return list. Optional. The default value is 30. If no count value is given, the default value count will be 30. • **sort** – ASC, DSC sorting: • By default, the orders list must be sorted by the orderId.
Header	After decrypting the guaranteed supplier authentication key, the guaranteed supplier ID is used to select the data.

JSON request	{
	"orderId": "integer",
	"orderTypes": [
	"string"
	],
	"submittedDateFrom": "dateTime",
	"submittedDateTo": "dateTime",
	"dateFrom": "date",
	"dateTo": "date",
	"latestStatuses": [
	"string"
	],
	"auto": "boolean",
	"userNameSearch": "string",
	"orderParametersSearch": "string"
	}

HTTP Response code	HTTP status code	Reason	Description
	200	OK	Request completed successfully.
	204	No Content	No data found according to the given parameters.
	400	Bad Request	Request error. The HTTP response body provides a list of errors in JSON format.
	401	Unauthorized	An attempt was made to connect to a non-public method that requires authentication, but no user credentials were provided.
	403	Forbidden	According to the access control policy, the current user does not have access to perform the requested action.
	404	Not Found	Either there is no API method associated with the request URL path, or the request contains one or more parameters that did not return the data.

JSON Response

```
[
    {
        "orderId": "integer",
        "orderType": "string",
        "submittedDate": "datetime",
        "dateFrom": "date",
        "dateTo": "date",
        "orderParameters": "string",
        "latestStatus": "string",
        "statusDate": "datetime",
        "expireDate": "datetime",
        "auto": "boolean",
        "userName": "string"
    }
]
```

JSON error response

```
{
    "errorMessages": [
        {
            "code": "integer",
            "text": "string"
        }
```

	] }				
Rules	**No.**	**Rule description**	**Error code**	**Error message**	**Attributes**
	1.	If an attribute has defined possible values, the value index can be specified by specifying the value of the attribute in the request. Indices of all possible values start from 0.	-	-	All attributes with specified values.
	2.	The date from cannot be later than the date to but can be equal.	1002	Date from cannot be later than date to.	dateFrom, dateTo, submittedDateFrom, submittedDateTo
	3.	Submitted date cannot be later than the current date but can be equal.	1010	Submitted date cannot be later than the current date.	SubmittedDateFrom, SubmittedDateTo

7.1.1.1 JSON Request structure

The table below describes the structure of the JSON request:

No.	Attribute	Type	Obligation	Description
1.	orderId	integer	not required	The report ordering primary surrogate key.
2.	orderTypes	string	not required	The short name of the order type. Possible meanings: • data-hr-15min-obj-lvl (Automated quantities at the object level).
3.	submittedDateFrom	datetime	not required	Order’s submission date from.
4.	submittedDateTo	datetime	not required	Order’s submission date to.

5.	dateFrom	date	not required	The beginning of the reporting period: • The format: YYYY-MM-DD.
6.	dateTo	date	not required	The end of the reporting period: • The format: YYYY-MM-DD.
7.	latestStatuses	[string]	not required	The status of the order. Possible meanings: • IV – Completed; • V – In progress; • P – Submitted; • K – Error. More than one type can be submitted.
8.	auto	boolean	not required	Indication that the order was ordered automatically.
9.	userNameSearch	string (240)	not required	The user who ordered the order.
10.	orderParametersSearch	string	not required	The order parameters.

7.1.1.2 JSON Response structure

The table below describes the structure of the JSON response:

No.	Attribute	Type	Obligation	Description
1.	orderId	integer	required	The report ordering primary surrogate key.
2.	orderType	string (100)	required	The short name of the order type. Possible meanings: • data-hr-15min-obj-lvl (Automated quantities at the object level).
3.	submittedDate	Datetime	required	The date of the order submission.

4.	dateFrom	Date	required	The beginning of the reporting period: • The format: YYYY-MM-DD.
5.	dateTo	Date	required	The end of the reporting period: • The format: YYYY-MM-DD.
6.	orderParameters	string (4000)	required	The search parameters by which the data in the ordered order was filtered.
7.	latestStatus	string (20)	required	The current status of the order.
8.	statusDate	dateTime	required	The latest status date.
9.	expireDate	dateTime	required	Date of validity of the order. • The ordered report with **status = Completed** by default, is available only for 24 hours.
10.	auto	boolean	required	Indication that the report order was ordered automatically.
11.	userName	string (240)	required	The user who ordered the order.

