

---

# **browsercompat API Documentation**

*Release 0.1.0*

**Mozilla**

December 16, 2015



<b>1</b>	<b>Status</b>	<b>3</b>
<b>2</b>	<b>Development</b>	<b>5</b>
2.1	Installation . . . . .	5
2.2	Contributing . . . . .	7
2.3	Technologies . . . . .	9
2.4	v1 BrowserCompat API . . . . .	9
2.5	Tools . . . . .	64
2.6	Issues . . . . .	67
2.7	History . . . . .	69
2.8	Indices and tables . . . . .	69



The Browser Compatibility API will support compatibility data on the [Mozilla Developer Network](#). This currently takes the form of browser compatibility tables, such as the one on the [CSS display property](#) page. The API will centralize this data, and allow it to be kept consistent across languages and presentations.

The project started in December 2013. The initial goals and requirements are documented on the [MozillaWiki](#).



---

### Status

---

The beta v1 API is being served at <https://browsercompat.herokuapp.com/api/v1/>. Alpha users are using the `importer` to find and fix data issues on MDN. A small number of pages on MDN have been converted to use API-backed compatibility tables. Beta users can view the new tables, and non-beta users see the traditional wiki-backed tables. As the beta is expanded to more pages and more users, the API is changed to handle new use cases. See the issues page for details of planned changes.

The v1 API uses release candidate 1 (RC1) of the JSON API specification, which was released July 2014, but is currently undocumented. See the v1 API docs for details of the API implementation.





---

## Development

---

**Code** <https://github.com/mdn/browsercompat>

**Server** <https://browsercompat.herokuapp.com> (based on [mdn/browsercompat-data](https://github.com/mdn/browsercompat-data))

**Issues** <https://bugzilla.mozilla.org/buglist.cgi?quicksearch=compat-data> (tracking bug)

[https://bugzilla.mozilla.org/showdependencytree.cgi?id=996570&hide\\_resolved=1](https://bugzilla.mozilla.org/showdependencytree.cgi?id=996570&hide_resolved=1) (blocking issues for v1)

**Dev Docs** <https://browsercompat.readthedocs.org>

<https://github.com/mdn/browsercompat/wiki>

**Mailing list** <https://lists.mozilla.org/listinfo/dev-mdn>

**IRC** <irc://irc.mozilla.org/mdndev>

Contents:

## 2.1 Installation

### 2.1.1 Install Django Project

For detailed local instation instructions, including OS-specific instructions, see the [Installation page on the wiki](#).

1. Install system packages and libraries. The required packages are [Python](#) (2.7, 3.4, or both), [pip](#) (latest), and [virtualenv](#) (latest). To match production and for a smooth installation of Python packages, install [PostgreSQL](#) (9.2 or later recommended) and [Memcached](#) (latest). [virtualenvwrapper](#) and [autoenv](#) will make your development life easier.
2. Optionally, provision a PostgreSQL database, recommended to match production. The default Django database settings will use a [SQLite](#) database named `db.sqlite3`.
3. Optionally, run [Redis](#) or [Memcached](#) for improved read performance (production uses Redis for caching and for the [Celery](#) backend) The default settings will run without a cache or asynchronous tasks.
4. [Clone project locally](#).
5. [Create a virtualenv](#).
6. Install dependencies with `pip install -r requirements.txt -r requirements-dev.txt`.
7. Customize the configuration with environment variables. See `wpcsite/settings.py` and `env.dist` for advice and available settings.

8. Initialize the database and a superuser account with `./manage.py migrate`.
9. Verify that tests pass with `./manage.py test` or `make test`.
10. Run it with `./manage.py runserver` or `./manage.py runserver_plus`.

### 2.1.2 Install in Heroku

Heroku allows you to quickly deploy browsercompat. Heroku hosts the beta version of the service at <https://browsercompat.herokuapp.com>, using the add-ons:

- [heroku-postgresql](#) (hobby-basic tier, \$9/month, required for size of dataset)
- [memcachier](#) (free dev tier)
- [heroku-redis](#) (free hobby-dev tier)
- Mozilla's New Relic account ([Heroku New Relic](#) available, including free [Wayne](#) tier)

For more details of the beta server, see [Beta Server](#) on the github wiki.

To deploy with Heroku, you'll need to [signup for a free account](#) and install the [Heroku Toolbelt](#). Then you can:

1. Clone project locally
2. `heroku apps:create`
3. `git push heroku master`
4. See the current config with `heroku config`, and then customize with environment variables using `heroku config:set` (see `wpcsite/settings.py` and `env.dist`)
5. Add superuser account (`heroku run ./manage.py createsuperuser`)

### 2.1.3 Configuring authentication

The project uses [django-allauth](#) as a framework for local and social authentication. The [public service](#) uses username and password for local authentication, and [Firefox Accounts](#) (FxA) for social authentication.

django-allauth supports multiple emails per user, with one primary email used for communication. Email addresses are validated by sending a confirmation link. For a public server, you'll need to [configure Django to send email](#), by configuring your mail server and setting environment variables. For local development, it is easiest to print emails to the console:

```
export EMAIL_BACKEND="django.core.mail.backends.console.EmailBackend"
```

django-allauth supports many social authentication providers. See the [providers documentation](#) for the current list and hints for configuration. Using a authentication provider is not required, especially for local development. Instead, use local authentication with a username and password.

If you need FxA integration, see the [Firefox Accounts page on the wiki](#) for install hints.

### 2.1.4 Load Data

There are several ways to get data into your API:

1. Load data from the github export
2. Load data from another browsercompat server
3. Load sample data from the [WebPlatform project](#) and [MDN](#)

## Load from GitHub

The data on [browsercompat.herokuapp.com](https://browsercompat.herokuapp.com) is archived in the [browsercompat-data](#) github repo, and this is the fastest way to get data into your empty API:

1. Clone the github repo (`git clone https://github.com/mdn/browsercompat-data.git`)
2. Run the API (`./manage.py runserver`)
3. Import the data (`tools/upload_data.py --data /path/to/browsercompat-data/data`)

## Load from another browsercompat server

If you have read access to a browsercompat server that you'd like to clone, you can grab the data for your own server.

1. Download the data (`tools/download_data.py --api https://browsercompat.example.com`)
2. Run the API (`./manage.py runserver`)
3. Import the data (`tools/upload_data.py`)

## Load Sample Data

The [WebPlatform project](#) imported data from [MDN](#), and stored the formatted compatibility data in a [github project](#). There is a lot of data that was not imported, so it's not a good data source for re-displaying on MDN. However, combining this data with specification data from MDN will create a good data set for testing the API at scale.

To load sample data:

1. Run the API (`./manage.py runserver`)
2. Load a subset of the WebPlatform data (`tools/load_webcompat_data.py`) or full set of data (`tools/load_webcompat.py --all-data`)
3. Load specification data (`tools/load_spec_data.py`)

## 2.2 Contributing

Contributions should follow the [MDN Contribution Guidelines](#), and follow the standards of this project:

- You agree to license your contributions under [MPL 2](#)
- Discuss large changes on the [dev-mdn mailing list](#) or on a [bugzilla bug](#) before coding.
- **Code should follow Python and JavaScript standards.**
  - Python code style should conform to [PEP8 standards](#), as well as [PEP257 standards](#) for documentation strings.
  - JavaScript code should follow the conventions of [jslint](#).
  - Code can be checked locally with `make lint`, which must pass before a pull request will be considered.
- **All commit messages must start with “bug NNNNNNN” or “fix bug NNNNNNN”**
  - Reason: This makes it possible to read the commit log and find the bug that includes the original request and design conversation for all changes.
  - Exceptions: “Merge” and “Revert” commits

- “fix bug NNNNNNNN” - will trigger a github bot to automatically mark the bugzilla bug as “RESOLVED:FIXED”
- Multiple commits are allowed, and authors are encouraged to split, merge, and rearrange commits in feature branches to make code review easier. Each commit must start with the “bug” or “fix bug” prefix.
- **MDN module owner or peer must review and merge all pull requests.**
  - Reason: Owner and peers are accountable for the quality of MDN code changes
  - Exceptions: Owners/peers may commit directly to master for critical security/down-time fixes; they must file a bug for follow-up review.
- **MDN reviewers must verify 100% test coverage on all changes.**
  - Reason: Automated tests reduce human error involved in reviews, and allow radical refactoring of the code.
  - Notes: The Django site has [good testing docs](#), and Django REST framework has some [additional testing docs](#).
  - Coverage can be checked locally with `make coverage`.

### 2.2.1 What to work on

There is a [tracking bug](#) for this project, and a [specific bug](#) for the data store, the primary purpose of this project. The dependant bugs represent areas of work, and are not exhaustive. If you want to contribute at this phase in development, take a look at the bugs and linked documents, to familiarize yourself with the project, and then get in touch with the team on IRC ([#mdndev](#) or [#browsercompat](#) on [irc.mozilla.org](#)) to carve out a piece of the project.

### 2.2.2 GitHub workflow

1. Get your environment setup
2. Set up mozilla remote (`$ git remote add mozilla git://github.com/mdn/browsercompat.git`)
3. Create a branch for a bug (`$ git checkout -b new-issue-888888`)
4. Develop on bug branch.  
*Time passes, the mdn/browsercompat repository accumulates new commits*
5. Commit changes to bug branch (`$ git add . ; git commit -m 'fix bug 888888 - commit message'`)
6. Fetch mozilla (`$ git fetch mozilla`)
7. Update local master (`$ git checkout master; git pull mozilla master`)  
*Repeat steps 4-7 till dev is complete*
8. Rebase issue branch (`$ git checkout new-issue-888888; git rebase master`)
9. Push branch to GitHub (`$ git push origin new-issue-888888`)
10. Issue pull request (Click Pull Request button)

## 2.3 Technologies

The technologies used to implement the API include:

- Django, a web framework
- Django REST Framework, an API framework
- django-simple-history, for recording changes to models
- django-mptt, for efficiently storing hierarchical data
- django-oauth2-provider, for script-based updates of content

See the [source code](#) for a detailed list.

## 2.4 v1 BrowserCompat API

The v1 API was designed to store and maintain information about web technologies, such as HTML and CSS, in the manner used on [MDN](#). This takes the form of **Specification** tables, which detail the specifications for technologies, and **Browser Compatibility** tables, which detail when browsers implemented those technologies. A simple example is for the [HTML element <address>](#). A more complex example is the [CSS property display](#).

The v1 API was designed in March 2014, and implemented over the next year. It was allowed to be partially implemented, and labeled as “draft”, so that the design could be modified as new problems were discovered. The API as designed is mostly implemented as of December 2015, but further changes will be made to the v1 API in 2016. See the [issues page](#) for details.

The v1 API was based on release candidate 1 (RC1) of the [JSON API](#) specification, which was released 2014-07-05. Starting with RC2 on 2015-02-18, The JSON API team rapidly iterated on the design, making significant changes informed by the experience of implementors. [JSON API v1.0](#) was released May 2015, and is significantly different from RC1. The JSON API team does not preserve documentation for release candidates (online or with a git tag), so it is impossible to refer to the documentation for RC1.

v1 will remain on JSON API RC1. The next version of the API, v2, will support JSON API 1.0. Both the v1 and v2 APIs will be supported until the tools are updated, and then the v1 API will be retired.

