

Incremental Feed API Reference

Sitecore Discover

v 1.0.8 20210621

Table of Contents

| | | |
|-----------|---|----|
| Chapter 1 | Introduction | 3 |
| Chapter 2 | Basics | 4 |
| Chapter 3 | API URL | 5 |
| Chapter 4 | API Request: POST method | 6 |
| 4.1 | Example | 7 |
| Chapter 5 | Data Object | 8 |
| 5.1 | Product..... | 9 |
| 5.2 | Store | 10 |
| 5.3 | Restriction / Availability..... | 11 |
| 5.4 | Locale | 12 |
| 5.5 | Attributes | 13 |
| Chapter 6 | Response Status and Error Messages | 14 |
| 6.1 | Response | 15 |
| 6.2 | Error Object | 16 |
| 6.3 | Error Codes..... | 17 |
| Chapter 7 | Examples | 18 |
| 7.1 | API request using Curl..... | 19 |
| 7.2 | On success – Update price on sku “123” | 20 |
| 7.3 | On success – Update stock_quantity of sku “123” for store “STORE001” | 21 |
| 7.4 | On Error: “sku” not provided and price is sku specific | 22 |
| 7.5 | On Error: no incremental fields provided | 23 |
| Chapter 8 | Disabling Active Skus | 24 |
| 8.1 | Example: Disable sku “123” using stock_quantity | 25 |
| 8.2 | Example: Disable sku “123” only for store “STORE001” using stock_quantity | 26 |
| 8.3 | Example: Disable sku “123” using stock_status attribute..... | 27 |

Sitecore® is a registered trademark. All other brand and product names are the property of their respective holders. The contents of this document are the property of Sitecore. Copyright © 2001-2022 Sitecore. All rights reserved.

Chapter 1 Introduction

This document describes the Sitecore Discover API that you use to send incremental data feed updates to Sitecore Discover.

You must provision your domain in the Sitecore Discover system in order to receive incremental feed updates. The attributes that you can update depend on how Sitecore Discover is provisioned.

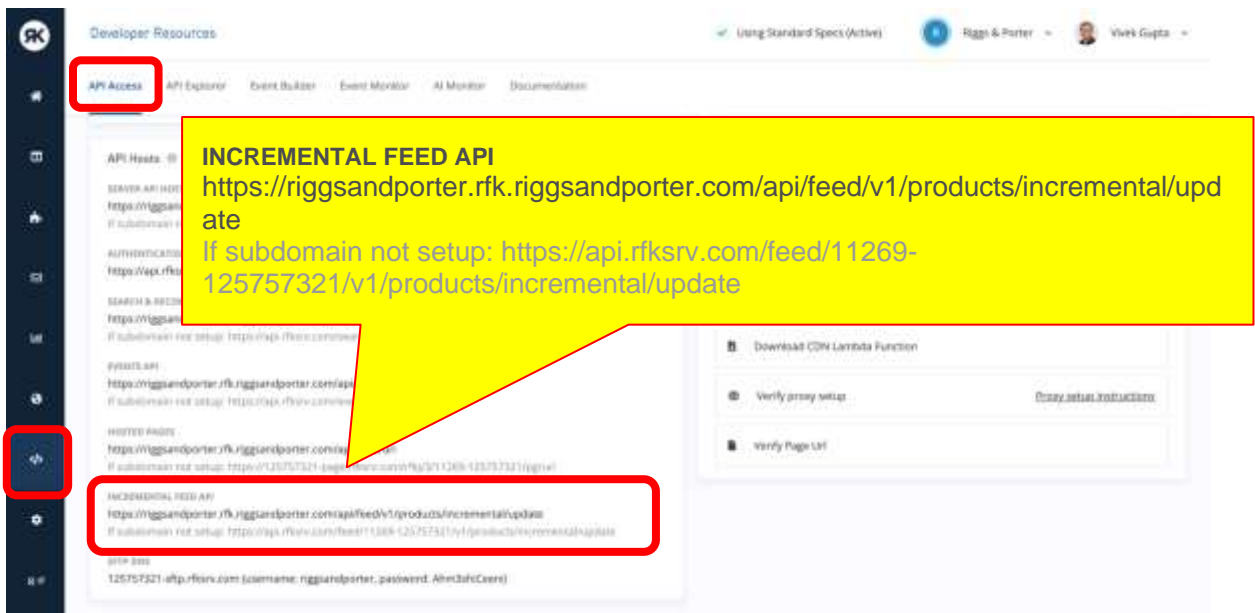
Chapter 2 Basics

Sitecore Discover supports both the HTTP and HTTPS protocols for the APIs.

Sitecore Discover APIs accept requests in the form of JSON objects. All information of inquiry to the APIs must be in a key data. The value of the key can either be a JSON string or a URL encoded JSON string.