7.1.1.3 Error Response Structure

The following table describes the JSON structure in the event of a response error:

No.	Attribute	Type	Obligation	Description
1.	code	integer	required	Error code.
2.	text	string (4000)	required	Error message.

7.1.2 POST/gateway/guaranteed-supplier/order/data-hr-15min-obj-lvl

<table>
<tr><td><b>URL</b></td><td colspan="3">POST/gateway/guaranteed-supplier/order/data-hr-15min-obj-lvl</td></tr>
<tr><td><b>Description</b></td><td colspan="3">The method is designed for ordering data for “Automated quantities at the object level”. Using this method guaranteed supplier can only order the data of objects in guaranteed supply.</td></tr>
<tr><td><b>Parameter</b></td><td colspan="3"><b>URL parameters:</b>
The JSON data is contained in the HTTP request (BODY) (<i>see JSON structure, below</i>).</td></tr>
<tr><td><b>Header</b></td><td colspan="3">After decrypting the guaranteed supplier authentication key, the guaranteed supplier ID is used to select the data.</td></tr>
<tr><td><b>JSON request</b></td><td colspan="3">{
"dateFrom": "date",
"dateTo": "date",
"consumptionCategories": [
"string",
"string"
],
"objectNumbers": [
"string",
"string"
],
"interval": "string"
}</td></tr>
<tr><td rowspan="5"><b>HTTP Response code</b></td><td><b>HTTP status code</b></td><td><b>Reason</b></td><td><b>Description</b></td></tr>
<tr><td>201</td><td>Created</td><td>Request completed successfully.</td></tr>
<tr><td>400</td><td>Bad Request</td><td>Request error. The HTTP response body provides a list of errors in JSON format.</td></tr>
<tr><td>401</td><td>Unauthorized</td><td>An attempt was made to connect to a non-public method that requires authentication, but no user credentials were provided.</td></tr>
<tr><td>403</td><td>Forbidden</td><td>According to the access control policy, the current user does not have access to perform the requested action.</td></tr>
</table>

	404	Not Found	Either there is no API method associated with the request URL path, or the request contains one or more parameters that did not return the data.

JSON response

```
{
 "orderId": "integer"
}
```

JSON error response

```
{
   "errorMessages": [
      {
         "code": "integer",
         "text": "string"
      }
   ]
}
```

Rules	No.	Rule description	Error code	Error message	Attributes
	1.	If an attribute has defined possible values, the value index can be specified by specifying the value of the attribute in the request. Indices of all possible values start from 0.	-	-	All attributes with specified values.
	2.	The date from cannot be later than the date to but can be equal.	1002	Date from cannot be later than date to.	dateFrom, dateTo
	3.	The date from and date to cannot be later than the current date but can be equal.	1008	Date from and date to cannot be later than the current date.	dateFrom
	4.	Object meter must be automated.	2007	The submitted object number: **[objectNumber (if there is more than one object, objects must be separated by the semicolon)]**, was not found or the meter of object is not automated.	objectNumbers
	5.	Data cannot be older than 36 months old.	2012	Date from cannot be older than 36 months old.	dateFrom

	6.	Report can be ordered maximum for 12 months.	2013	The report can only be ordered for 12 months or less.	dateFrom, dateTo
	7.	A maximum of 500 objects can be submitted in a report order.	2021	A maximum of 500 objects can be submitted in a report order.	objectNumbers
	8.	If objectNumbers [null], then the report can be ordered for a maximum of 1 month period.	2023	The report without specifying the objects can only be ordered for 1 month or less.	objectNumbers, dateFrom, dateTo

7.1.2.1 JSON Request structure

The table below describes the structure of the JSON request:

No.	Attribute	Type	Obligation	Description
1.	dateFrom	date	required	The beginning of the reporting period: • The format: YYYY-MM-DD.
2.	dateTo	date	required	The end of the reporting period: • The format: YYYY-MM-DD.
3.	consumptionCategory	[string (20)]	required	The consumption category. Possible meanings: • P+ (active P+ electricity). • P- (active P- electricity). • Q+ (reactive Q+ electricity). • Q- (reactive Q+ electricity).
4.	objectNumbers	[string]	required	Object numbers.
5.	interval	string	required	Consumption interval. Possible meanings: • HOUR • QUARTER

7.1.2.2 JSON Response structure

The table below describes the structure of the JSON response:

No.	Attribute	Type	Obligation	Description
1.	orderId	integer	required	The report ordering primary surrogate key.

