

# Stock Affiliate API workflow

The purpose of this document is to illustrate the overall process and technical workflow for Adobe Stock partners who want to integrate the Adobe Stock Search API into their applications. The Stock API allows you to interact with Adobe Stock programmatically, rather than through a user interface.

## Document contents

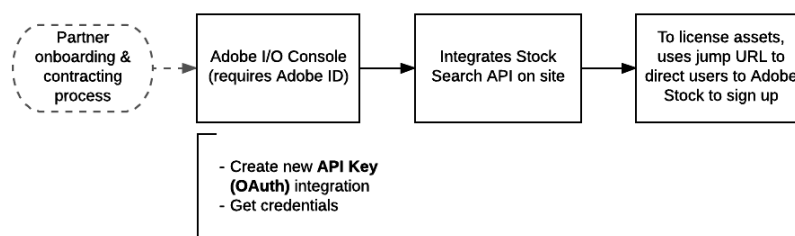
Introduction .....	1
Workflow overview .....	1
Adobe I/O Console workflow .....	2
Creating a Search API application .....	3
Authenticating requests .....	3
Application flow details .....	4

## Introduction

This paper outlines the workflow for Adobe Stock affiliate partners. This could include:

- **Stock aggregator sites.** Aggregators have websites that may include stock content from multiple providers, including Adobe Stock, as well as their own collection of assets. Adobe Stock search could be added to the site, with links referring users back to the Adobe Stock website for completing the sale. The aggregator could receive a commission fee if it leads to a new subscription.
- **Independent developers.** The affiliate workflow is also suitable for developers who want to get familiar with the Adobe Stock APIs, and/or want to consume them within the context of a test application. In this case, no commissions on sales would occur (it would not be expected that the test application would create any sales).

## Workflow overview



- Assuming that the partner wants to receive commissions, the process begins with a contracting process with the Adobe Stock Business Development team. During this process, the partner will enroll in the Adobe Affiliate program, and accept the terms of use for the Stock API.
  - Contracting and commissions are beyond the scope of this document. To get more information, please contact Grp-AdobeStockPartnerships@adobe.com.
- Whether the partner goes the commission route or simply wants to test the APIs, the rest of the process is the same. Developers will register their application using the Adobe I/O Console, which will allow it access Stock APIs. The result of this process is an API key which is used to allow requests.
- Now the integrator can start writing an application that uses the Stock API to make search queries and get back image thumbnail results, which the application will display.
- When the website end-user wishes to license an asset, the API provides a "jump" URL which links directly to the details page on the Stock site where the user can sign up for a plan.
  - If the partner has signed up as an affiliate, the transaction will be tracked by Adobe and referral credit given back to the partner.

## Adobe I/O Console workflow

To use the Adobe Stock API, you need to generate an application key on the Adobe I/O Console. Adobe I/O will whitelist this key and permit your application access to the APIs.

- Access the Adobe I/O Console here: <https://console.adobe.io>.
  - If you do not already have an Adobe ID, you will need to create one (for free).
- Click the **New Integration** button.
- Select the following items, clicking **Continue** each time:
  - **Access an API > Adobe Stock > OAuth<sup>1</sup> integration > New integration**
- This opens a screen where you will enter your integration details:
  - **Name:** Your application's name. This will not be sent in your API requests; however, a good practice might be to give it the same web-friendly name you will be using later when sending the required X-Product header (see *Application flow details*, below).
  - **Description:** E.g., "Integration of Stock API with MyWebsite.com."
  - **Platform:** Choose Web. This document assumes you are authoring a web/browser-based application as opposed to a native iOS or Android application.
  - **Default redirect URI:** As mentioned in the footnote above, this field is only required for true OAuth integrations. Since this is actually an API Key integration, this field is not required. However, you must still fill it out with something.

---

<sup>1</sup> The label "OAuth" is a misnomer. This is not a true OAuth integration, but an API Key integration. Adobe Stock supports three integration types: Service Account, (true) OAuth, and API Key. The Adobe I/O Console currently only shows two of those options, and the API Key integration is incorrectly labeled as "OAuth," which is a bug.

- You can use any arbitrary URL address (e.g, `https://mysite.com`.)
- **Redirect URI pattern:** Same as above; this field is not required for this integration method but you must still put some information here. You may enter the same URL as above, but you must escape periods (.) with `\\`.
  - Ex: `https://mysite\\.com`
- Once saved, the I/O Console will generate a few pieces of data.
  - The *only* information you need from this screen is the **API Key (Client ID)**. You do not need the **Client Secret** for this workflow. Copy this somewhere for the next step. This whitelists your identifies your application, and is required for receiving your commission (if you went through the partner contract process).

## Creating a Search API application

Whether you are creating a standalone web application or integrating Stock Search into your existing website or application, the process is the same. For the purposes of this document, the term “application” will describe the integration you are creating. The application will make and receive RESTful calls between the end-user and Adobe Stock, parse the JSON response, and update the UI to show the search results.

### Authenticating requests

The Stock Search API only requires API key authentication; otherwise, requests are essentially anonymous. Consequently, this integration type cannot be used by itself for API methods that require user- or organization-level authentication. For example, the API key alone cannot be used to license images or get a past license history.

### *Stock headers and example*

With each HTTP request, you will set mandatory and optional headers in addition to any search query parameters.