Chapter 3 API URL

You can obtain the API URL in the Sitecore Discover Portal, in the **Developer Resources -> API Access** section.



Chapter 4 API Request: POST method

When calling from the server, use the API key. You can use the API key directly without an access token to make API calls.

Set the header as follows:

```
Host: <INCREMENTAL_API_URL>  
Header: Authorization: <api-key>  
Header: Content-Type: application/json
```

api-key can be assigned different scopes. A scope to call /data and to call /account/<version>/access-token.

The incremental API modifies data in the Sitecore Discover system that will surface on your website.

Note:

You should NOT call this API from the browser.

Required keys:

| Key | Data type | Description |
|---------|-------------|---|
| api-key | string | An API key provided by Sitecore Discover. This can be used to make calls from the server. It should not be used to call from the browser. |
| data | json string | JSON that represents the request. |

4.1 Example

```
POST <INC_API_URL>/feed/products/incremental/update
{
  "data": "{<your_key_value_data>}"
}

Header:
Content-Type: application/json
Authorization: "<your api-key>"
OR
Authorization: Bearer <your access-token>
```

To obtain an access token:

```
POST api.rfksrv.com/account/1/access-token
{
  "data": '{"scope":["data"]}'
}

Header:
Content-Type: application/json
x-api-key: "<your api-key>"

Curl example:

curl -X POST https://api.rfksrv.com/account/1/access-token \
-H 'Content-Type: application/json' \
-H 'x-api-key: <api-key>' \
-d '{"scope":["data"]}'
```

Chapter 5 Data Object

This section describes the format and elements of the data object in API request.

Format:

```
{
  "data": {
    "products": [
      {
        "sku": "<sku number>",
        "store": {
          "id": <store id>,
          "group_id": <store group id>
        },
        "locale_country": "<2 letter country code, eg: us>",
        "locale_language": "<2 letter language code, eg: en>",
        "restriction": {
          "isAvailable": true|false,
        }
        "attributes": {
          "<attribute-name-to-update>": <value>,
          ...
        }
      },
      { ... }
    ]
  }
}
```

The following table lists supported keys:

| Key | Value Type | Description |
|----------|------------------|--|
| products | array of objects | List of Product be updated. You may send upto 100 updates in one request. |

5.1 Product

The following table lists supported keys:

| Key | Value Type | Description |
|-------------|------------|--|
| sku | String | <p>The value identifying the sku to be updated.</p> <p>Note: If an attribute to be updated is not sku specific, you can specify any sku that belongs to the product.</p> <p>Required.</p> |
| store | Object | <p>Store object.</p> <p>Required if your site supports stores or is B2B site supporting customer/customer groups, and you are updating store specific attribute.</p> |
| restriction | Object | <p>Update the restriction of a given product or sku for a particular customer group/customer if you are setup as a B2B account.</p> <p>Note: In general, store inventory is managed through stock_quantity. However, there are times when availability of products/stores might also be set up as a restriction within Sitecore Discover. If so, you can use this to update availability of products/skus for particular stores.</p> |
| attributes | Object | <p>Feed Attributes object.</p> <p>Required.</p> |

5.2 Store

Type: Object

The following table lists supported keys:

| Key | Value Type | Description |
|----------|------------|---|
| id | String | <p>Store id. This id should be consistent with the id shared as part of product feed.</p> <p>B2B websites use this ID to identify customer.</p> <p>Note: The id must have been present in the batch feed. If not present, the behavior is not defined.</p> |
| group_id | String | <p>group_id. If your stores are organized in two level hierarchy (for example as zones and stores), you can pass the top level organization identifier as group_id. This id should be consistent with group_id shared as part of product feed.</p> <p>For B2B websites, if your customers are organized in two-levels, then you can pass the top level organization identifier as group_id.</p> <p>Note: The group_id must have been present in the batch feed. If not present, the behavior is not defined.</p> |

5.3 Restriction / Availability

Availability of products or skus is applicable only if the Sitecore Discover system is configured for availability. Availability is a feature of B2B whereby a particular product or sku might be available for a particular customer group or customer depending on the setup.

In certain cases, product/sku availability in stores might also be set up as a restriction within Discover. If so, you can use this to update availability of products/skus for particular stores.