### 2.4.1 Entrypoints

The v1 API can be found at <https://browsercompat.herokuapp.com/api/v1/>. It includes a browsable API to ease application development.

The API supports two representations:

**application/vnd.api+json** (*default*) JSON mostly conforming to the [JSON API RC1](#) specification.

**text/html** the Django REST Framework browsable API.

The API supports user accounts with [Persona](#) authentication.

A developer-centered website is available at <https://browserscompat.herokuapp.com/>. This site includes a [data browser](#) and an [importer](#) for scraping data from [MDN](#).

### 2.4.2 Resources

Resources are simple objects supporting [CRUD](#) operations. Read operations can be done anonymously. Creating and updating require account permissions, and deleting requires admin account permissions.

All resources support similar operations using HTTP methods:

- GET /api/v1/<type> - List or search instances (paginated)
- POST /api/v1/<type> - Create a new instance
- GET /api/v1/<type>/<id> - Retrieve an instance
- PUT /api/v1/<type>/<id> - Update an instance
- DELETE /api/v1/<type>/<id> - Delete an instance

Because the operations are similar, only *browsers* has complete examples, and others just show retrieving an instance (GET /api/v1/<type>/<id>). Full requests and responses are generated and stored in the [source repository](#)

### Contents

- *Resources*
  - *Browsers*
    - \* *List*
    - \* *Retrieve by ID*
    - \* *Retrieve by Slug*
    - \* *Create*
    - \* *Update*
    - \* *Partial Update*
    - \* *Update order of related resources*
    - \* *Reverting to a previous instance*
    - \* *Deletion*
    - \* *Reverting a deletion*
  - *Versions*
  - *Features*
  - *Supports*
  - *Specifications*
  - *Sections*
  - *Maturities*

## Browsers

A **browser** is a brand of web client that has one or more versions. This follows most users' understanding of browsers, i.e., `firefox_desktop` represents desktop Firefox, `safari_desktop` represents desktop Safari, and `firefox_android` represents Firefox on Android.

The **browsers** representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **slug** (*write-once*) - Unique, human-friendly slug
  - **name** (*localized*) - Browser name
  - **note** (*localized*) - Notes, intended for related data like OS, applicable device, engines, etc.
- **links**
  - **versions** (*many*) - Associated *versions*, ordered roughly from earliest to latest. User can change the order.

- **history\_current** (*one*) - Current historical\_browsers. Can be set to a value from **history** to revert changes.
- **history** (*many*) - Associated historical\_browsers in time order (most recent first). Changes are ignored.

*Note: bug 1078699 is proposing that select users will be able to modify slugs*

## List

To request the paginated list of **browsers**:

```
GET /api/v1/browsers HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "browsers": [
    {
      "id": "1",
      "slug": "android",
      "name": {
        "en": "Android Browser"
      },
      "note": null,
      "links": {
        "history": [
          "1"
        ],
        "history_current": "1",
        "versions": [
          "1",
          "2",
          "3"
        ]
      }
    },
    {
      "id": "2",
      "slug": "android_webview",
      "name": {
        "en": "Android Webview"
      },
      "note": null,
      "links": {
        "history": [
          "2"
        ],
        "history_current": "2",
        "versions": [
          "4"
        ]
      }
    }
  ]
}
```

```
    },
    {
      "id": "3",
      "slug": "blackberry",
      "name": {
        "en": "BlackBerry Browser"
      },
      "note": null,
      "links": {
        "history": [
          "3"
        ],
        "history_current": "3",
        "versions": [
          "5",
          "6",
          "7"
        ]
      }
    },
    {
      "id": "4",
      "slug": "chrome_desktop",
      "name": {
        "en": "Chrome for Desktop"
      },
      "note": null,
      "links": {
        "history": [
          "4"
        ],
        "history_current": "4",
        "versions": [
          "8",
          "9",
          "10"
        ]
      }
    },
    {
      "id": "5",
      "slug": "chrome_for_android",
      "name": {
        "en": "Chrome for Android"
      },
      "note": null,
      "links": {
        "history": [
          "5"
        ],
        "history_current": "5",
        "versions": [
          "11",
          "12",
          "13"
        ]
      }
    }
  ],
}
```



```
{
  "id": "6",
  "slug": "firefox_android",
  "name": {
    "en": "Firefox for Android"
  },
  "note": null,
  "links": {
    "history": [
      "6"
    ],
    "history_current": "6",
    "versions": [
      "14",
      "15",
      "16"
    ]
  }
},
{
  "id": "7",
  "slug": "firefox_desktop",
  "name": {
    "en": "Firefox for Desktop"
  },
  "note": null,
  "links": {
    "history": [
      "7"
    ],
    "history_current": "7",
    "versions": [
      "17",
      "18",
      "19",
      "20",
      "21"
    ]
  }
},
{
  "id": "8",
  "slug": "firefox_os",
  "name": {
    "en": "Firefox OS"
  },
  "note": null,
  "links": {
    "history": [
      "8"
    ],
    "history_current": "8",
    "versions": [
      "22",
      "23",
      "24"
    ]
  }
}
```

```
    },
    {
      "id": "9",
      "slug": "ie_desktop",
      "name": {
        "en": "Internet Explorer for Desktop"
      },
      "note": null,
      "links": {
        "history": [
          "9"
        ],
        "history_current": "9",
        "versions": [
          "25",
          "26",
          "27",
          "28",
          "29"
        ]
      }
    },
    {
      "id": "10",
      "slug": "ie_mobile",
      "name": {
        "en": "Internet Explorer Mobile"
      },
      "note": null,
      "links": {
        "history": [
          "10"
        ],
        "history_current": "10",
        "versions": [
          "30",
          "31",
          "32"
        ]
      }
    }
  ],
  "links": {
    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  },
  "meta": {
    "pagination": {
```

```

    "browsers": {
      "previous": null,
      "next": "https://browsercompat.org/api/v1/browsers?page=2",
      "count": 15
    }
  }
}

```

## Retrieve by ID

To request a single **browser** with a known ID:

```

GET /api/v1/browsers/7 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json

```

A sample response is:

```

HTTP/1.1 200 OK
Content-Type: application/vnd.api+json

```

```

{
  "browsers": {
    "id": "7",
    "slug": "firefox_desktop",
    "name": {
      "en": "Firefox for Desktop"
    },
    "note": null,
    "links": {
      "history": [
        "7"
      ],
      "history_current": "7",
      "versions": [
        "17",
        "18",
        "19",
        "20",
        "21"
      ]
    }
  },
  "links": {
    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  }
}

```

```
}  
}
```

### Retrieve by Slug

*Note: bug 1078699 is proposing an alternate URL format.*

To request a **browser** by slug:

```
GET /api/v1/browsers?slug=firefox_desktop HTTP/1.1  
Host: browsercompat.org  
Accept: application/vnd.api+json
```

The response includes the desired browser, in list format:

```
HTTP/1.1 200 OK  
Content-Type: application/vnd.api+json
```

```
{  
  "browsers": [  
    {  
      "id": "7",  
      "slug": "firefox_desktop",  
      "name": {  
        "en": "Firefox for Desktop"  
      },  
      "note": null,  
      "links": {  
        "history": [  
          "7"  
        ],  
        "history_current": "7",  
        "versions": [  
          "17",  
          "18",  
          "19",  
          "20",  
          "21"  
        ]  
      }  
    }  
  ],  
  "links": {  
    "browsers.history": {  
      "type": "historical_browsers",  
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"  
    },  
    "browsers.history_current": {  
      "type": "historical_browsers",  
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"  
    },  
    "browsers.versions": {  
      "type": "versions",  
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"  
    }  
  },  
  "meta": {
```

```

    "pagination": {
      "browsers": {
        "previous": null,
        "next": null,
        "count": 1
      }
    }
  }
}

```

## Create

Creating **browser** instances require authentication with create privileges. To create a new **browser** instance, POST a representation with at least the required parameters. Some items (such as the `id` attribute and the `history_current` link) will be picked by the server, and will be ignored if included.

Here's an example of creating a **browser** instance, with cookie-based authentication:

```

POST /api/v1/browsers HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 132
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn

```

```

{
  "browsers": {
    "slug": "amazon-silk-mobile",
    "name": {
      "en": "Amazon Silk Mobile"
    }
  }
}

```

A sample response is:

```

HTTP/1.1 201 CREATED
Content-Type: application/vnd.api+json

```

```

{
  "browsers": {
    "id": "16",
    "slug": "amazon-silk-mobile",
    "name": {
      "en": "Amazon Silk Mobile"
    }
  },
  "note": null,
  "links": {
    "history": [
      "16"
    ],
    "history_current": "16",
    "versions": []
  }
},
"links": {

```

```

    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  }
}

```

This, and other methods that change resources, will create a new changeset, and associate the new `historical_browsers` with that changeset. To assign to an existing changeset, add it to the URI:

```

POST /api/v1/browsers?changeset=4 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 220
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn

```

```

{
  "browsers": {
    "slug": "nintendo-ds",
    "name": {
      "en": "Nintendo DS Browser",
      "ja": "\u30cb\u30f3\u30c6\u30f3\u30c9\u30fc\u30d6\u30e9\u30a6\u30b6"
    }
  }
}

```

A sample response is:

```

HTTP/1.1 201 CREATED
Content-Type: application/vnd.api+json

```

```

{
  "browsers": {
    "id": "18",
    "slug": "nintendo-ds",
    "name": {
      "en": "Nintendo DS Browser",
      "ja": ""
    },
    "note": null,
    "links": {
      "history": [
        "18"
      ],
      "history_current": "18",
      "versions": []
    }
  },
  "links": {

```

```

    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  }
}

```

## Update

Updating a **browser** instance require authentication with create privileges. Some items (such as the `id` attribute and `history` links) can not be changed, and will be ignored if included. A successful update will return a 200 OK, add a new ID to the `history` links list, and update the `history_current` link.

This update changes the English name from “Internet Explorer” to “Microsoft Internet Explorer”:

```

PUT /api/v1/browsers/9 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 1000
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn

```

```

{
  "browsers": {
    "id": "9",
    "slug": "ie_desktop",
    "name": {
      "en": "Microsoft Internet Explorer"
    },
    "note": null,
    "links": {
      "history": [
        "9"
      ],
      "history_current": "9",
      "versions": [
        "25",
        "26",
        "27",
        "28",
        "29"
      ]
    }
  },
  "links": {
    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    }
  }
}

```

```
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  }
}
```

With this response:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "browsers": {
    "id": "9",
    "slug": "ie_desktop",
    "name": {
      "en": "Microsoft Internet Explorer"
    },
    "note": null,
    "links": {
      "history": [
        "19",
        "9"
      ],
      "history_current": "19",
      "versions": [
        "25",
        "26",
        "27",
        "28",
        "29"
      ]
    }
  },
  "links": {
    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  }
}
```



## Partial Update

An update can just update the target fields. This is a further request to change the English name for the Internet Explorer browser.

