

# Peppered Events feed & Stories feed

This is the related documentation to the Events feed and Stories feed API's.

[Overview](#)

[How to make a request to the new events and stories API \(JSON\)](#)

[Endpoint](#)

[Language](#)

[Methods](#)

[Authorization](#)

[Body](#)

[Response](#)

[Image URL's](#)

[How to make a request to the XML/CSV Feeds](#)

[Endpoints](#)

[Methods](#)

[Authorization](#)

[Response](#)

[Facebook Feeds notes - Troubleshooting](#)

[Google Feeds notes - Troubleshooting](#)

[Usage examples](#)

[Other technicals details](#)

## Overview

Currently we offer a JSON API for Stories, Events, two XML feeds for facebook and a CSV file for Google.

JSON:

- <https://{site}/{language}/feed/events> (**new**)
- <https://{site}/{language}/feed/stories> (**new**)

XML:

- <https://{site}/feed/facebook>
- <https://{site}/feed/facebook/stream>

CSV:

- <https://{site}/feed/google>

The endpoint **stories** also returns all visible website pages. Use story types to narrow the results.

The endpoints are protected by rate limiting it (max. 5 request p/m) and by API Key which can be generated and maintained through the **Dashboard** (Dashboard > System > API Keys).

## How to make a request to the new events and stories API (JSON)

### Endpoint

<https://{site}/{language}/feed/events>

<https://{site}/{language}/feed/stories>

[https://{site}/{language}/feed/stories/\[storyID\]](https://{site}/{language}/feed/stories/[storyID])

*(By adding the storyID a detailed feed of all relevant content and media of that story is provided)*

**{language}** is **optional**, and must be two chars like: **nl**, **de**, **en**, **fr**, **gb**, etc (**ISO country codes two-letter code (alpha-2)**).

## Language

- Adding language to the url will change the return language for the whole feed.
- If no language is provided the default one will be used, which generally is dutch.
- If an nonexistent language is provided, a 404 page will be shown.

## Methods

POST

GET

## Authorization

Must be in the header or as a GET parameter

Key: apikey

Value: {generated\_dashboard\_key}

## Body

Must be form-data but they can also be passed as GET parameters

startDate: (optional, YYYY-MM-DD):

endDate: (optional, YYYY-MM-DD):

productionCode (optional string): **the production code** or **empty**

type (optional string): **the story type**

## Response

### SUCCESS

Content-Type: application/json

JSON containing a list of events or stories

### FAILURE

#### Response on Wrong/Empty/Invalid API Key

```
{
  http_code: 403,
  status: "Forbidden",
}
```

#### Response when parameter past is provided but not startDate

```
{
  http_code: 400,
  status: "When parameter 'past' is provided, the parameter 'startDate' must also"
}
```

#### Any other error:

```
{
  http_code: 500,
  status: "{error_message}",
}
```

## Image URL's

In the request you set (via the headers) if you prefer JPG or WEBP. Your browser will probably set WEBP. Using it as a developer you should be able to set it in your code.

If the `Accepts:` header does *not* specifically mention WEBP, the proxy will serve a JPG by default.

## How to make a request to the XML/CSV Feeds

### Endpoints

Endpoint	Authorization	Method	Rate Limiting
https://{site}/feed/facebook	Yes	POST / GET	Yes (5req/60sec)
https://{site}/feed/facebook/stream	Yes	POST / GET	Yes (5req/60sec)
https://{site}/feed/google	Yes	POST / GET	Yes (5req/60sec)

### Methods

POST

GET

### Authorization

Must be in the header or as a GET parameter

Key: apikey

Value: {generated\_dashboard\_key}

### Response

#### SUCCESS

Content-Type: application/xml

XML containing a list of events

#### FAILURE

##### Response on Wrong/Empty/Invalid API Key

```
<?xml version="1.0"?>
<result>
  <http_code>403</http_code>
  <status>Forbidden</status>
</result>
```

##### Any other error:

```
<?xml version="1.0"?>
<result>
  <http_code>500</http_code>
  <status>{error_message}</status>
</result>
```

## Facebook Feeds notes - Troubleshooting

- Facebook feeds can become a headache to work with, that's why I'd advice:
  - Get access to Peppered Facebook Account
  - Go to the Facebook Commerce page located here <https://business.facebook.com/commerce/>
  - On that page a blue button says "Add Catalog", there pick either
    - **E-Commerce** > Products which uses the template facebook\_feed.xml.twig
    - **Entertainment and media** > Media titles which uses the template facebook\_stream\_feed.xml.twig
  - Whichever your choice, just click next until you get to Overview
  - Top right, there's a gray button to add articles, choose the option to upload (data overview).
  - From the two radio buttons choose "No, I need an overview template", and then just follow the steps until you get an XML or CSV.
  - I'd expect that, if at some points our clients uploads fail, then Facebook would update these so we can quickly fix the inconsistency or missing field, or whatever they changed.
- Related page <https://www.facebook.com/business/help/1898524300466211?id=725943027795860>

## Google Feeds notes - Troubleshooting

- Further information can be found here over fields and their length, and other details [https://docs.google.com/spreadsheets/d/1liGxkrvxhWrBTOe0buMXNjdtPHAmI3ztUAU8\\_AyecRU/edit#gid=0](https://docs.google.com/spreadsheets/d/1liGxkrvxhWrBTOe0buMXNjdtPHAmI3ztUAU8_AyecRU/edit#gid=0)

## Usage examples

- <https://www.theater-haarlem.nl/feed/facebook/stream/?apikey={insert your key here without the brackets}>
- <https://www.wilminktheater.nl/feed/facebook/?apikey={insert your key here without the brackets}>

## Other technicals details

- The feeds eventbase, peppered and iamsterdam are end of life on 31 December 2023
- Agendabe has a completely different approach on how to get events.-
- Agendabe does not support filters at the moment.-
- We should aim to use on a future the **EventsFeedManager** and **ProductionsFeedManager classes** for all feeds including **agendabe**.
- **Diagram:** <https://app.diagrams.net/#G1PdGemTqvKw0IntvrewfRPpt6MER4NwcY>