OAR API Documentation - Extended Profile

Last updated
Mar 20, 2022

API Endpoints
- GET /api/facility-processing-types
- GET /api/workers-ranges
- GET /api/product-types
- GET /api/facilities
  - New Details Query Parameter and Default Behavior
  - New Extended Profile Query Parameters for Filtering
    - Number of Workers
    - Parent Company
    - Product Type
    - Facility Type
    - Processing Type
- POST /api/facilities
  - Example CURL Request
  - Sample Data Submission
  - Sample Potential Match Response
  - Sample Automatic Match Response
- GET /api/facilities/{id}
  - Example CURL Request
  - Sample Response

Interpreting Extended Field Values
- Shared Schema
- Field Value Schemas
  - Number of Workers
  - Parent Company
  - Product Type
  - Facility Type and Processing Type
API Endpoints

GET /api/facility-processing-types
Returns an object where the keys are valid facility types and the values are valid processing types.

GET /api/workers-ranges
Returns a list of valid choices for filtering facilities by the number of workers.

GET /api/product-types
Returns a list of suggested choices for filtering facilities by product type.

GET /api/facilities
Returns a list of facilities in GeoJSON format for a given query. (Maximum of 50 facilities per page.)

New Details Query Parameter and Default Behavior
To increase search performance the results returned from this API will now exclude contributor details and extended fields by default. To include these fields in your results at the expense of slower response time, include detail=true as a query string parameter.

New Extended Profile Query Parameters for Filtering
Number of Workers

number_of_workers=Less%20than%201000
The facility must have a number of workers within the specified range.

This value may be set to one of the values returned from the GET /api/workers-ranges endpoint.
Parent Company

parent_company=abc%20international
The facility must have a matching parent company name submitted by any contributor.

parent_company=123
The facility must have a matching parent company ID submitted by any contributor.

parent_company=abc%20international&parent_company=123
The parameter may be repeated to search for multiple parent companies. This will match facilities with either parent company. Name and ID matches can be mixed.

Product Type

product_type=hats
The facility must have a matching product type submitted by any contributor.

product_type=hats&product_type=gloves
The parameter may be repeated to search for multiple product types. This will match facilities with either product type.

Facility Type

facility_type=raw%20material%20processing%20or%20production
The facility must be of a matching type submitted by any contributor.

facility_type=raw%20material%20processing%20or%20production&facility_type=final%20product%20assembly
The parameter may be repeated to search for multiple facility types. This will match facilities with either of the specified facility types.

The value may be set to one of the keys returned from the GET /api/facility-processing-types endpoint.

Processing Type

processing_type=sorting
The facility must have a matching processing type submitted by any contributor.

processing_type=sorting&processing_type=dry%20spinning
May be repeated to search for multiple processing types. This will match facilities with either of the specified processing types.
Values can be any of the processing types listed in the “Facility Taxonomy” table of the taxonomy workbook.

**POST /api/facilities**

Match submitted facility details to an existing facility record or create a new facility record.

**Example CURL Request**

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' --header 'Authorization: Token TOKEN abcl23...' -d '{
   "country": "China",
   "name": "Nantong Jackbeanie Headwear & Garment Co. Ltd.",
   "address": "No.808, the third industry park, Guoyuan Town, Nantong 226500.",
   "number_of_workers": "300 to 1,000",
   "parent_company": "The HAT Group",
   "facility_type_processing_type": ["Assembly", "HQ"],
   "product_type": ["Hats", "Gloves"],
}' 'https://staging.openapparel.org/api/facilities/?create=true'
```

**NOTE:** facility_type and processing type are stored as separate values however they are directly related to each other and for simplicity are submitted to the API using a single **facility_type_processing_type** property in the POST body.

**Sample Data Submission**

Note the misspelling of “Assembly” in **facility_type_processing_type** and how this is corrected in the returned data.

```
{
   "country": "China",
   "name": "Nantong Jackbeanie Headwear & Garment Co. Ltd.",
   "address": "No.808, the third industry park, Guoyuan Town, Nantong 226500.",
   "number_of_workers": "300 to 1,000",
   "parent_company": "The HAT Group",
   "facility_type_processing_type": ["Assembly"],
   "product_type": ["Hats"]
}
```

**Sample Potential Match Response**

For potential match responses, no extended fields are returned until the match is confirmed.