```
PUT /api/v1/browsers/9 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 78
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn
```

```
{
  "browsers": {
    "name": {
      "en": "IE"
    }
  }
}
```

With this response:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "browsers": {
    "id": "9",
    "slug": "ie_desktop",
    "name": {
      "en": "IE"
    },
    "note": null,
    "links": {
      "history": [
        "20",
        "19",
        "9"
      ],
      "history_current": "20",
      "versions": [
        "25",
        "26",
        "27",
        "28",
        "29"
      ]
    }
  },
  "links": {
    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    }
  }
}
```

```
"browsers.versions": {
  "type": "versions",
  "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
}
}
```

### Update order of related resources

In many cases, related resources (which appear in the “links” attribute”) are sorted by ID. In some cases, the order is significant, and is set on a related field. For example, **versions** for a **browser** are ordered by updating the order on the **browser** object:

To change just the *versions* order:

```
PUT /api/v1/browsers/9 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 205
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn
```

```
{
  "browsers": {
    "links": {
      "versions": [
        "25",
        "26",
        "27",
        "29",
        "28"
      ]
    }
  }
}
```

With this response:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "browsers": {
    "id": "9",
    "slug": "ie_desktop",
    "name": {
      "en": "IE"
    },
    "note": null,
    "links": {
      "history": [
        "21",
        "20",
        "19",
        "9"
      ]
    }
  },
}
```

```

    "history_current": "21",
    "versions": [
      "25",
      "26",
      "27",
      "29",
      "28"
    ]
  },
  "links": {
    "browsers.history": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
    },
    "browsers.history_current": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
    },
    "browsers.versions": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
    }
  }
}

```

### Reverting to a previous instance

To revert to an earlier instance, set the `history_current` link to a previous value. This resets the content and creates a new `historical_browsers` object:

```

PUT /api/v1/browsers/9 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 91
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn

```

```

{
  "browsers": {
    "links": {
      "history_current": "9"
    }
  }
}

```

With this response (note that the name and version order have reverted to the *original values*):

```

HTTP/1.1 200 OK
Content-Type: application/vnd.api+json

```

```

{
  "browsers": {
    "id": "9",
    "slug": "ie_desktop",
    "name": {

```

```

    "en": "Internet Explorer for Desktop"
  },
  "note": null,
  "links": {
    "history": [
      "22",
      "21",
      "20",
      "19",
      "9"
    ],
    "history_current": "22",
    "versions": [
      "25",
      "26",
      "27",
      "29",
      "28"
    ]
  }
},
"links": {
  "browsers.history": {
    "type": "historical_browsers",
    "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
  },
  "browsers.history_current": {
    "type": "historical_browsers",
    "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
  },
  "browsers.versions": {
    "type": "versions",
    "href": "https://browsercompat.org/api/v1/versions/{browsers.versions}"
  }
}
}

```

## Deletion

To delete a browser:

```

DELETE /api/v1/browsers/16 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 0
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn

```

The response has no body:

```
HTTP/1.1 204 NO CONTENT
```

## Reverting a deletion

Reverting deletions is not currently possible, and is tracked in [bug 1159349](#).

## Versions

A **version** is a specific release of a Browser.

The **versions** representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **version** (*write-once*) - Version of browser. Numeric or text string, depending on the status (see table below).
  - **release\_day** - Day that browser was released in [ISO 8601](#) format, or null if unknown.
  - **retirement\_day** - Approximate day the browser was “retired” (stopped being a current browser), in [ISO 8601](#) format, or null if unknown.
  - **status** - One of beta, current, future, retired-beta, retired, or unknown (see table below).
  - **release\_notes\_uri** (*localized*) - URI of release notes for this version, or null if none.
  - **note** (*localized*) - Engine, OS, etc. information, or null
  - **order** (*read-only*) - The relative order among versions for this browser. The order can be changed on the **browser** resource.
- **links**
  - **browser** - The related **browser**
  - **supports** (*many*) - Associated **supports**, in ID order. Changes are ignored; work on the **supports** to add, change, or remove.
  - **history\_current** (*one*) - Current **historical\_versions**. Set to a value from **history** to revert to that version.
  - **history** (*many*) - Associated **historical\_versions**, in time order (most recent first). Changes are ignored.

The version is either a numeric value, such as "11.0", or text, such as "Nightly". The version format depends on the chosen status:

Status	Version	Meaning
beta	numeric	A release candidate suggested for early adopters or testers
current	numeric or the text "current"	A current and preferred release for most users
future	text such as "Nightly"	A named but unnumbered future release
retired-beta	numeric	An old beta version, replaced by a new beta or release
retired	numeric	An old released version no longer recommended for users
unknown	numeric	A release with an unknown status

To get a single **version**:

```
GET /api/v1/versions/21 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "versions": {
    "id": "21",
    "version": "16.0",
    "release_day": "2012-10-09",
    "retirement_day": "2012-11-20",
    "status": "retired",
    "release_notes_uri": {
      "en": "https://developer.mozilla.org/en/Firefox/Releases/16",
      "de": "https://developer.mozilla.org/de/Firefox/Releases/16",
      "es": "https://developer.mozilla.org/es/Firefox/Releases/16",
      "fr": "https://developer.mozilla.org/fr/Firefox/Versions/16",
      "ja": "https://developer.mozilla.org/ja/Firefox/Releases/16",
      "ko": "https://developer.mozilla.org/ko/Firefox/Releases/16",
      "pl": "https://developer.mozilla.org/pl/Firefox/Releases/16",
      "pt-PT": "https://developer.mozilla.org/pt-PT/Firefox/Releases/16",
      "ru": "https://developer.mozilla.org/ru/Firefox/Releases/16",
      "zh-CN": "https://developer.mozilla.org/zh-CN/Firefox/Releases/16",
      "zh-TW": "https://developer.mozilla.org/zh-TW/Firefox/Releases/16"
    },
    "note": null,
    "order": 4,
    "links": {
      "browser": "7",
      "supports": [
        "12",
        "22"
      ],
      "history": [
        "21"
      ],
      "history_current": "21"
    }
  },
  "links": {
    "versions.browser": {
      "type": "browsers",
      "href": "https://browsercompat.org/api/v1/browsers/{versions.browser}"
    },
    "versions.supports": {
      "type": "supports",
      "href": "https://browsercompat.org/api/v1/supports/{versions.supports}"
    },
    "versions.history": {
      "type": "historical_versions",
      "href": "https://browsercompat.org/api/v1/historical_versions/{versions.history}"
    },
    "versions.history_current": {
      "type": "historical_versions",
      "href": "https://browsercompat.org/api/v1/historical_versions/{versions.history_current}"
    }
  }
}
```

## Features

A **feature** is a web technology. This could be a precise technology, such as the value `cover` for the CSS `background-size` property. It could be a hierarchical group of related technologies, such as the CSS `background-size` property or the set of all CSS properties. Some features correspond to a page on [MDN](#), which will display the list of specifications and a browser compatibility table of the sub-features.

The **features** representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **slug** (*write-once*) - Unique, human-friendly slug
  - **mdn\_uri** (*optional, localized*) - The URI of the language-specific MDN page that this feature was first scraped from. If the path contains unicode, it should be percent-encoded as in [RFC 3987](#). May be used in UX or for debugging import scripts.
  - **experimental** - True if a feature is considered experimental, such as being non-standard or part of a non-ratified spec.
  - **standardized** - True if a feature is described in a standards-track spec, regardless of the spec's maturity.
  - **stable** - True if a feature is considered suitable for production websites.
  - **obsolete** - True if a feature should not be used in new development.
  - **name** (*canonical or localized*) - Feature name. If the name is the code used by a developer, then the value is a string, and should be wrapped in a `<code>` block when displayed. If the name is a description of the feature, then the value is the available translations, including at least an `en` translation, and may include HTML markup. For example, `display` and `display: none` are canonical names for the CSS `display` property and one of the values for that property, while `Basic support`, `<code>none, inline</code>` and `<code>block</code>`, and `CSS Properties` are non-canonical names that should be translated.
- **links**
  - **sections** (*many*) - Associated [sections](#). Order can be changed by the user.
  - **supports** (*many*) - Associated [supports](#), Order is in ID order, changes are ignored.
  - **parent** (*one or null*) - The feature one level up, or null if top-level. Can be changed by user.
  - **children** (*many*) - The features that have this feature as parent, in display order. Can be an empty list, for “leaf” features. Can be re-ordered by the user.
  - **history\_current** (*one*) - Current `historical_features`. User can set to a valid **history** to revert to that version.
  - **history** (*many*) - Associated `historical_features`, in time order (most recent first). Changes are ignored.

To get a specific **feature** (in this case, a leaf feature with a translated name):

```
GET /api/v1/features/12 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```

{
  "features": {
    "id": "12",
    "slug": "web-css-transform-three-value-syntax",
    "mdn_uri": null,
    "experimental": false,
    "standardized": true,
    "stable": true,
    "obsolete": false,
    "name": {
      "en": "Three-value syntax",
      "es": "Sintaxis con tres valores",
      "ja": "3-"
    },
  },
  "links": {
    "sections": [],
    "supports": [
      "20",
      "21",
      "22",
      "23",
      "24",
      "25",
      "26"
    ],
    "parent": "10",
    "children": [],
    "history_current": "12",
    "history": [
      "12"
    ]
  }
},
"links": {
  "features.sections": {
    "type": "sections",
    "href": "https://browsercompat.org/api/v1/sections/{features.sections}"
  },
  "features.supports": {
    "type": "supports",
    "href": "https://browsercompat.org/api/v1/supports/{features.supports}"
  },
  "features.parent": {
    "type": "features",
    "href": "https://browsercompat.org/api/v1/features/{features.parent}"
  },
  "features.children": {
    "type": "features",
    "href": "https://browsercompat.org/api/v1/features/{features.children}"
  },
  "features.history_current": {
    "type": "historical_features",
    "href": "https://browsercompat.org/api/v1/historical_features/{features.history_current}"
  },
  "features.history": {
    "type": "historical_features",
    "href": "https://browsercompat.org/api/v1/historical_features/{features.history}"
  }
}

```



```
}
}
```

Here's an example of a branch feature with a canonical name (the parent of the previous example):