7.1.2.3 Error Response Structure

The following table describes the JSON structure in the event of a response error:

No.	Attribute	Type	Obligation	Description
1.	code	integer	required	Error code.
2.	text	string (4000)	required	Error message.

7.1.3 GET/gateway/guaranteed-supplier/order/{orderId}/count

URL	GET/gateway/guaranteed-supplier/order/{orderId}/count
Description	Method which will return count (number), how many items the guaranteed supplier will get in ordered report (reports could have more than 1 item, so it is List). It should be used when the guaranteed supplier needs to split data in few portions. This response should be used in reports' GET methods request, where guaranteed supplier can provide method parameters information.
Parameters	**URL parameters:** • **orderId** – order identification number. Required.
Header	After decrypting the guaranteed supplier authentication key, the guaranteed supplier ID is used to select the data.

<table>
<tr><td><b>JSON request</b></td><td colspan="5">GET request does not have the BODY part.</td></tr>
<tr><td><b>Response</b></td><td><b>HTTP status code</b></td><td><b>Reason</b></td><td colspan="3"><b>Description</b></td></tr>
<tr><td></td><td>200</td><td>OK</td><td colspan="3">Request completed successfully.</td></tr>
<tr><td></td><td>204</td><td>No content</td><td colspan="3">No data found according to the given parameters.</td></tr>
<tr><td></td><td>400</td><td>Bad Request</td><td colspan="3">Request error. The HTTP response body provides a list of errors in JSON format.</td></tr>
<tr><td></td><td>401</td><td>Unauthorized</td><td colspan="3">An attempt was made to connect to a non-public method that requires authentication, but no user credentials were provided.</td></tr>
<tr><td></td><td>403</td><td>Forbidden</td><td colspan="3">According to the access control policy, the current user does not have access to perform the requested action.</td></tr>
<tr><td></td><td>404</td><td>Not Found</td><td colspan="3">Either there is no API method associated with the request URL path, or the request contains one or more parameters that did not return the data.</td></tr>
<tr><td><b>JSON Response</b></td><td colspan="5">{
    "count": "integer"
}</td></tr>
<tr><td><b>JSON error response</b></td><td colspan="5">{
    "errorMessages": [
        {
            "code": "integer",
            "text": "string"
        }
    ]
}</td></tr>
<tr><td><b>Rules</b></td><td><b>No.</b></td><td><b>Rule description</b></td><td><b>Error code</b></td><td><b>Error message</b></td><td><b>Attributes</b></td></tr>
<tr><td></td><td>1.</td><td>Counts report list items and return SUM of all report lines.</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td></td><td>2.</td><td>The order status must be <b>Completed</b>.</td><td>2010</td><td>Invalid report order status.</td><td>orderId</td></tr>
</table>

	3.	Report order doesn't exist in the system.	2016	According to the submitted order number: **[orderId]**, the order does not exist.	orderId
	4.	Invalid method selected for report data or incorrect parameter.	2017	Invalid method selected or parameter specified incorrectly. According to the submitted order number: [**orderId**] report type is: [**orderType**].	orderId, orderType
	5.	No data found based on the search parameters submitted in the POST method.	2018	There is no data for the selected search parameters, the response is empty.	orderId

7.1.3.1 JSON Request structure

The table below describes the structure of the JSON request:

No.	Attribute	Type	Obligation	Description

7.1.3.2 JSON Response structure

The table below describes the structure of the JSON response:

No.	Attribute	Type	Obligation	Description
1.	count	integer	required	Number of rows, objects, accounts, depending on the selected report.

7.1.3.3 Error Response Structure

The following table describes the JSON structure in the event of a response error:

No.	Attribute	Type	Obligation	Description
1.	code	integer	required	Error code.
2.	text	string (4000)	required	Error message.

