

Spreading our data further with the Wikibase REST API

Lydia Pintscher

Q18016466

Portfolio Lead for Wikidata



Why a new API for Wikibase?





We want more people
to build applications on
top of Wikidata's data.

(Also other Wikibase instances!)



But it's hard!



WDC
WIKI DATA CON

Wikibase REST API to the rescue!

- Provide a more industry- standard and versioned way to access and manipulate data in Wikibase
- Provide an API that is more tailored to the Wikibase data model
- Solve a number of issues in the current API that are easier to solve with REST



Meet the new Wikibase REST API!

[Documentation](#)

Wikibase REST API ^{0.1} OAS3

OpenAPI definition of Wikibase REST API
Wikimedia Deutschland - Wikibase Product Platform Team - Website
GNU General Public License v2.0 or later

Servers

items Wikibase Items Wikibase Data Model - Items ^

- GET** /entities/items/{item_id} Retrieve a single Wikibase Item by ID

properties Wikibase Properties Wikibase Data Model - Properties ^

- GET** /entities/properties/{property_id} Retrieve a single Wikibase Property by ID

labels Wikibase Labels Wikibase Data Model - Terms ^

- GET** /entities/items/{item_id}/labels Retrieve an Item's labels
- PATCH** /entities/items/{item_id}/labels Change an Item's Labels
- GET** /entities/properties/{property_id}/labels Retrieve a Property's labels
- PATCH** /entities/properties/{property_id}/labels [WSP] Change a Property's Labels
- GET** /entities/items/{item_id}/labels/{language_code} Retrieve an Item's label in a specific language
- PUT** /entities/items/{item_id}/labels/{language_code} Add / Replace an Item's label in a specific language
- GET** /entities/properties/{property_id}/labels/{language_code} Retrieve a Property's label in a specific language
- PUT** /entities/properties/{property_id}/labels/{language_code} Add / Replace a Property's label in a specific language

descriptions Wikibase Descriptions Wikibase Data Model - Terms ^

- GET** /entities/items/{item_id}/descriptions Retrieve an Item's descriptions
- PATCH** /entities/items/{item_id}/descriptions Change an Item's descriptions
- GET** /entities/properties/{property_id}/descriptions Retrieve a Property's descriptions
- GET** /entities/items/{item_id}/descriptions/{language_code} Retrieve an Item's description in a specific language
- PUT** /entities/items/{item_id}/descriptions/{language_code} Add / Replace an Item's description in a specific language
- GET** /entities/properties/{property_id}/descriptions/{language_code} Retrieve a Property's description in a specific language
- PUT** /entities/properties/{property_id}/descriptions/{language_code} Add / Replace a Property's description in a specific language

What you can already do

- Get a single Item/Property and their labels, descriptions and aliases (in all and specific languages)
- Get data of a statement or all statements of an Item/Property
- Modify data of a single statement on an Item/Property
- Modify labels and descriptions of an Item/Property



What's different in the REST API?

Item/Property data

-> use statements instead of claims

```
GET /w/api.php?action=wbgetentities&ids=Q11&format=json
```

```
{
  "entities": {
    "Q11": {
      "type": "item",
      "id": "Q11",
      "labels": { ... },
      "descriptions": { ... },
      "aliases": { ... },
      "claims": {
        "P31": [
          { ... }
        ]
      },
      "sitelinks": {}
    }
  }
}
```

```
GET /w/rest.php/wikibase/v0/entities/items/Q11
```

```
{
  "type": "item",
  "id": "Q11",
  "labels": { ... },
  "descriptions": { ... },
  "aliases": { ... },
  "statements": {
    "P31": [
      { ... }
    ]
  },
  "sitelinks": {}
}
```

Item/Property data

-> use flat structure

GET /w/api.php?action=wbgetentities&ids=Q11&format=json

```
{
  "entities": {
    "Q11": {
      "type": "item",
      "id": "Q11",
      "labels": {
        "en": {
          "language": "en",
          "value": "non-empty-item-R5Gt64V3Eg"
        }
      },
      "descriptions": {
        "en": {
          "language": "en",
          "value": "non-empty-item-description"
        }
      },
      "aliases": {
        "en": [
          {
            "language": "en",
            "value": "non-empty-item-alias"
          }
        ]
      },
      "claims": { ... },
      "sitelinks": {}
    }
  }
}
```

GET /w/rest.php/wikibase/v0/entities/items/Q11

```
{
  "type": "item",
  "id": "Q11",
  "labels": {
    "en": "non-empty-item-R5Gt64V3Eg"
  },
  "descriptions": {
    "en": "non-empty-item-description"
  },
  "aliases": {
    "en": [
      "non-empty-item-alias"
    ]
  },
  "statements": { ... },
  "sitelinks": {}
}
```