- Required:
  - `x-api-key`: The API key assigned to your application integration when you signed up through the Adobe I/O Console, above. It is a GUID-style string, generated programmatically.
  - `X-Product`: The name of your product or website that is using the API. This can be whatever you want, however best practice is that you use something a title similar to the integration name you created on the Adobe I/O Console.
- Optional:
  - `Authorization`: Access token issued by Adobe to an end user or organization. If included in search requests, this will indicate whether assets are already licensed or not.

- As mentioned above, this requires a different kind of workflow. The Affiliate workflow does not provide authorization, and it is implied that integrations will be making anonymous requests only.
- **X-Request-Id**: A unique request identifier that you define, used to trace the request in logs. The logs are not accessible to partners, but Adobe may request that you add this header to help troubleshoot issues.

Most search queries are issued with an HTTP “GET” request, however an HTTP “POST” is required if you want to upload an image for similarity search. Here are examples of using required headers.

#### *Sample request with headers*

```
GET /Rest/Media/1/Search/Files?locale=en_US
&search_parameters[words]=flamingos HTTP/1.1
Host: stock.adobe.io
X-Product: CFSTest/1.0
X-API-Key: ...0e3f
```

## Application flow details

It is assumed the application will provide a UI for searching and displaying results, since the Stock APIs do not provide this.

1. The workflow begins when the partner’s website customer, the *end-user*, performs a search on the partner’s site for Adobe Stock images. Search parameters indicate what kind of search to perform (for example a word search, search by ID, search by artist, or a visual similarity search), as well as what filters to apply (search only on photos, vectors, videos, templates, etc.)
2. The partner’s web application submits a **Search/Files** request to the Stock API.

#### *Search/Files request<sup>1</sup> (using cURL syntax)*

```
curl -X GET \
  'https://stock.adobe.io/Rest/Media/1/Search/Files?locale=en-US&search_parameters%5Bwords%5D=kittens' \
  -H 'x-api-key: ...0e3f' \
  -H 'x-product: TeamCFS_1.0'
```

- The request supplies any number of search parameters, filters and response fields.
3. The Stock API responds with a JSON object containing the number of results (**nb\_results**), and a **files** array of metadata for each asset found.

---

<sup>1</sup> For a complete reference, see <https://www.adobe.io/apis/creativecloud/stock/docs/api/search.html>.

*Search/Files response*

```
{
  "nb_results": 247038,
  "files": [
    {
      "id": 75950374,
      "title": "five kittens",
      "width": 2500,
      "height": 1667,
      "creator_name": "adyafoto",
      "creator_id": 205216144,
      "thumbnail_url":
        "https://as2.ftcdn.net/jpg/00/75/95/03/500_F_75950374_yNANaKsbx7oLzG
        JUFsZxW7j5cGoiKDT9.jpg",
      "thumbnail_html_tag": "<img
        src=\"https://as2.ftcdn.net/jpg/00/75/95/03/500_F_75950374_yNANaKsbx
        7oLzGJUFsZxW7j5cGoiKDT9.jpg\" alt=\"five kittens\" title=\"Photo:
        five kittens\" zoom_ratio=\"1.25\" zoom_depth_max=\"2\" />",
      "thumbnail_width": 500,
      "thumbnail_height": 334,
      "media_type_id": 1,
      "vector_type": null,
      "content_type": "image/jpeg",
      "category": {
        "id": 44,
        "name": "Cats"
      },
      "premium_level_id": 0
    }, ...
  ]
}
```

- Response is truncated to show one file result. The API allows full control over what fields are returned in the search results.
4. The partner's application parses the results and shows thumbnails for each asset and any other data (such as keywords) that may help the user refine their search. The end-user may use watermarked versions of images until they are ready to license.
  5. If the partner wants to allow the user to see more details about an asset, the application can submit additional queries to get an array of similar images or more images within that same series.

*Search for similar image*

```
curl -X GET \
  'https://stock.adobe.io/Rest/Media/1/Search/Files?locale=en-
  US&search_parameters%5Bs%5Bsimilar%5D=%2092901240&search_parameters%5Bs%5B
  milar_image%5D=1&search_parameters%5Blimit%5D=5' \
```

```
-H 'x-api-key: ...0e3f' \
-H 'x-product: TeamCFS_1.0'
```

6. If the user wants to license the image, you can give them a link to the detail page of that asset on the Adobe Stock website, via a “jump” URL provided by the API. You get this URL in your response data by adding the `result_columns[]=details_url` parameter.
  - Note that adding *any* `result_columns` parameter to the query string will cause only those fields to appear in the JSON results. You can add multiple `result_columns` commands to get exactly the results you want.

*Filter search for details\_url and title only for ID*

```
curl -X GET \
  'https://stock.adobe.io/Rest/Media/1/Search/Files?locale=en-US&search_parameters%5Bmedia_id%5D=75950374&result_columns%5B%5D=title&result_columns%5B%5D=details_url' \
  -H 'x-api-key: ...0e3f' \
  -H 'x-product: TeamCFS_1.0'
```

- In the jump URL returned, the `as_channel=affiliate` and other parameters allows Adobe tracking so that your application gets referral credit. Be sure not to remove them.

*Filtered results*

```
{
  "files": [
    {
      "title": "five kittens",
      "details_url":
      "https://stock.adobe.com/75950374?as_channel=affiliate&as_source=api&as_content=8af2713e4ace4087889ddfd04a94b6ac"
    }
  ]
}
```