```
{
   "matches": [
```

Sample Automatic Match Response

For automatic matches or new facilities, extended fields are returned immediately.

**NOTE:** Another contributor had already submitted the same value for `facility_type_processing_type` so the `value_count` property is 2 for each of the objects in the `extended_fields.processing_type` array.

**NOTE:** The example matching facility was claimed by the owner and has the name provided by the owner under `extended_fields.name`
"id": 2,
"name": "Service Provider A",
"is_verified": false
},
{
  "id": 4,
  "name": "Brand A (Summer 2019 Affiliate List)",
  "is_verified": false
}
",
"country_name": "China",
"claim_info": null,
"other_locations": [],
"ppe_product_types": null,
"ppe_contact_phone": null,
"ppe_contact_email": null,
"ppe_website": null,
"is_closed": null,
"activity_reports": [],
"contributor_fields": [],
"new_oar_id": null,
"has_inexact_coordinates": false,
"extended_fields": {
  "country": [],
  "name": [
    {
      "id": 5,
      "is_verified": false,
      "value": "Nantong Jackbeanie Headwear Company Limited",
      "updated_at": "2021-11-22T08:40:56.900060Z",
      "contributor_name": "Nantong Jackbeanie Headwear Company Limited",
      "contributor_id": 7,
      "value_count": 1,
      "is_from_claim": true,
      "field_name": "name"
    }
  ]
},
"address": [],
"number_of_workers": [
  {
    "id": 5,
    "is_verified": false,
    "value": {
      "max": 1000,
      "min": 300
    },
    "updated_at": "2021-12-03T19:40:56.900060Z",
}
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 1,
"is_from_claim": false,
"field_name": "number_of_workers"
},

"parent_company": [
{
"id": 6,
"is_verified": false,
"value": {
"name": "The HAT Group",
"raw_value": "The HAT Group"
}
"updated_at": "2021-12-03T19:40:56.900060Z",
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 1,
"is_from_claim": false,
"field_name": "parent_company"
}

"processing_type": [
{
"id": 7,
"is_verified": false,
"value": {
"raw_values": "Asembly",
"matched_values": [
[
"PROCESSING_TYPE",
"FUZZY",
"Final Product Assembly",
"Assembly"
]
]
}
"updated_at": "2021-12-03T19:40:56.900060Z",
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 2,
"is_from_claim": false,
"field_name": "processing_type"
},
{
"id": 8,
"is_verified": false,
"value": {
  "raw_values": "Assembly"
  "matched_values": [
   ["PROCESSING_TYPE",
    "EXACT",
    "Final Product Assembly",
    "Assembly"
   ]
  ]
}

"updated_at": "2021-11-03T13:40:56.900060Z",
"contributor_name": "Brand A",
"contributor_id": 4,
"value_count": 2,
"is_from_claim": false,
"field_name": "processing_type"
},
"facility_type": [
  {
    "id": 9,
    "is_verified": false,
    "value": {
      "raw_values": "Assembly",
      "matched_values": [
       ["PROCESSING_TYPE",
        "FUZZY",
        "Final Product Assembly",
        "Assembly"
       ]
      ]
    }
  
    "updated_at": "2021-12-03T19:40:56.900060Z",
   "contributor_name": "Service Provider A",
   "contributor_id": 2,
   "value_count": 2,
   "is_from_claim": false,
   "field_name": "facility_type"
  },
  {
    "id": 10,
    "is_verified": false,
    "value": {
      "raw_values": "Assembly"
      "matched_values": [
"PROCESSING_TYPE",
"EXACT",
"Final Product Assembly",
"Assembly"
]
]

"updated_at": "2021-11-03T13:40:56.900060Z",
"contributor_name": "Brand A",
"contributor_id": 4,
"value_count": 2,
"is_from_claim": false,
"field_name": "facility_type"
}
],
"product_type": [
{
"id": 11,
"is_verified": false,
"value": {
"raw_values": [
"Hats",
"Gloves"
]
},
"updated_at": "2021-12-03T19:40:56.900060Z",
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 2,
"is_from_claim": false,
"field_name": "product_type"
}
]

"confidence": 0.9588
}
],
"item_id": 936,
"geocoded_geometry": {
"type": "Point",
"coordinates": [
120.596047,
32.172013
]
},
"geocoded_address": "Guoyuanzhen, Rugao, Nantong, Jiangsu, China",
"status": "MATCHED",
}
GET /api/facilities/{id}

Get the full details for a facility specified by OAR ID.

Example CURL Request


Sample Response

{
  "id": "CN20213361HEB5J",
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      120.596047,
      32.172013
    ]
  },
  "properties": {
    "name": "Nantong Jackbeanie Headwear Garment Co. Ltd.",
    "address": "No. 808, The Third Industry Park, Guoyuan Town, Rugao City Nantong",
    "country_code": "CN",
    "oar_id": "CN20213361HEB5J",
    "other_names": [
      "Nantong Jackbeanie Headwear & Garment Co. Ltd."
    ],
    "other_addresses": [
      "No. 808, the third industry park, Guoyuan Town, Nantong 226500."
    ],
    "contributors": [
      {
        "id": 2,
        "name": "Service Provider A",
        "is_verified": false
      },
      {
        "id": 4,
        "name": "Brand A (Summer 2019 Affiliate List)",
        "is_verified": false
      }
    ]
  }
}
"country_name": "China",
"claim_info": null,
"other_locations": [],
"ppe_product_types": null,
"ppe_contact_phone": null,
"ppe_contact_email": null,
"ppe_website": null,
"is_closed": null,
"activity_reports": [],
"contributor_fields": [],
"new_oar_id": null,
"has_inexact_coordinates": false,
"extended_fields": {
  "country": [],
  "name": [
    {
      "id": 5,
      "is_verified": false,
      "value": "Nantong Jackbeanie Headwear Company Limited",
      "updated_at": "2021-11-22T08:40:56.900060Z",
      "contributor_name": "Nantong Jackbeanie Headwear Company Limited",
      "contributor_id": 7,
      "value_count": 1,
      "is_from_claim": true,
      "field_name": "name"
    }
  ],
  "address": [],
  "number_of_workers": [
    {
      "id": 5,
      "is_verified": false,
      "value": {
        "max": 1000,
        "min": 300
      },
      "updated_at": "2021-12-03T19:40:56.900060Z",
      "contributor_name": "Service Provider A",
      "contributor_id": 2,
      "value_count": 1,
      "is_from_claim": false,
      "field_name": "number_of_workers"
    }
  ],
  "parent_company": [
    {
      "id": 6,
"is_verified": false,
"value": {
    "name": "The HAT Group",
    "raw_value": "The HAT Group"
}
"updated_at": "2021-12-03T19:40:56.900060Z",
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 1,
"is_from_claim": false,
"field_name": "parent_company"
},
"processing_type": [
{
    "id": 7,
    "is_verified": false,
    "value": {
        "raw_values": "Assembly",
        "matched_values": [
            ["PROCESSING_TYPE",
             "FUZZY",
             "Final Product Assembly",
             "Assembly"
         ]
     ]
},
"updated_at": "2021-12-03T19:40:56.900060Z",
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 2,
"is_from_claim": false,
"field_name": "processing_type"
},
{
    "id": 8,
    "is_verified": false,
    "value": {
        "raw_values": "Assembly",
        "matched_values": [
            ["PROCESSING_TYPE",
             "EXACT",
             "Final Product Assembly",
             "Assembly"
         ]
     ]
}
"updated_at": "2021-11-03T13:40:56.900060Z",
"contributor_name": "Brand A",
"contributor_id": 4,
"value_count": 2,
"is_from_claim": false,
"field_name": "processing_type"
},
"facility_type": [
{
"id": 9,
"is_verified": false,
"value": {
"raw_values": "Assembly",
"matched_values": [

[ "PROCESSING_TYPE",
  "FUZZY",
  "Final Product Assembly",
  "Assembly"
]
]
}
]"updated_at": "2021-12-03T19:40:56.900060Z",
"contributor_name": "Service Provider A",
"contributor_id": 2,
"value_count": 2,
"is_from_claim": false,
"field_name": "facility_type"
},
{
"id": 10,
"is_verified": false,
"value": {
"raw_values": "Assembly",
"matched_values": [

[ "PROCESSING_TYPE",
  "EXACT",
  "Final Product Assembly",
  "Assembly"
]
]
}
]"updated_at": "2021-11-03T13:40:56.900060Z",
"contributor_name": "Brand A",
"contributor_id": 4,
Interpreting Extended Field Values

The extended_fields property of a facility object returned from the OAR API has the following schema