```
GET /api/v1/features/10 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "features": {
    "id": "10",
    "slug": "web-css-transform-origin",
    "mdn_uri": {
      "en": "https://developer.mozilla.org/en-US/docs/Web/CSS/transform-origin",
      "es": "https://developer.mozilla.org/es/docs/Web/CSS/transform-origin",
      "fr": "https://developer.mozilla.org/fr/docs/Web/CSS/transform-origin",
      "ja": "https://developer.mozilla.org/ja/docs/Web/CSS/transform-origin"
    },
    "experimental": false,
    "standardized": true,
    "stable": true,
    "obsolete": false,
    "name": "transform-origin",
    "links": {
      "sections": [
        "4"
      ],
      "supports": [],
      "parent": "2",
      "children": [
        "11",
        "12"
      ],
      "history_current": "10",
      "history": [
        "10"
      ]
    }
  },
  "links": {
    "features.sections": {
      "type": "sections",
      "href": "https://browsercompat.org/api/v1/sections/{features.sections}"
    },
    "features.supports": {
      "type": "supports",
      "href": "https://browsercompat.org/api/v1/supports/{features.supports}"
    },
    "features.parent": {
      "type": "features",
      "href": "https://browsercompat.org/api/v1/features/{features.parent}"
    },
    "features.children": {
```

```
    "type": "features",
    "href": "https://browsercompat.org/api/v1/features/{features.children}"
  },
  "features.history_current": {
    "type": "historical_features",
    "href": "https://browsercompat.org/api/v1/historical_features/{features.history_current}"
  },
  "features.history": {
    "type": "historical_features",
    "href": "https://browsercompat.org/api/v1/historical_features/{features.history}"
  }
}
```

## Supports

A **support** is an assertion that a particular Version of a Browser supports (or does not support) a feature.

The **support** representation includes:

- **attributes**

- **id** (*server selected*) - Database ID
- **support** - Assertion of support of the *version* for the *feature*, one of "yes", "no", "partial", or "unknown"
- **prefix** - Prefix used to enable support, such as "moz"
- **prefix\_mandatory** - True if the prefix is required
- **alternate\_name** - An alternate name associated with this feature, such as "RTCPeerConnectionIdentityEvent"
- **alternate\_name\_mandatory** - True if the alternate name is required
- **requires\_config** - A configuration string required to enable the feature, such as "media.peerconnection.enabled=on"
- **default\_config** - The configuration string in the shipping browser, such as "media.peerconnection.enabled=off"
- **protected** - True if the feature requires additional steps to enable in order to protect the user's security or privacy, such as geolocation and the Bluetooth API.
- **note** (*localized*) - Note on support, designed for display after a compatibility table, can contain HTML

- **links**

- **version** (*one*) - The associated *version*. Cannot be changed by the user after creation.
- **feature** (*one*) - The associated *feature*. Cannot be changed by the user after creation. The version and feature combo must be unique.
- **history\_current** (*one*) - Current *historical\_supports*. Can be changed to a valid **history** to revert to that version.
- **history** (*many*) - Associated *historical\_supports* in time order (most recent first). Changes are ignored.

To get a single **support**:

```
GET /api/v1/supports/22 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "supports": {
    "id": "22",
    "support": "yes",
    "prefix": null,
    "prefix_mandatory": false,
    "alternate_name": null,
    "alternate_mandatory": false,
    "requires_config": null,
    "default_config": null,
    "protected": false,
    "note": null,
    "links": {
      "version": "21",
      "feature": "12",
      "history_current": "22",
      "history": [
        "22"
      ]
    }
  },
  "links": {
    "supports.version": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{supports.version}"
    },
    "supports.feature": {
      "type": "features",
      "href": "https://browsercompat.org/api/v1/features/{supports.feature}"
    },
    "supports.history_current": {
      "type": "historical_supports",
      "href": "https://browsercompat.org/api/v1/historical_supports/{supports.history_current}"
    },
    "supports.history": {
      "type": "historical_supports",
      "href": "https://browsercompat.org/api/v1/historical_supports/{supports.history}"
    }
  }
}
```

## Specifications

A **specification** is a standards document that specifies a web technology.

The **specification** representation includes:

- **attributes**

- **id** (*server selected*) - Database ID
- **slug** - Unique, human-friendly key
- **mdn\_key** - Key used in the KumaScript macros `SpecName` and `Spec2`.
- **name** (*localized*) - Specification name
- **uri** (*localized*) - Specification URI, without subpath and anchor

• **links**

- **maturity** (*one*) - Associated *maturity*. Can be changed by the user.
- **sections** (*many*) - Associated *sections*. The order can be changed by the user.
- **history\_current** (*one*) - Current *historical\_specifications*. Can be changed to a valid **history** to revert to that version.
- **history** (*many*) - Associated *historical\_specifications* in time order (most recent first). Changes are ignored.

To get a single **specification**:

```
GET /api/v1/specifications/2 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "specifications": {
    "id": "2",
    "slug": "css2_1",
    "mdn_key": "CSS2.1",
    "name": {
      "en": "CSS Level&nbsp;2 (Revision&nbsp;1)"
    },
    "uri": {
      "en": "http://www.w3.org/TR/CSS2/"
    },
    "links": {
      "maturity": "1",
      "sections": [
        "2"
      ],
      "history_current": "2",
      "history": [
        "2"
      ]
    }
  },
  "links": {
    "specifications.maturity": {
      "type": "maturities",
      "href": "https://browsercompat.org/api/v1/maturities/{specifications.maturity}"
    },
    "specifications.sections": {
      "type": "sections",

```

```

    "href": "https://browsercompat.org/api/v1/sections/{specifications.sections}",
  },
  "specifications.history_current": {
    "type": "historical_specifications",
    "href": "https://browsercompat.org/api/v1/historical_specifications/{specifications.history_current}",
  },
  "specifications.history": {
    "type": "historical_specifications",
    "href": "https://browsercompat.org/api/v1/historical_specifications/{specifications.history}",
  }
}

```

## Sections

*Note: bug 1216786 is proposing splitting Sections into Sections and References.*

A **section** refers to a specific area of a *specification* document.

The **section** representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **number** (*optional, localized*) - The section number
  - **name** (*localized*) - Section name
  - **subpath** (*localized, optional*) - A subpage (possibly with an #anchor) to get to the subsection in the doc. Can be empty string.
  - **note** (*localized, optional*) - Notes for this section
- **links**
  - **specification** (*one*) - The *specification*. Can be changed by the user.
  - **features** (*many*) - The associated *features*. In ID order, changes are ignored.
  - **history\_current** (*one*) - Current *historical\_sections*. Can be changed to a valid **history** to revert to that version.
  - **history** (*many*) - Associated *historical\_sections* in time order (most recent first). Changes are ignored.

To get a single **section**:

```

GET /api/v1/sections/3 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json

```

A sample response is:

```

HTTP/1.1 200 OK
Content-Type: application/vnd.api+json

```

```

{
  "sections": {
    "id": "3",
    "number": {
      "en": "16"
    }
  },
}

```

```

    "name": {
      "en": "The float property"
    },
    "subpath": {
      "en": "#the-float-property"
    },
    "note": {
      "en": "Lots of new values, not all clearly defined yet. Any differences in behavior unre..."
    },
    "links": {
      "specification": "3",
      "features": [
        "5"
      ],
      "history_current": "3",
      "history": [
        "3"
      ]
    }
  },
  "links": {
    "sections.specification": {
      "type": "specifications",
      "href": "https://browsercompat.org/api/v1/specifications/{sections.specification}"
    },
    "sections.features": {
      "type": "features",
      "href": "https://browsercompat.org/api/v1/features/{sections.features}"
    },
    "sections.history_current": {
      "type": "historical_sections",
      "href": "https://browsercompat.org/api/v1/historical_sections/{sections.history_current}"
    },
    "sections.history": {
      "type": "historical_sections",
      "href": "https://browsercompat.org/api/v1/historical_sections/{sections.history}"
    }
  }
}

```

## Maturities

A **maturity** refers to the maturity of a *specification* document.

The **maturity** representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **slug** - A human-friendly identifier for this maturity. When applicable, it match the key in the KumaScript macro `Spec2`
  - **name** (*localized*) - Status name
- **links**
  - **specifications** (*many*) - Associated *specifications*. In ID order, changes are ignored.

- **history\_current** (*one*) - Current historical\_maturities. Can be changed to a valid **history** to revert to that version.
- **history** (*many*) - Associated historical\_maturities in time order (most recent first). Changes are ignored.

To get a single **maturity**:

```
GET /api/v1/maturities/1 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "maturities": {
    "id": "1",
    "slug": "REC",
    "name": {
      "en": "Recommendation",
      "de": "Empfehlung",
      "ja": "",
      "ru": ""
    },
    "links": {
      "specifications": [
        "1",
        "2"
      ],
      "history_current": "1",
      "history": [
        "1"
      ]
    }
  },
  "links": {
    "maturities.specifications": {
      "type": "specifications",
      "href": "https://browsercompat.org/api/v1/specifications/{maturities.specifications}"
    },
    "maturities.history_current": {
      "type": "historical_maturities",
      "href": "https://browsercompat.org/api/v1/historical_maturities/{maturities.history_current}"
    },
    "maturities.history": {
      "type": "historical_maturities",
      "href": "https://browsercompat.org/api/v1/historical_maturities/{maturities.history}"
    }
  }
}
```

### 2.4.3 Change Control Resources

Change Control Resources help manage changes to resources.

### Users

A **user** represents a person or process that creates, changes, or deletes a resource.

The representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **username** - The user's email or ID
  - **created** (*server selected*) - Time that the account was created, in ISO 8601 format.
  - **agreement** - The version of the contribution agreement the user has accepted. "0" for not agreed, "1" for first version, etc.
  - **permissions** - A list of permissions. Permissions include "change-resource" (add or change any resource except *users* or history resources), "delete-resource" (delete any resource) "import-mdn" (setup import of an MDN page)
- **links**
  - **changesets** (*many*) - Associated *changesets*, in ID order, changes are ignored.

To get a single **user** representation:

```
GET /api/v1/users/1 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "users": {
    "id": "1",
    "username": "user",
    "created": "2015-04-20T18:06:48.567514Z",
    "agreement": 0,
    "permissions": [
      "change-resource",
      "delete-resource"
    ],
    "links": {
      "changesets": []
    }
  },
  "links": {
    "users.changesets": {
      "type": "changesets",
      "href": "https://browsercompat.org/api/v1/changesets/{users.changesets}"
    }
  }
}
```

*Note:* bug 1216786 proposes this method for retrieving the authenticated user:

```
GET /api/v1/users/me HTTP/1.1
Host: browsersupports.org
Accept: application/vnd.api+json
```



## Changesets

A **changeset** collects history resources into a logical unit, allowing for faster reversions and better history display. The **changeset** can be auto-created through a POST, PUT, or DELETE to a resource, or it can be created independently and specified by adding `changeset=<ID>` URI parameter (i.e., PUT `/browsers/15?changeset=73`).

The representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **created** (*server selected*) - When the changeset was created, in ISO 8601 format.
  - **modified** (*server selected*) - When the changeset was last modified, in ISO 8601 format.
  - **target\_resource\_type** (*write-once, optional*) - The name of the primary resource for this changeset, for example “browsers”, “versions”, etc.
  - **target\_resource\_id** (*write-once, optional*) - The ID of the primary resource for this changeset.
  - **closed** - True if the changeset is closed to new changes. Auto-created changesets are auto-closed, and cache invalidation is delayed until manually created changesets are closed.
- **links**
  - **user** (*one*) - The user who initiated this changeset, can not be changed.
  - **historical\_browsers** (*many*) - Associated historical\_browsers, in ID order, changes are ignored.
  - **historical\_features** (*many*) - Associated historical\_features, in ID order, changes are ignored.
  - **historical\_maturities** (*many*) - Associated historical\_maturities, in ID order, changes are ignored.
  - **historical\_sections** (*many*) - Associated historical\_sections, in ID order, changes are ignored.
  - **historical\_specificationss** (*many*) - Associated historical\_specificationss, in ID order, changes are ignored.
  - **historical\_supports** (*many*) - Associated historical\_supports, in ID order, changes are ignored.
  - **historical\_versions** (*many*) - Associated historical\_versions, in ID order, changes are ignored.

To get a single **changeset** representation:

```
GET /api/v1/changesets/2 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "changesets": {
    "id": "2",
    "created": "2015-04-20T18:22:47.046692Z",
    "modified": "2015-04-20T18:22:47.056433Z",
    "closed": true,
    "target_resource_type": null,
    "target_resource_id": null,
    "links": {
      "user": "1",
      "historical_browsers": [
```

```
        "16"
      ],
      "historical_features": [],
      "historical_maturities": [],
      "historical_sections": [],
      "historical_specifications": [],
      "historical_supports": [],
      "historical_versions": []
    }
  },
  "links": {
    "changesets.user": {
      "type": "users",
      "href": "https://browsercompat.org/api/v1/users/{changesets.user}"
    },
    "changesets.historical_browsers": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{changesets.historical_browsers}"
    },
    "changesets.historical_features": {
      "type": "historical_features",
      "href": "https://browsercompat.org/api/v1/historical_features/{changesets.historical_features}"
    },
    "changesets.historical_maturities": {
      "type": "historical_maturities",
      "href": "https://browsercompat.org/api/v1/historical_maturities/{changesets.historical_maturities}"
    },
    "changesets.historical_sections": {
      "type": "historical_sections",
      "href": "https://browsercompat.org/api/v1/historical_sections/{changesets.historical_sections}"
    },
    "changesets.historical_specifications": {
      "type": "historical_specifications",
      "href": "https://browsercompat.org/api/v1/historical_specifications/{changesets.historical_specifications}"
    },
    "changesets.historical_supports": {
      "type": "historical_supports",
      "href": "https://browsercompat.org/api/v1/historical_supports/{changesets.historical_supports}"
    },
    "changesets.historical_versions": {
      "type": "historical_versions",
      "href": "https://browsercompat.org/api/v1/historical_versions/{changesets.historical_versions}"
    }
  }
}
```

## 2.4.4 History Resources

History Resources are created when a Resource is created, updated, or deleted. By navigating the history chain, a caller can see the changes of a resource over time.

All history representations are similar, so one example should be enough to determine the pattern.

## Historical Browsers

A **historical\_browser** resource represents the state of a browser at a point in time, and who is responsible for that state. The representation includes:

- **attributes**
  - **id** (*server selected*) - Database ID
  - **date** (*server selected*) - The time of this change in ISO 8601
  - **event** (*server selected*) - The type of event, one of "created", "changed", or "deleted"
  - **browsers** - The **browsers** representation at this point in time
- **links**
  - **browser** (*one*) - Associated browser, can not be changed
  - **changeset** (*one*) - Associated changeset, can not be changed.

To get a single **historical\_browsers** representation:

```
GET /api/v1/historical_browsers/7 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "historical_browsers": {
    "id": "7",
    "date": "2015-04-20T18:44:09.905824Z",
    "event": "created",
    "browsers": {
      "id": "7",
      "slug": "firefox_desktop",
      "name": {
        "en": "Firefox for Desktop"
      },
      "note": null,
      "links": {
        "versions": [],
        "history_current": "7"
      }
    },
    "links": {
      "changeset": "1",
      "browser": "7"
    }
  },
  "links": {
    "historical_browsers.changeset": {
      "type": "changesets",
      "href": "https://browsercompat.org/api/v1/changesets/{historical_browsers.changeset}"
    },
    "historical_browsers.browser": {
      "type": "browsers",
      "href": "https://browsercompat.org/api/v1/browsers/{historical_browsers.browser}"
    }
  }
}
```

```
}  
  }  
}
```

### Historical Versions

A **historical\_versions** resource represents the state of a version at a point in time, and who is responsible for that representation. See *historical\_browsers* and *versions* for an idea of the representation.

### Historical Features

A **historical\_features** resource represents the state of a feature at a point in time, and who is responsible for that representation. See *historical\_browsers* and *features* for an idea of the representation.

### Historical Sections

A **historical\_sections** resource represents the state of a section at a point in time, and who is responsible for that representation. See *historical\_browsers* and *sections* for an idea of the representation.

### Historical Specifications

A **historical\_specifications** resource represents the state of a specification at a point in time, and who is responsible for that representation. See *historical\_browsers* and *specifications* for an idea of the representation.

### Historical Supports

A **historical\_supports** resource represents the state of a support at a point in time, and who is responsible for that representation. See *historical\_browsers* and *supports* for an idea of the representation.

### Historical Maturities

A **historical\_maturities** resource represents the state of a maturity at a point in time, and who is responsible for that representation. See *historical\_browsers* and *maturities* for an idea of the representation.

## 2.4.5 Views

A **View** is a combination of resources for a particular presentation. It is suitable for anonymous viewing of content. It uses the flexibility of the JSON API specification to include a basket of related resources in a response, but doesn't use the official method of [Inclusion of Related Resources](#).

### View a Feature

This view collects the data for a feature, including the related resources needed to display it on MDN.

Here is a simple example, the view for the CSS property `float`:

```
GET /api/v1/view_features/5 HTTP/1.1  
Host: browsercompat.org  
Accept: application/vnd.api+json
```

A sample response is:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.api+json
```

```
{
  "features": {
    "id": "5",
    "slug": "web-css-float",
    "mdn_uri": {
      "en": "https://developer.mozilla.org/en-US/docs/Web/CSS/float",
      "de": "https://developer.mozilla.org/de/docs/Web/CSS/float",
      "es": "https://developer.mozilla.org/es/docs/Web/CSS/float",
      "ja": "https://developer.mozilla.org/ja/docs/Web/CSS/float",
      "ru": "https://developer.mozilla.org/ru/docs/Web/CSS/float"
    },
    "experimental": false,
    "standardized": true,
    "stable": true,
    "obsolete": false,
    "name": "float",
    "links": {
      "sections": [
        "1",
        "2",
        "3"
      ],
      "supports": [],
      "parent": "2",
      "children": [
        "6"
      ],
      "history_current": "5",
      "history": [
        "5"
      ]
    }
  },
  "links": {
    "features.sections": {
      "type": "sections",
      "href": "https://browsercompat.org/api/v1/sections/{features.sections}"
    },
    "features.supports": {
      "type": "supports",
      "href": "https://browsercompat.org/api/v1/supports/{features.supports}"
    },
    "features.parent": {
      "type": "features",
      "href": "https://browsercompat.org/api/v1/features/{features.parent}"
    },
    "features.children": {
      "type": "features",
      "href": "https://browsercompat.org/api/v1/features/{features.children}"
    },
    "features.history_current": {
      "type": "historical_features",
      "href": "https://browsercompat.org/api/v1/historical_features/{features.history_current}"
    }
  }
}
```

```
"features.history": {
  "type": "historical_features",
  "href": "https://browsercompat.org/api/v1/historical_features/{features.history}"
},
"browsers.history": {
  "type": "historical_browsers",
  "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history}"
},
"browsers.history_current": {
  "type": "historical_browsers",
  "href": "https://browsercompat.org/api/v1/historical_browsers/{browsers.history_current}"
},
"versions.browser": {
  "type": "browsers",
  "href": "https://browsercompat.org/api/v1/browsers/{versions.browser}"
},
"versions.history": {
  "type": "historical_versions",
  "href": "https://browsercompat.org/api/v1/historical_versions/{versions.history}"
},
"versions.history_current": {
  "type": "historical_versions",
  "href": "https://browsercompat.org/api/v1/historical_versions/{versions.history_current}"
},
"supports.version": {
  "type": "versions",
  "href": "https://browsercompat.org/api/v1/versions/{supports.version}"
},
"supports.feature": {
  "type": "features",
  "href": "https://browsercompat.org/api/v1/features/{supports.feature}"
},
"supports.history_current": {
  "type": "historical_supports",
  "href": "https://browsercompat.org/api/v1/historical_supports/{supports.history_current}"
},
"supports.history": {
  "type": "historical_supports",
  "href": "https://browsercompat.org/api/v1/historical_supports/{supports.history}"
},
"maturities.history_current": {
  "type": "historical_maturities",
  "href": "https://browsercompat.org/api/v1/historical_maturities/{maturities.history_current}"
},
"maturities.history": {
  "type": "historical_maturities",
  "href": "https://browsercompat.org/api/v1/historical_maturities/{maturities.history}"
},
"specifications.maturity": {
  "type": "maturities",
  "href": "https://browsercompat.org/api/v1/maturities/{specifications.maturity}"
},
"specifications.history_current": {
  "type": "historical_specifications",
  "href": "https://browsercompat.org/api/v1/historical_specifications/{specifications.history_current}"
},
"specifications.history": {
  "type": "historical_specifications",
```

```

    "href": "https://browsercompat.org/api/v1/historical_specifications/{specifications.history_current}",
  },
  "sections.specification": {
    "type": "specifications",
    "href": "https://browsercompat.org/api/v1/specifications/{sections.specification}"
  },
  "sections.history_current": {
    "type": "historical_sections",
    "href": "https://browsercompat.org/api/v1/historical_sections/{sections.history_current}"
  },
  "sections.history": {
    "type": "historical_sections",
    "href": "https://browsercompat.org/api/v1/historical_sections/{sections.history}"
  }
},
"linked": {
  "browsers": [
    {
      "id": "1",
      "slug": "android",
      "name": {
        "en": "Android Browser"
      },
      "note": null,
      "links": {
        "history": [
          "1"
        ],
        "history_current": "1"
      }
    },
    {
      "id": "2",
      "slug": "android_webview",
      "name": {
        "en": "Android WebView"
      },
      "note": null,
      "links": {
        "history": [
          "2"
        ],
        "history_current": "2"
      }
    },
    {
      "id": "4",
      "slug": "chrome_desktop",
      "name": {
        "en": "Chrome for Desktop"
      },
      "note": null,
      "links": {
        "history": [
          "4"
        ],
        "history_current": "4"
      }
    }
  ]
}

