Contact : contact@thinksmartgrids.fr

Technical Documentation: API Supplier FlexReady requirements

To be certified as FlexReady, a supplier must provide an API that satisfies these requirements.

API Documentation - Supplier Signal

Version: 1.0

1. Introduction

This document describes the technical specifications required to develop an API compatible with our certification system. The goal is to enable a supplier to expose their power, cost, and CO_2 emission data for a given delivery point and over a defined time horizon.
The API must be a **REST API** and communicate exclusively in **JSON (application/json)**.
The approach is designed to be flexible: apart from these 5 aspects (described in detail below):

- adhering to the endpoint,
- adhering to authentication method,
- adhering to versioning,
- adhering to data format,
- basic exception handling,

the choice of technologies, authentication mechanisms, or other specific feature is left to the developer's discretion.

2. Authentication

Suppliers API must use **OAuth2** for authentication. OAuth2 is a standard authorization framework that allows secure access to resources via access tokens. These tokens are exchanged between the client and the server to validate and authorize API requests.
The access token must be included in the HTTP header of each request as follows:

```
Authorization: Bearer <access_token>
```

3. Versioning

To ensure backward compatibility and facilitate smooth updates, Suppliers API must implement versioning, following the FlexReady specifications version. The API version must be explicitly specified in the URL path, before the endpoint, as follows:

```
https://api.supplier.com/v1/endpoint
```

4. Endpoint

A single endpoint is required for the current version of the API.

Method: GET
URL: {base_url}/suppliers/commodity/prices

Description

This endpoint returns the supply signals (cost, power, CO_2) for a specific delivery point over a given time horizon, with a time step of one quarter-hour.

Query Parameters

Parameter Name	Type	Required	Description
delivery_point	string	Yes	The unique identifier of the delivery point.
start_date	string (ISO 8601)	Yes	The horodate of the beginning of the interval of commodity prices to request. Must be timezone-aware and not in past.
end_date	string (ISO 8601)	Yes	The horodate of the end of the interval of commodity prices to request. Must be timezone-aware, and posterior to start_date.

Example of a full request:

GET
https://api.supplier.com/v1/suppliers/commodity/prices?delivery_point=ABCDEFG1234567&start_date=2026-01-01T00:00:00Z&end_date=2026-01-01T01:00:00Z

NB: to specify time zone in a URL, if you want to precise a positive offset (eg "+02:00"), you might have to encode "+" as "%2B", because it is the way this symbol is encoded in URLs. Otherwise, it can be interpreted as a blank (" ") by a URL parser.

5. Response Structure

On a successful request (HTTP 200 OK status code), the response must be a JSON object that strictly adheres to the following structure.

Field Descriptions

- **file_generation_date** (string, ISO 8601 format, timezone-aware): Timestamp of the data file generation.
- **delivery_point** (string): Echo of the requested delivery point identifier.
- **start_date** (string, ISO 8601 format, timezone-aware): Timestamp of the beginning of the interval of commodity prices to request.
- **end_date** (string, ISO 8601 format, timezone-aware): Timestamp of the end of the interval of commodity prices to request.
- **supplier_signal** (JSON object): Object containing the various time-series data.
    - **step** (array of integers): Time step number (starting at 1) for each measurement. The step interval is 15 minutes.

- **horodate** (array of strings, ISO 8601 format, timezone-aware): Array of timestamps for each time step.
- **cost** (array of objects): Time-series data for cost.
- **power** (array of objects, Optional): Time-series data for power.
- **co2** (array of objects, Optional): Time-series data for CO_2 emissions

Format of Signal Objects (power, cost, co2)

Each data point in these arrays must be a JSON object with the following structure:

Key	Type	Description
value	float	The numerical value of the measurement.
unit	string	The unit of measurement (e.g., "W", "€/kWh", "gCO2eq/kWh").
multiplier	string	The power-of-10 multiplier (e.g., "k" for kilo, "M" for mega). Leave empty if no multiplier is used.

Example of a Complete JSON Response

```
{
   "file_generation_date": "2025-01-01T00:00:00+01:00",
   "delivery_point": "ABCDEFG1234567",
   "start_date" : "2026-01-01T00:00:00+01:00",
   "end_date" : "2026-01-01T00:45:00+01:00",
   "supplier_signal": {
      "step": [1, 2, 3, 4],
      "horodate": [
         "2026-01-01T00:00:00+01:00",
         "2026-01-01T00:15:00+01:00",
         "2026-01-01T00:30:00+01:00",
         "2026-01-01T00:45:00+01:00"
      ],
      "power": [
         {"value": 12, "unit": "W", "multiplier": "k"},
         {"value": 12.5, "unit": "W", "multiplier": "k"},
         {"value": 13, "unit": "W", "multiplier": "k"},
         {"value": 13.5, "unit": "W", "multiplier": "k"}
      ],
      "cost": [
         {"value": 1200, "unit": "€/kWh", "multiplier": ""},
         {"value": 120, "unit": "€/kWh", "multiplier": ""},
         {"value": 12, "unit": "€/kWh", "multiplier": ""},
         {"value": 1, "unit": "€/kWh", "multiplier": ""}
      ],
      "co2": [
         {"value": 3.1, "unit": "gCO2eq/kWh", "multiplier": ""},
         {"value": 3.14, "unit": "gCO2eq/kWh", "multiplier": ""},
         {"value": 3.141, "unit": "gCO2eq/kWh", "multiplier": ""},
         {"value": 3.1415, "unit": "gCO2eq/kWh", "multiplier": ""}
      ]
   }
}
```

Error Exceptions:

It is expected for the API to return correct responses code when the input query is incorrect. The supplier API is expected to handle at least these exceptions (the response codes will be checked during API validation):

Code	Type	Description
400	Bad request	Returned when the request is malformed or syntactically incorrect. This can happen if a required parameter like delivery_point is missing.
422	Unprocessable Entity	Returned when the server understands the request and its syntax is correct but cannot process it due to a semantic or business logic error. For example, if a delivery_point does not match business syntax rules, or if the start_date and end_date are in past or not in the correct order.