```json
{
    "field_name_1": [
        { "... submitted value 1 ... },
        { "... submitted value 2 ... }
    ],
    "field_name_2": [
        { "... submitted value 1 ... },
        { "... submitted value 2 ... }
    ]
}
```
Different values for these fields can be submitted by multiple contributors at different times which is why each field key has an array of values. The objects in the array are sorted based on confidence that the submitted value is most correct.

**Shared Schema**

The schema for each submitted value object depends on the field, but all submitted values have these properties in common

<table>
<thead>
<tr>
<th>PROPERTY NAME</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Number</td>
<td>The unique identifier for the submitted value</td>
</tr>
<tr>
<td>is_verified</td>
<td>Boolean</td>
<td>A flag indicating that the value is more likely to be correct</td>
</tr>
<tr>
<td>updated_at</td>
<td>ISO 8601 Date String</td>
<td>The UTC date and time when the value was submitted</td>
</tr>
<tr>
<td>contributor_name</td>
<td>String</td>
<td>The name of the OAR contributor account under which the value was submitted</td>
</tr>
<tr>
<td>contributor_id</td>
<td>Number</td>
<td>The unique identifier of the OAR contributor account under which the value was submitted</td>
</tr>
<tr>
<td>value_count</td>
<td>Number</td>
<td>The total number of times this value was submitted for this field</td>
</tr>
<tr>
<td>is_from_claim</td>
<td>Boolean</td>
<td>True if this value was set from an approved facility claim</td>
</tr>
<tr>
<td>field_name</td>
<td>String</td>
<td>The name of the extended field for which this value was submitted</td>
</tr>
</tbody>
</table>
verified_count | Number | A count of is_verified flags (if both the submitted value and the contributor themselves are verified, this will have a value of 2)

### Field Value Schemas

**Number of Workers**

The value property of a submitted value is an object with the following schema