Statement data

- > mainsnak field is removed
- > mainsnak.hash and datavalue.type are removed
- > New top level property field is introduced, holding both the id and data-type
- > New top level value field is added, consisting of two fields: content and type
- > All fields are always included in the response, even when empty

old

```
{
  "claims": {
    "P31": [
      {
        "mainsnak": {
          "snaktype": "value",
          "property": "P31",
          "hash": "884531f6c60d8fbf3030857f2abd2086337af23c",
          "datavalue": {
            "value": "something",
            "type": "string"
          },
          "datatype": "string"
        },
        "type": "statement",
        "id": "Q11$4A2F60EA-C779-42D5-8516-A8C26E3ED571",
        "rank": "normal"
      }
    ]
  }
}
```

new

```
{
  "statements": {
    "P31": [
      {
        "id": "Q11$4A2F60EA-C779-42D5-8516-A8C26E3ED571",
        "rank": "normal",
        "qualifiers": [],
        "references": [],
        "property": {
          "id": "P31",
          "data-type": "string"
        },
        "value": {
          "type": "value",
          "content": "something"
        }
      }
    ]
  }
}
```

Metadata

- > pageid, ns, and title fields are omitted
- > revision ID and last modified date are included in HTTP response headers
- > success state is represented in HTTP response code (2xx vs 4xx)

```
GET /w/api.php?action=wbgetentities&ids=Q11&format=json
```

```
{
  "entities": {
    "Q11": {
      "pageid": 17,
      "ns": 120,
      "title": "Item:Q11",
      "lastrevid": 1484,
      "modified": "2022-10-10T13:21:24Z",
      "type": "item",
      "id": "Q11",
      "labels": { ... },
      "descriptions": { ... },
      "aliases": { ... },
      "claims": { ... },
      "sitelinks": {}
    }
  },
  "success": 1
}
```

```
GET /w/rest.php/wikibase/v0/entities/items/Q11
```

```
{
  "type": "item",
  "id": "Q11",
  "labels": { ... },
  "descriptions": { ... },
  "aliases": { ... },
  "statements": { ... },
  "sitelinks": {}
}
```

Set a label / description via PUT

Action API:

```
api.php?action=wbsetlabel&id=Q4115189&language=en&value=Wikimedia&format=jsonfm
```

REST API:

The screenshot shows a REST client interface with the following details:

- Method:** PUT
- URL:** /rest.php/wikibase/v0/entities/items/Q4115189/labels/en-us
- Status:** 200 OK
- Time:** 2.68 s
- Size:** 9 B
- Response Body (JSON):**

```
1 {  
2   "label": "Sandbox",  
3   "comment": "set new en-us label"  
4 }
```
- Response Body (Preview):**

```
1 "Sandbox"
```

Create a statement via POST

Action API:

```
api.php?action=wbcREATEclaim&entity=Q4115189&property=P9002&snaktype=value&value="itsastring"
```

REST API:

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** /rest.php/wikibase/v0/entities/items/Q10/statements
- Status:** 201 Created
- Response Time:** 84.7 ms
- Response Size:** 191 B
- Authentication:** Bearer
- Response Format:** JSON
- Response Body (Left Panel):**

```
1 {
2   "statement": {
3     "property": {
4       "id": "P123"
5     },
6     "value": {
7       "type": "value",
8       "content": "itsastring"
9     }
10  }
11 }
```
- Response Body (Right Panel - Preview):**

```
1 {
2   "id": "Q10$276DB959-2A33-43D9-A006-1E8471180799",
3   "rank": "normal",
4   "qualifiers": [],
5   "references": [],
6   "property": {
7     "id": "P123",
8     "data-type": "string"
9   },
10  "value": {
11    "type": "value",
12    "content": "itsastring"
13  }
14 }
```

Modify labels / descriptions via PATCH

PATCH `/rest.php/wikibase/v0/entities/items/Q10/labels` Send **200 OK** 74.9 ms 79 B

JSON ▼ Bearer ▼ Query Headers ¹ Docs Preview ▼ Headers ¹¹ Cookies

```
1 {
2   "patch": [
3     {
4       "op": "remove",
5       "path": "/de-ch"
6     },
7     {
8       "op": "add",
9       "path": "/en-gb",
10      "value": "Wikidata Sandbox"
11    },
12    {
13      "op": "replace",
14      "path": "/en-us",
15      "value": "Wikidata Sandbox"
16    }
17  ]
18 }
19
```

```
1 {
2   "en": "Wikidata Sandbox",
3   "en-gb": "Wikidata Sandbox",
4   "en-us": "Wikidata Sandbox"
5 }
```

What's next?

- Modifying aliases of Items/Properties
- Getting and modifying sitelink data
- Even better error responses
- API implementation versioning

See you on Wikidata!

Try it out and give feedback:

[Wikidata:REST API](#)

Get in touch:

- lydia.pintscher@wikimedia.de
- [nightrose@mastodon.social](https://mstdn.social/@nightrose)
- [User:Lydia Pintscher \(WMDE\)](#)
- [Q18016466](#)