```

```
    },
    {
      "id": "6",
      "slug": "firefox_android",
      "name": {
        "en": "Firefox for Android"
      },
      "note": null,
      "links": {
        "history": [
          "6"
        ],
        "history_current": "6"
      }
    },
    {
      "id": "7",
      "slug": "firefox_desktop",
      "name": {
        "en": "Firefox for Desktop"
      },
      "note": null,
      "links": {
        "history": [
          "7"
        ],
        "history_current": "7"
      }
    },
    {
      "id": "9",
      "slug": "ie_desktop",
      "name": {
        "en": "Internet Explorer for Desktop"
      },
      "note": null,
      "links": {
        "history": [
          "22",
          "21",
          "20",
          "19",
          "9"
        ],
        "history_current": "22"
      }
    },
    {
      "id": "10",
      "slug": "ie_mobile",
      "name": {
        "en": "Internet Explorer Mobile"
      },
      "note": null,
      "links": {
        "history": [
          "10"
        ],

```



```

        "history_current": "10"
    }
},
{
    "id": "11",
    "slug": "opera_desktop",
    "name": {
        "en": "Opera for Desktop"
    },
    "note": null,
    "links": {
        "history": [
            "11"
        ],
        "history_current": "11"
    }
},
{
    "id": "14",
    "slug": "safari_desktop",
    "name": {
        "en": "Safari for Desktop"
    },
    "note": null,
    "links": {
        "history": [
            "14"
        ],
        "history_current": "14"
    }
},
{
    "id": "15",
    "slug": "safari_ios",
    "name": {
        "en": "Safari for iOS"
    },
    "note": null,
    "links": {
        "history": [
            "15"
        ],
        "history_current": "15"
    }
}
],
"versions": [
    {
        "id": "2",
        "version": "1.0",
        "release_day": null,
        "retirement_day": null,
        "status": "unknown",
        "release_notes_uri": null,
        "note": null,
        "order": 1,
        "links": {
            "browser": "1",

```

```
        "history": [
            "2"
        ],
        "history_current": "2"
    }
},
{
    "id": "4",
    "version": "current",
    "release_day": null,
    "retirement_day": null,
    "status": "current",
    "release_notes_uri": null,
    "note": null,
    "order": 0,
    "links": {
        "browser": "2",
        "history": [
            "4"
        ],
        "history_current": "4"
    }
},
{
    "id": "9",
    "version": "1.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 1,
    "links": {
        "browser": "4",
        "history": [
            "9"
        ],
        "history_current": "9"
    }
},
{
    "id": "15",
    "version": "1.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 1,
    "links": {
        "browser": "6",
        "history": [
            "15"
        ],
        "history_current": "15"
    }
},
{
```

```
    "id": "18",
    "version": "1.0",
    "release_day": "2004-11-09",
    "retirement_day": "2005-11-29",
    "status": "retired",
    "release_notes_uri": null,
    "note": null,
    "order": 1,
    "links": {
      "browser": "7",
      "history": [
        "18"
      ],
      "history_current": "18"
    }
  },
  {
    "id": "26",
    "version": "4.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 1,
    "links": {
      "browser": "9",
      "history": [
        "26"
      ],
      "history_current": "26"
    }
  },
  {
    "id": "32",
    "version": "6.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 2,
    "links": {
      "browser": "10",
      "history": [
        "32"
      ],
      "history_current": "32"
    }
  },
  {
    "id": "34",
    "version": "7.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
```

```
    "order": 1,
    "links": {
      "browser": "11",
      "history": [
        "34"
      ],
      "history_current": "34"
    }
  },
  {
    "id": "42",
    "version": "1.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 1,
    "links": {
      "browser": "14",
      "history": [
        "42"
      ],
      "history_current": "42"
    }
  },
  {
    "id": "44",
    "version": "5.1",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 3,
    "links": {
      "browser": "14",
      "history": [
        "44"
      ],
      "history_current": "44"
    }
  },
  {
    "id": "46",
    "version": "1.0",
    "release_day": null,
    "retirement_day": null,
    "status": "unknown",
    "release_notes_uri": null,
    "note": null,
    "order": 1,
    "links": {
      "browser": "15",
      "history": [
        "46"
      ],
      "history_current": "46"
    }
  }
}
```

```
    }
  },
  "supports": [
    {
      "id": "1",
      "support": "yes",
      "prefix": null,
      "prefix_mandatory": false,
      "alternate_name": null,
      "alternate_mandatory": false,
      "requires_config": null,
      "default_config": null,
      "protected": false,
      "note": null,
      "links": {
        "version": "2",
        "feature": "6",
        "history_current": "1",
        "history": [
          "1"
        ]
      }
    },
    {
      "id": "2",
      "support": "yes",
      "prefix": null,
      "prefix_mandatory": false,
      "alternate_name": null,
      "alternate_mandatory": false,
      "requires_config": null,
      "default_config": null,
      "protected": false,
      "note": null,
      "links": {
        "version": "9",
        "feature": "6",
        "history_current": "2",
        "history": [
          "2"
        ]
      }
    },
    {
      "id": "3",
      "support": "yes",
      "prefix": null,
      "prefix_mandatory": false,
      "alternate_name": null,
      "alternate_mandatory": false,
      "requires_config": null,
      "default_config": null,
      "protected": false,
      "note": null,
      "links": {
        "version": "15",
        "feature": "6",
```

```
        "history_current": "3",
        "history": [
            "3"
        ]
    },
    {
        "id": "4",
        "support": "yes",
        "prefix": null,
        "prefix_mandatory": false,
        "alternate_name": null,
        "alternate_mandatory": false,
        "requires_config": null,
        "default_config": null,
        "protected": false,
        "note": null,
        "links": {
            "version": "18",
            "feature": "6",
            "history_current": "4",
            "history": [
                "4"
            ]
        }
    },
    {
        "id": "5",
        "support": "yes",
        "prefix": null,
        "prefix_mandatory": false,
        "alternate_name": null,
        "alternate_mandatory": false,
        "requires_config": null,
        "default_config": null,
        "protected": false,
        "note": null,
        "links": {
            "version": "26",
            "feature": "6",
            "history_current": "5",
            "history": [
                "5"
            ]
        }
    },
    {
        "id": "6",
        "support": "yes",
        "prefix": null,
        "prefix_mandatory": false,
        "alternate_name": null,
        "alternate_mandatory": false,
        "requires_config": null,
        "default_config": null,
        "protected": false,
        "note": null,
        "links": {
```

```

        "version": "32",
        "feature": "6",
        "history_current": "6",
        "history": [
            "6"
        ]
    },
    {
        "id": "7",
        "support": "yes",
        "prefix": null,
        "prefix_mandatory": false,
        "alternate_name": null,
        "alternate_mandatory": false,
        "requires_config": null,
        "default_config": null,
        "protected": false,
        "note": null,
        "links": {
            "version": "34",
            "feature": "6",
            "history_current": "7",
            "history": [
                "7"
            ]
        }
    },
    {
        "id": "8",
        "support": "yes",
        "prefix": null,
        "prefix_mandatory": false,
        "alternate_name": null,
        "alternate_mandatory": false,
        "requires_config": null,
        "default_config": null,
        "protected": false,
        "note": null,
        "links": {
            "version": "42",
            "feature": "6",
            "history_current": "8",
            "history": [
                "8"
            ]
        }
    },
    {
        "id": "9",
        "support": "yes",
        "prefix": null,
        "prefix_mandatory": false,
        "alternate_name": null,
        "alternate_mandatory": false,
        "requires_config": null,
        "default_config": null,
        "protected": false,
    }
}

```

```
    "note": null,
    "links": {
      "version": "44",
      "feature": "6",
      "history_current": "9",
      "history": [
        "9"
      ]
    }
  },
  {
    "id": "10",
    "support": "yes",
    "prefix": null,
    "prefix_mandatory": false,
    "alternate_name": null,
    "alternate_mandatory": false,
    "requires_config": null,
    "default_config": null,
    "protected": false,
    "note": null,
    "links": {
      "version": "46",
      "feature": "6",
      "history_current": "10",
      "history": [
        "10"
      ]
    }
  },
  {
    "id": "27",
    "support": "yes",
    "prefix": null,
    "prefix_mandatory": false,
    "alternate_name": null,
    "alternate_mandatory": false,
    "requires_config": null,
    "default_config": null,
    "protected": false,
    "note": null,
    "links": {
      "version": "4",
      "feature": "6",
      "history_current": "27",
      "history": [
        "27"
      ]
    }
  }
],
"maturities": [
  {
    "id": "1",
    "slug": "REC",
    "name": {
      "en": "Recommendation",
      "de": "Empfehlung",
    }
  }
]
```



```

        "ja": "",
        "ru": ""
    },
    "links": {
        "history_current": "1",
        "history": [
            "1"
        ]
    }
},
{
    "id": "2",
    "slug": "WD",
    "name": {
        "en": "Working Draft",
        "de": "Arbeitsentwurf",
        "ja": "",
        "ru": ""
    },
    "links": {
        "history_current": "2",
        "history": [
            "2"
        ]
    }
}
],
"specifications": [
    {
        "id": "1",
        "slug": "css1",
        "mdn_key": "CSS1",
        "name": {
            "en": "CSS Level&nbsp;1"
        },
        "uri": {
            "en": "http://www.w3.org/TR/CSS1/"
        },
        "links": {
            "maturity": "1",
            "history_current": "1",
            "history": [
                "1"
            ]
        }
    },
    {
        "id": "2",
        "slug": "css2_1",
        "mdn_key": "CSS2.1",
        "name": {
            "en": "CSS Level&nbsp;2 (Revision&nbsp;1)"
        },
        "uri": {
            "en": "http://www.w3.org/TR/CSS2/"
        },
        "links": {
            "maturity": "1",

```

```
        "history_current": "2",
        "history": [
            "2"
        ]
    },
    {
        "id": "3",
        "slug": "css3_box",
        "mdn_key": "CSS3 Box",
        "name": {
            "en": "CSS Basic Box Model"
        },
        "uri": {
            "en": "http://dev.w3.org/csswg/css3-box/"
        },
        "links": {
            "maturity": "2",
            "history_current": "3",
            "history": [
                "3"
            ]
        }
    }
],
"sections": [
    {
        "id": "1",
        "number": {
            "en": "5.5.25"
        },
        "name": {
            "en": "'float'"
        },
        "subpath": {
            "en": "#float"
        },
        "note": {
            "en": "Initial definition."
        },
        "links": {
            "specification": "1",
            "history_current": "1",
            "history": [
                "1"
            ]
        }
    },
    {
        "id": "2",
        "number": {
            "en": "9.5.1"
        },
        "name": {
            "en": "Positioning the float: the 'float' property"
        },
        "subpath": {
            "en": "visuren.html#float-position"
        }
    }
]
```

```

    },
    "note": {
      "en": "No change."
    },
    "links": {
      "specification": "2",
      "history_current": "2",
      "history": [
        "2"
      ]
    }
  },
  {
    "id": "3",
    "number": {
      "en": "16"
    },
    "name": {
      "en": "The float property"
    },
    "subpath": {
      "en": "#the-float-property"
    },
    "note": {
      "en": "Lots of new values, not all clearly defined yet. Any differences in behavior"
    },
    "links": {
      "specification": "3",
      "history_current": "3",
      "history": [
        "3"
      ]
    }
  }
],
"features": [
  {
    "id": "6",
    "slug": "web-css-float_basic-support",
    "mdn_uri": null,
    "experimental": false,
    "standardized": true,
    "stable": true,
    "obsolete": false,
    "name": {
      "en": "Basic Support",
      "ja": ""
    },
    "links": {
      "sections": [],
      "supports": [
        "1",
        "2",
        "3",
        "4",
        "5",
        "6",
        "7",