```json
{
    "min": NUMBER,
    "max": NUMBER
}
```

Contributors may submit `number_of_workers` as either a single value or a range. If a contributor submits a single value, both min and max will be set to that value.

**Parent Company**

The value property of a submitted value is an object with one of two schemas.

If the contributor submitted a company name that was a match to an existing contributor, the value schema will be

```json
{
    "raw_value": STRING,
    "contributor_id": NUMBER,
    "contributor_name": STRING
}
```

If the contributor submitted a company name that did not match an existing contributor name the values schema will be

```json
{
    "raw_value": STRING,
    "name": STRING
}
```
Product Type

The value property of a submitted value is an object with the following schema

```javascript
{
  "raw_values": [
    STRING,
    STRING,
    ...
    STRING
  ]
}
```

Facility Type and Processing Type

The value property for these two fields have the same schema

```javascript
{
  "raw_values": STRING or [STRING, ...],
  "matched_values": [
    [STRING, STRING, STRING, STRING],
    [STRING, STRING, STRING, STRING],
    ...
    [STRING, STRING, STRING, STRING]
  ]
}
```

The items in the matched_values array are an array of 4 elements. All 4 elements will be set to string values or all 4 elements will be null. The order of elements in the matched_values array matches the order of the items in the raw_values array.

[field_type, match_type, facility_type, processing_type]

All processing types have an associated facility type and all values of facility type are also valid values for processing type in the taxonomy. Facility type and processing type are both included in the 4-element array because they are always directly related and one can always be determined using the other.

<table>
<thead>
<tr>
<th>FIELD NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>field_type</td>
<td>Either FACILITY_TYPE or PROCESSING_TYPE</td>
</tr>
<tr>
<td>match_type</td>
<td>One of EXACT, FUZZY, or ALIAS</td>
</tr>
<tr>
<td>facility_type</td>
<td>The matching facility type</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>processing_type</td>
<td>The matching processing type</td>
</tr>
</tbody>
</table>