You provide availability for each of the product/sku in the feed. See Sitecore Discover Data Feed Specifications (available on [Sitecore Developer Portal](#)) for more information. During the course of the day, between full batch feed processing, when the products/skus are sold out in different stores or is replenished, you can choose to keep the Discover system up-to-date on the availability of products/skus via restriction interface.

| Key | Value Type | Description |
|-------------|------------|--|
| isAvailable | Boolean | <p>True if the product/sku is available for the given customer or store.</p> <p>False if the product/sku is no more available for the given customer or store.</p> <p>Example:</p> <pre>"restriction": { "isAvailable": true false }</pre> |

5.4 Locale

If your domain is set up for multi-locale, then you must specify the locale to update the attributes for. You may choose to specify only the language. If so, all countries configured with the specified language will be updated with the given values.

| Key | Value Type | Description |
|-----------------|------------|--|
| locale_language | String | Required for multi-locale setup. Two letter ISO 639-1 code representing the language. Example: <code>"locale_language": "es"</code> |
| locale_country | String | Optional. Two letter ISO 3166-1 alpha-2 code representing the country. If specified, the specific locale (language+country) will be updated. If not specified, the attribute values are updated for all countries that are configured for the given language. Example: <code>"locale_country": "us"</code> |

5.5 Attributes

Type: Map of key: value

The following table lists supported keys:

| Key | Value Type | Description |
|----------------|------------|--|
| "key": <value> | Various | <p>One or more field/value entries to be updated.</p> <p>Note: you can specify only the attributes that have been approved by Sitecore Discover. Passing any other attribute will result in your request being rejected.</p> <p>Examples:</p> <pre>"price": 23.45 "stock_quantity": 30</pre> |

Chapter 6 Response Status and Error Messages

This section lists and describes the response status and error messages that Sitecore Discover can return.

6.1 Response

| Http Status | Response Object | Description |
|-------------|------------------------------|---|
| 200 | Empty | Returned upon success. |
| 400 | Error Object | Bad Request. Request can be retried after fixing the error. |
| 500 | - | Internal Server Error. Request may be retried again. |

6.2 Error Object

Format:

```
{
  "message": "<descriptive message>",
  "code": "one of the enumerated codes. see Error Codes",
  "type" : "HTTP Failure status code",
  "data": {
    <key: value>
  }

  // additional fields that may change over time
  // for internal use only.

  "request-id" : "<unique id of the request>",
  "error-id": "<error id>",
}
```

Example:

```
Example:
{
  "message": "Invalid value type for provided 'attribute'",
  "code": "INVALID_ATTRIBUTE_TYPE",
  "type" : "BadRequest",
  "data": {
    "attribute": "final price",
    "expected_type": "float",
    "product_index": 1,
    "value": false
  }
  "request-id" : "0715ad4e-b68f-4a32-8002-d34d77fc292a",
  "error-id": "ec6a5a22-0302-4e91-b086-413e14f3efe0",
}
```

| Key | Value Type | Description |
|---------|------------|---|
| code | String | One of the enumerated error code values. |
| message | String | Descriptive message explaining the error. |
| data | Object | Detailed information specific to the error that is being reported to help understand the exact error. The object attributes depends on the type of error. |

6.3 Error Codes

The following table lists and describes error codes that a response can return.

| Error Code | Description |
|-------------------------------------|--|
| EMPTY_PRODUCTS | No products provided in the body. |
| EMPTY_SKU | No sku provided in product. |
| NO_INC_FIELDS | No incremental fields provided in product. |
| EMPTY_STORE_ID | Store-specific field is updated but no store id is provided. |
| EMPTY_STORE_GROUP_ID | Store id is provided but no store group id is provided. |
| NO_INC_FIELDS_FOR_STORE | No incremental fields provided in 'product' for the given store. |
| INVALID_DOMAIN | Provided domain is not valid. |
| INCREMENTAL_FEED_NOT_ENABLED | Incremental feed is not enabled for domain. |
| DOMAIN_NOT_ACTIVE | Provided domain is not active. |
| ATTRIBUTE_NOT_FOUND | Provided attribute is not configured in the Portal. |
| INVALID_ATTRIBUTE_TYPE | Invalid type for provided attribute value. |
| ATTRIBUTE_NOT_MARKED_AS_INCREMENTAL | Provided attribute is not marked as incremental in the Sitecore Discover Portal. |
| EMPTY_LOCALE_LANGUAGE | locale_language is required when passing locale_country. |
| NOT_SUPPORTED_LOCALE | Provided locale is not supported by the domain. |

Chapter 7 Examples

This section provides examples of API calls and responses.