```

```
        "8",
        "9",
        "10",
        "27"
    ],
    "parent": "5",
    "children": [],
    "history_current": "6",
    "history": [
        "6"
    ]
}
}
]
},
"meta": {
    "compat_table": {
        "supports": {
            "5": {},
            "6": {
                "1": [
                    "1"
                ],
                "2": [
                    "27"
                ],
                "4": [
                    "2"
                ],
                "6": [
                    "3"
                ],
                "7": [
                    "4"
                ],
                "9": [
                    "5"
                ],
                "10": [
                    "6"
                ],
                "11": [
                    "7"
                ],
                "14": [
                    "8"
                ],
                "15": [
                    "10"
                ]
            }
        },
        "tabs": [
            {
                "name": {
                    "en": "Desktop Browsers"
                },
                "browsers": [
```

```

        "4",
        "7",
        "9",
        "11",
        "14"
    ]
},
{
    "name": {
        "en": "Mobile Browsers"
    },
    "browsers": [
        "1",
        "2",
        "6",
        "10",
        "15"
    ]
},
],
"child_pages": false,
"pagination": {
    "linked_features": {
        "previous": null,
        "next": null,
        "count": 1
    }
},
"languages": [
    "en",
    "de",
    "es",
    "ja",
    "ru"
],
"notes": {}
}
}

```

One way to use this representation is:

1. Parse into an in-memory object store,
2. **Create the “Specifications” section:**
  - (a) Add the Specifications header
  - (b) Create an HTML table with a header row “Specification”, “Status”, “Comment”
  - (c) **For each id in features.links.sections (["746", "421", "70"]):**
    - Add the first column: a link to specifications.uri.(lang or en) + sections.subpath.(lang or en), with link text specifications.name.(lang or en), with title based on sections.name.(lang or en) or feature.name.(lang or en).
    - Add the second column: A span with class “spec-” + maturities.slug, and the text maturities.name.(lang or en).
    - Add the third column: maturities.notes.(lang or en), or empty string

(d) Close the table, and add an edit button.

**3. Create the Browser Compatibility section:**

(a) Add The “Browser compatibility” header

(b) For each item in `meta.compat_table.tabs`, create a table with the proper name (“Desktop”, “Mobile”)

(c) For each browser id in `meta.compat-table.tabs.browsers`, add a column with the translated browser name.

(d) **For each feature in `features.features`:**

- Add the first column: the feature name. If it is a string, then wrap in `<code>`. Otherwise, use the best translation of `feature.name`, in a `lang=(lang)` block.
- Add any feature flags, such as an obsolete or experimental icon, based on the feature flags.
- **For each browser id in `meta.compat-table-important`:**
  - Get the important support IDs from `meta.compat-table-important.supports.<feature ID>.<browser ID>`
  - If null, then display “?”
  - If just one, display “<version>”, or “<support>”, depending on the defined attributes
  - If multiple, display as subcells
  - Add prefixes, alternate names, config, ank notes link as appropriate

(e) Close each table, add an edit button

(f) Add notes for displayed supports

This may be done by including the JSON in the page as sent over the wire, or loaded asynchronously, with the tables built after initial page load.

This can also be used by a “[caniuse](#)” table layout by ignoring the meta section and displaying all the included data. This will require more client-side processing to generate, or additional data in the `<meta>` section.

### Including Child Pages

By default, `view_feature` only includes *row children*, which are subfeatures that are represented as rows in the MDN table. These row children are identified by not having a value for `mdn_uri`.

There can also be *page children*, which are represented as more detailed page on MDN. For example, *Web/CSS* has the page child *Web/CSS/Display*. By default, these are not included, but can be included by setting the query parameter `child_pages=1`:

```
GET /api/v1/view_features/5?child_pages=1 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
```

### Updating Views with Changesets

Updating the page requires a sequence of requests. For example, if a user wants to change Chrome support for `<address>` from an unknown version to version 1, you’ll have to create the version for that version, then add the support for the support.

The first step is to create a changeset as an authenticated user:

```
POST /api/v1/changesets HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 107
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn
```

```
{
  "changesets": {
    "target_resource_type": "features",
    "target_resource_id": "5"
  }
}
```

A sample response is:

```
HTTP/1.1 201 CREATED
Content-Type: application/vnd.api+json
```

```
{
  "changesets": {
    "id": "36",
    "created": "2015-04-20T20:36:06.794827Z",
    "modified": "2015-04-20T20:36:06.795315Z",
    "closed": false,
    "target_resource_type": "features",
    "target_resource_id": 5,
    "links": {
      "user": "1",
      "historical_browsers": [],
      "historical_features": [],
      "historical_maturities": [],
      "historical_sections": [],
      "historical_specifications": [],
      "historical_supports": [],
      "historical_versions": []
    }
  },
  "links": {
    "changesets.user": {
      "type": "users",
      "href": "https://browsercompat.org/api/v1/users/{changesets.user}"
    },
    "changesets.historical_browsers": {
      "type": "historical_browsers",
      "href": "https://browsercompat.org/api/v1/historical_browsers/{changesets.historical_browsers}"
    },
    "changesets.historical_features": {
      "type": "historical_features",
      "href": "https://browsercompat.org/api/v1/historical_features/{changesets.historical_features}"
    },
    "changesets.historical_maturities": {
      "type": "historical_maturities",
      "href": "https://browsercompat.org/api/v1/historical_maturities/{changesets.historical_maturities}"
    }
  }
}
```

```
    },
    "changesets.historical_sections": {
      "type": "historical_sections",
      "href": "https://browsercompat.org/api/v1/historical_sections/{changesets.historical_sect",
    },
    "changesets.historical_specifications": {
      "type": "historical_specifications",
      "href": "https://browsercompat.org/api/v1/historical_specifications/{changesets.historica",
    },
    },
    "changesets.historical_supports": {
      "type": "historical_supports",
      "href": "https://browsercompat.org/api/v1/historical_supports/{changesets.historical_sup",
    },
    },
    "changesets.historical_versions": {
      "type": "historical_versions",
      "href": "https://browsercompat.org/api/v1/historical_versions/{changesets.historical_ver",
    }
  }
}
```

Next, use the changeset ID when creating the version:

```
POST /api/v1/versions?changeset=36 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 138
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn
```

```
{
  "versions": {
    "version": "2.0",
    "status": "retired",
    "links": {
      "browser": "3"
    }
  }
}
```

A sample response is:

```
HTTP/1.1 201 CREATED
Content-Type: application/vnd.api+json
```

```
{
  "versions": {
    "id": "50",
    "version": "2.0",
    "release_day": null,
    "retirement_day": null,
    "status": "retired",
    "release_notes_uri": null,
    "note": null,
    "order": 3,
  }
}
```



```

    "links": {
      "browser": "3",
      "supports": [],
      "history": [
        "53"
      ],
      "history_current": "53"
    },
  },
  "links": {
    "versions.browser": {
      "type": "browsers",
      "href": "https://browsercompat.org/api/v1/browsers/{versions.browser}"
    },
    "versions.supports": {
      "type": "supports",
      "href": "https://browsercompat.org/api/v1/supports/{versions.supports}"
    },
    "versions.history": {
      "type": "historical_versions",
      "href": "https://browsercompat.org/api/v1/historical_versions/{versions.history}"
    },
    "versions.history_current": {
      "type": "historical_versions",
      "href": "https://browsercompat.org/api/v1/historical_versions/{versions.history_current}"
    }
  }
}

```

Finally, create the support:

```

POST /api/v1/supports?changeset=36 HTTP/1.1
Host: browsercompat.org
Accept: application/vnd.api+json
Content-Length: 112
Content-Type: application/vnd.api+json
Cookie: csrftoken=p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn; sessionid=wurexa2wq416ftlvd5plesngwa28183h
X-Csrftoken: p7FqFyNp6hZS0FJYKyQxVmLrZILldjqn

```

```

{
  "supports": {
    "links": {
      "version": "50",
      "feature": "6"
    }
  }
}

```

A sample response is:

```

HTTP/1.1 201 CREATED
Content-Type: application/vnd.api+json

```

```

{
  "supports": {

```

```

    "id": "29",
    "support": "yes",
    "prefix": null,
    "prefix_mandatory": false,
    "alternate_name": null,
    "alternate_mandatory": false,
    "requires_config": null,
    "default_config": null,
    "protected": false,
    "note": null,
    "links": {
      "version": "50",
      "feature": "6",
      "history_current": "32",
      "history": [
        "32"
      ]
    }
  },
  "links": {
    "supports.version": {
      "type": "versions",
      "href": "https://browsercompat.org/api/v1/versions/{supports.version}"
    },
    "supports.feature": {
      "type": "features",
      "href": "https://browsercompat.org/api/v1/features/{supports.feature}"
    },
    "supports.history_current": {
      "type": "historical_supports",
      "href": "https://browsercompat.org/api/v1/historical_supports/{supports.history_current}"
    },
    "supports.history": {
      "type": "historical_supports",
      "href": "https://browsercompat.org/api/v1/historical_supports/{supports.history}"
    }
  }
}

```

The `historical_versions` and `historical_supports` resources will both refer to changeset 36, and this changeset is linked to feature 5, despite the fact that no changes were made to the feature. This will facilitate displaying a history of the compatibility tables, for the purpose of reviewing changes and reverting vandalism.

## Updating View with PUT

`view_features` supports PUT for bulk updates of support data. Here is a simple example that adds a new subfeature without support:

```

PUT /api/v1/view_features/html-element-address HTTP/1.1
Host: browsersupports.org
Content-Type: application/vnd.api+json
Authorization: Bearer mF_9.B5f-4.1JqM

```

```

{
  "features": {

```

```

    "id": "816",
    "slug": "html-element-address",
    "mdn_uri": {
      "en": "https://developer.mozilla.org/en-US/docs/Web/HTML/Element/address"
    },
    "experimental": false,
    "standardized": true,
    "stable": true,
    "obsolete": false,
    "name": "address",
    "links": {
      "sections": ["746", "421", "70"],
      "supports": [],
      "parent": "800",
      "children": ["191"],
      "history_current": "216",
      "history": ["216"]
    }
  },
  "linked": {
    "features": [
      {
        "id": "_New Subfeature",
        "slug": "html-address-new-subfeature",
        "name": {
          "en": "New Subfeature"
        },
        "links": {
          "parent": "816"
        }
      }
    ]
  }
}

```

The response is the feature view with new and updated items, or an error response.