7.1.1 GET /gateway/guaranteed-supplier/order/{orderId}/data-hr-15min-obj-lvl

<table>
<tr><td><b>URL</b></td><td colspan="3">GET /gateway/guaranteed-supplier/order/{orderId}/data-hr-15min-obj-lvl?first={integer}&count={integer}</td></tr>
<tr><td><b>Description</b></td><td colspan="3">The method for receive the order report "Automated quantities at the object level".</td></tr>
<tr><td><b>Parameter</b></td><td colspan="3"><b>URL parameters:</b>
• <b>orderId</b> – order identification number.
• <b>first</b> - the index of the object, which must be the first in the return list (starting from 0). Optional. The default value is 0.
• <b>count</b> - the number of objects in the return list. Optional. The default value is 10000. If no count value is given, the default value count will be 10000.</td></tr>
<tr><td><b>Header</b></td><td colspan="3">After decrypting the guaranteed supplier authentication key, the guaranteed supplier ID is used to select the data.</td></tr>
<tr><td><b>JSON request</b></td><td colspan="3"></td></tr>
<tr><td rowspan="4"><b>HTTP Response code</b></td><td><b>HTTP status code</b></td><td><b>Reason</b></td><td><b>Description</b></td></tr>
<tr><td>200</td><td>OK</td><td>Request completed successfully.</td></tr>
<tr><td>204</td><td>No content</td><td>No data found according to the given parameters.</td></tr>
<tr><td>400</td><td>Bad Request</td><td>Request error. The HTTP response body provides a list of errors in JSON format.</td></tr>
</table>

	401	Unauthorized	An attempt was made to connect to a non-public method that requires authentication, but no user credentials were provided.
	403	Forbidden	According to the access control policy, the current user does not have access to perform the requested action.
	404	Not Found	Either there is no API method associated with the request URL path, or the request contains one or more parameters that did not return the data.

JSON response

```
{
   "personCode": "string",
   "personName": "string",
   "personSurname": "string",
   "objectBsId": "integer",
   "objectNumber": "string",
   "consumptionCategories": [
      {
         "consumptionCategory": "string",
         "consumptions": [
            {
               "consumptionTime": "datetime with timeZone",
               "amount": "integer",
               "valueType": "string"
            }
         ]
      }
   ]
}
```

JSON error response

```
{
   "errorMessages": [
      {
         "code": "integer",
         "text": "string"
      }
   ]
}
```

Rules	**No.**	**Rule description**	**Error code**	**Error message**	**Attributes**

	1.	The order status must be Completed**.**	2010	Invalid report order status.	orderId
	2.	According to the submitted order number: **[orderId]**, the order does not exist.	2016	Report order doesn't exist in the system.	orderId
	3.	Invalid method selected or parameter specified incorrectly. According to the submitted order number: [**orderId**] report type is: [**orderType**].	2017	Invalid method selected for report data or incorrect parameter.	orderId, orderType
	4.	No data found based on the search parameters submitted in the POST method.	2018	There is no data for the selected search parameters, the response is empty.	orderId
	5.	The number of objects in the return list must be less than or equal to 10000.	2022	The number of objects in the return list must be less than or equal to [**10000**].	count

7.1.1.1 JSON Request structure

The table below describes the structure of the JSON request:

No.	**Attribute**	**Type**	**Obligation**	**Description**

7.1.1.2 JSON Response structure

The table below describes the structure of the JSON response:

No.	**Attribute**	**Type**	**Obligation**	**Description**
1.	personCode	string (20)	required	Person / company code.

2.	personName	dtring (200)	required	Person / company name.
3.	personSurname	string (50)	required	Person surname.
4.	objectBsId	integer	required	Object Id.
5.	objectNumber	string (20)	required	Object number.
6.	consumptionCategories: []			
6.1	consumptionCategory	string (2)	required	Consumption category. Possible meanings: • P+ (active P+ electricity). • P- (active P- electricity). • Q+ (reactive Q+ electricity). • Q- (reactive Q+ electricity).
7.	consumptions: []			
7.1	consumptionTime	datetime with timeZone	required	Consumption time.
7.2	amount	number	required	Consumption amount in kWh/kVArh.
7.3	valueType	string (3)	required	Consumption value type. Possible meanings: • EST – estimated. • VAL – validated.

7.1.1.3 Error Response Structure

The following table describes the JSON structure in the event of a response error:

No.	Attribute	Type	Obligation	Description
1.	code	integer	required	Error code.
2.	text	string (4000)	required	Error message.