7.1 API request using Curl

The following is an example of what every Curl calls to the API must include:

```
curl -X POST \  
  <INC_API_URL>/feed/products/incremental/update \  
  -H 'Accept: */*' \  
  -H 'Accept-Encoding: gzip, deflate' \  
  -H 'Authorization: 01-16d038d8-1f12d3e2b2...9b05467c7011af308' \  
  -H 'Content-Type: application/json' \  
  -d '<data-object>'
```

7.2 On success – Update price on sku “123”

The following shows an example the Data Object in a successful request to update price of sku “123”.

```
Data Object

{
  "data": {
    "products": [
      {
        "sku": "123",
        "attributes": {
          "price": 10.0,
        }
      },
      { ... }
    ]
  }
}
```

Returns 200 OK

7.3 On success – Update stock_quantity of sku “123” for store “STORE001”

The following shows an example the Data Object in a successful request to update a stock quantity of sku for a specific store ID.

```
Data Object

{
  "data": {
    "products": [
      {
        "sku": "123",
        "store": {
          "id": "STORE001"
        },
        "attributes": {
          "stock_quantity": 3
        }
      },
      { ... }
    ]
  }
}
```

Returns 200 OK

7.4 On Error: “sku” not provided and price is sku specific

The following shows an example a Data Object where the required sku is omitted and the response returns an error 400:

Data Object

```
{
  "data": {
    "products": [
      {
        "attributes": {
          "price": 10.0,
        }
      }
    ]
  }
}
```

Returns 400 with message:

```
{
  "code": "EMPTY SKU",
  "message": "sku not provided"
}
```

7.5 On Error: no incremental fields provided

The following shows an example a Data Object where the required incremental field is omitted and the response returns an error 400.

```
Data Object

{
  "data": {
    "products": [
      {
        "sku": "sku123",
      }
    ]
  }
}
```

```
Returns 400 with message:

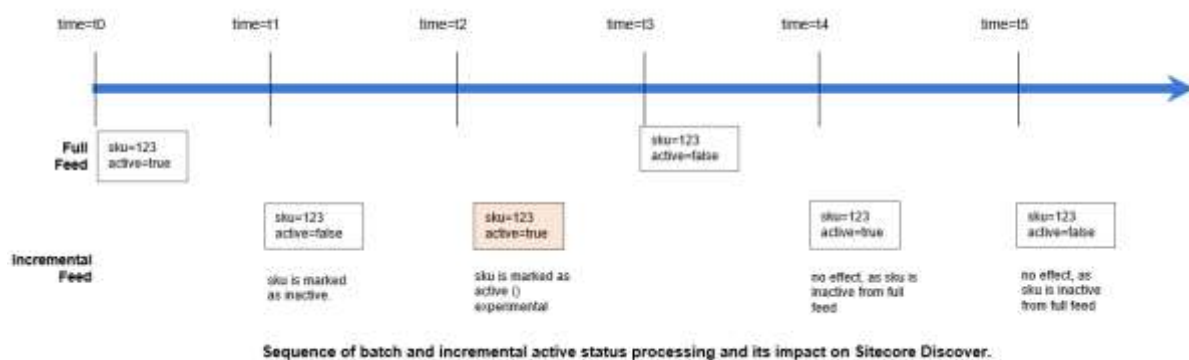
{
  "code": "NO_INC_FIELDS",
  "message": "Zero incremental fields provided"
}
```

Chapter 8 Disabling Active Skus

It is possible to disable an active sku and remove it from Sitecore Discover system using this API. You achieve this by passing the attributes that affect the availability of the sku as part of the Data. You can only disable skus that were previously active. To re-enable the sku, you must provide the active status in the full product feed.

Note:

The system might allow you to enable a sku that was disabled using this API previously before batch feed is processed again. However, this is an experimental feature and it is not officially supported.



8.1 Example: Disable sku "123" using stock_quantity

If the Sitecore Discover system is configured to mark the sku as disabled if the stock quantity is zero, you can disable the sku when you pass the attribute `stock_quantity` with value zero.

```
Data Object

{
  "data": {
    "products": [
      {
        "sku": "123",
        "attributes": {
          "stock_quantity": 0,
        }
      },
      { ... }
    ]
  }
}
```

Returns 200 OK

8.2 Example: Disable sku "123" only for store "STORE001" using stock_quantity

Let us say that Sitecore Discover is configured to mark the sku as disabled if the stock quantity is zero. You can disable the sku for the given store by passing the attribute `stock_quantity` with value zero and passing the store id, as show in the following example:

```
Data Object

{
  "data": {
    "products": [
      {
        "sku": "123",
        "store": {
          "id": "STORE001"
        },
        "attributes": {
          "stock_quantity": 0,
        }
      },
      { ... }
    ]
  }
}
```

Returns 200 OK

8.3 Example: Disable sku "123" using stock_status attribute

If Sitecore Discover is configured to mark the sku as disabled if the stock status is RED, you can disable the sku when you pass the attribute `stock_status` with value RED.

```
Data Object

{
  "data": {
    "products": [
      {
        "sku": "123",
        "attributes": {
          "stock_status": "RED",
        }
      },
      { ... }
    ]
  }
}
```

Returns 200 OK