This is a trivial use case, which would be better implemented by creating the feature directly, but it can be extended to bulk updates of existing feature views, or for first-time importing of subfeatures and support data. It has some quirks:

- New items should be identified with an ID starting with an underscore (\_). Relations to new items should use the underscored IDs.
- Only feature, support, and section resources can be added or updated. Features must be the target feature or a descendant, and supports and sections are restricted to those features.
- Deletions are not supported.
- Other resources (browsers, versions, etc) can not be added or changed. This includes adding links to new resources.

Once the MDN import is complete, this PUT interface will be deprecated in favor of direct POST and PUT to the standard resource API.

## 2.4.6 Services

A **Service** provides server functionality beyond basic manipulation of resources.

## Authentication

Several endpoints handle user authentication.

<https://browsersupports.org/accounts/profile/> is an HTML page that summarizes the users' account, which includes:

- The username, which can't be changed.
- Changing or setting the password, which is optional if a linked account is used.
- Viewing, adding, and removing linked accounts, which are optional if a password is set. The supported linked account type is [Firefox Accounts](#).
- Viewing, adding, and removing emails. Emails can be verified (a link is clicked or a trusted linked account says it is verified), and one is the primary email used for communication.

Additional endpoints for authentication:

- `/accounts/` - Redirect to login or profile, depending on login state
- `/accounts/signup/` - Create a new account, using username and password
- `/accounts/login/` - Login to an existing account, using username and password or selecting a linked account
- `/accounts/logout/` - Logout from site
- `/accounts/password/change/` - Change an existing password
- `/accounts/password/set/` - Set the password for an account using only a linked account
- `/accounts/email/` - Manage associated email addresses
- `/accounts/password/reset/` - Start a password reset via email
- `/accounts/social/connections/` - Manage social accounts
- `/accounts/fxa/login/` - Start a [Firefox Accounts](#) login

## Data Browser

The Data Browser at <https://browsercompat.herokuapp.com/browse/> is an [Ember.js](#) single-page app that allows browsing the resources currently available in the API. It is built with Ember libraries that work with JSON API RC1.

## Importer

The MDN Importer at <https://browsercompat.herokuapp.com/importer/> is used to scrape data from MDN, extract compatibility data, pinpoint data issues on MDN pages, and commit extracted data to the API.

## 2.5 Tools

Some potentially useful scripts can be found in the `/tools` folder:

### 2.5.1 `download_data.py`

Download data from API. Usage:

```
$ tools/download_data.py [--api API] [-vq] [--data DATA]
```

- `--api <API>` (*optional*): Set the base URL of the API (default: `http://localhost:8000`)
- `-v` (*optional*): Print debug information
- `-q` (*optional*): Only print warnings
- `--data <DATA>` (*optional*): Set the output data folder (default: data subfolder in the working copy)

This will create several files (`browsers.json`, `versions.json`, etc.) that represent the API resources without pagination.

## 2.5.2 import\_mdn.py

Import features from MDN, or reparse imported features. Usage:

```
$ tools/import_mdn.py [--api API] [--user USER] [-vq]
```

- `--api <API>` (*optional*): Set the base URL of the API (default: `http://localhost:8000`)
- `--user <USER>` (*optional*): Set the username to use on the API (default: prompt for username)
- `-v` (*optional*): Print debug information
- `-q` (*optional*): Only print warnings

## 2.5.3 load\_spec\_data.py

Import specification data from MDN's [SpecName](#) and [Spec2](#). Usage:

```
$ tools/load_spec_data.py [--api <API>] [--user <USER>]
                        [-vq] [--all-data]
```

- `--api <API>` (*optional*): Set the base URL of the API (default: `http://localhost:8000`)
- `--user <USER>` (*optional*): Set the username to use on the API (default: prompt for username)
- `-v` (*optional*): Print debug information
- `-q` (*optional*): Only print warnings

## 2.5.4 load\_webcompat\_data.py

Initialize with compatibility data from the [WebPlatform](#) project. Usage:

```
$ tools/load_webcompat_data.py [--api <API>] [--user <USER>]
                              [-vq] [--all-data]
```

- `--api <API>` (*optional*): Set the base URL of the API (default: `http://localhost:8000`)
- `--user <USER>` (*optional*): Set the username to use on the API (default: prompt for username)
- `-v` (*optional*): Print debug information
- `-q` (*optional*): Only print warnings
- `--all-data` (*optional*): Import all data, rather than a subset

## 2.5.5 make\_doc\_requests.py

Make documentation/integration requests against an API. Used by tools/run\_integration\_tests.sh. Usage:

```
$ tools/integration_requests.py [--mode {display,generate,verify}]
                                [--api API]
                                [--raw RAW] [--cases CASES]
                                [--user USER] [--password PASSWORD]
                                [--include-mod] [-vq]
                                [case name [case name ...]]
```

- **--mode {display,generate,verify} (optional): Set the mode. Values are:**
  - display (default): Run GET requests against an API, printing the actual requests and responses.
  - generate: Run all requests against an API. Throw away some headers, such as Allow and Server. Modify other headers, such as Cookies, to make them consistent from run to run. Standardize some response data, such as creation and modification times. Store the cleaned-up requests and responses in the docs folder, for documentation and integration testing.
  - verify: Run all requests against an API, standardizing the requests and responses and comparing them to those in the docs folder.
- --api API (optional): Set the base URL of the API (default: http://localhost:8000)
- --raw RAW (optional): Set the path to the folder containing raw requests and responses (default: docs/raw)
- --cases CASES (optional): Set the path to the documentation cases JSON file (default docs/doc\_cases.json)
- --user USER: Set the username to use for requests (default anonymous requests)
- --password PASSWORD: Set the password to use for requests (default is prompt if --user set, otherwise use anonymous requests)
- --include-mod: If --mode display, then include requests that would modify the data, such as POST, PUT, and DELETE.
- -v: Be more verbose
- -q: Be quieter
- case name: Run the listed cases, not the full suite of cases

## 2.5.6 mirror\_mdn\_features.py

Create and update API features for MDN pages. This will create the branch features, and then import\_mdn.py can be used to generate the leaf features. Usage:

```
$ tools/mirror_mdn_features.py [--api API] [--data DATA]
                                [--user USER] [--password PASSWORD]
```

- --api API (optional): Set the base URL of the API (default: http://localhost:8000)
- --user USER: Set the username to use for requests (default is prompt for username)
- --password PASSWORD: Set the password to use for requests (default is prompt for username)
- --data DATA: Set the data folder for caching MDN page JSON

## 2.5.7 run\_integration\_tests.sh

Run a local API server with known data, make requests against it, and look for issues in the response. Usage:

```
$ tools/run_integration_tests.sh [-ghqv]
```

- `-g`: Generate documentation / integration test samples. If omitted, then responses are checked against the documentation samples. Useful for adding new documentation cases, or updating when the API changes.
- `-h`: Show a usage statement
- `-q`: Show less output
- `-v`: Show more output

## 2.5.8 sample\_mdn.py

Load random pages from MDN in your browser. Usage:

```
$ tools/sample_mdn.py
```

## 2.5.9 upload\_data.py

Upload data to the API. Usage:

```
$ tools/upload_data.py [--api API] [--user USER]
                    [-vq] [--data DATA]
```

- `--api <API>` (*optional*): Set the base URL of the API (default: `http://localhost:8000`)
- `--user <USER>` (*optional*): Set the username to use on the API (default: prompt for username)
- `-v` (*optional*): Print debug information
- `-q` (*optional*): Only print warnings
- `--data <DATA>` (*optional*): Set the output data folder (default: data subfolder in the working copy)

This will load the local resources from files (`browsers.json`, `versions.json`, etc), download the resources from the API, and upload the changes to make the API match the local resource files.

## 2.6 Issues

The near-term goal is that compatibility data lives in the API rather than on the MDN page, and MDN contributors maintain this data in the API instead of editing wiki pages. As this plan is executed, limitations are discovered in API, and changes are made and deployed.

When MDN is using the API as the primary source of compatibility data, we'll increment the API version for backward-incompatible changes. Until then, the v1 and v2 APIs will be modified with the evolving design.

This page outlines planned changes in the API. It doesn't describe all the known bugs and issues - see [tracking bug 996570](#) for a detailed list.

### 2.6.1 API Tasks

Here are the planned tasks that will require code changes to the API:

- **Implement v2 API** ([bug 1159406](#)) - JSON API v1.0 was released May 2015, and is quite different from JSON API Release Candidate 1 (RC1), which was used to design the v1 API. Many libraries in many languages support JSON API v1.0, including Ember 2.0, released August 2015. JSON API v1.0 will be supported in the v2 API, and both v1 and v2 will be supported until tools have been migrated to the v2 API.
- **Refactor Slugs** - Slugs were designed to be human-friendly aliases for resources, and also to be natural key alternatives to auto-incrementing IDs. However, whenever they are used for these purposes, they are found to have issues and require adjusting. For example, **Browser** slugs were adjusted to remove the “desktop by default” bias ([bug 1128525](#)). **Feature** slugs had issues when the MDN page had a title that bumped up against the 50 character limit ([bug 1078699](#)). Slugs will be modified to be changable by privileged users, instead of being set once at creation time. They may also be expanded beyond 50 characters, become a list of optional aliases, and be renamed in some cases. 3rd party developers should not rely on slugs remaining constant for the life of the API.
- **Switch from Auto-Incrementing IDs to UUIDs** ([bug 1230306](#)) - Resources are accessed by IDs, which using the database’s auto-incrementing integers. This means that two instances of the API could have the same data available at different IDs. Tools currently use slugs and other “permanent” attributes to identify resources as the same, even when IDs change. This is error-prone, and since slugs are planned to change, it would be useful to use UUIDs instead, so that the ID becomes a strong identifier between API instances.
- **Recover deleted items** ([bug 1159349](#)) - The History API could be used to recover deleted items, by administrators or users. The API needs to be modified to allow the reversion PATCH to deleted resources.
- **Move to api.browsercompat.org** ([bug 1153329](#)) - <https://browsercompat.org> will be the user-facing contribution interface and data browser. <https://api.browsercompat.org> will be the application-facing interface. This may require renaming the github repository, moving paths, and dropping sample views.

### 2.6.2 Data Tasks

Here are the planned tasks that involve adding or changing data in the API:

- **Add Feature IDs to MDN document data** - The mechanism for including a Compatibility table on MDN is evolving. The latest idea is that the Feature ID will be stored in the MDN document model, so that the existing parameterless KumaScript can be used to inject the table. There are open design questions, such as how the IDs will be set, if they need to change, and, if so, who will change them and how.
- **Import Data from MDN** - The majority of the project work has been building a scraper capable of importing existing data from MDN. As of September 2015, the scraper can extract data from most MDN pages, and can pinpoint issues that require page changes. The scraped data has to be committed to the API, and this process is ongoing.
- **Populate Browser Version Data** - MDN has detailed information for Firefox releases, and much less for other browsers. The **version** model supports some data, such as a release date, a retirement date, and URLs for release notes. Versions can also be sorted with respect to the related **browser**. This information can’t be imported release date and a retirement date, as well as other version data. Some sources for this data include:
  - Google Chrome - [Google Chrome Release History](#) on Wikipedia
  - Mozilla Firefox - [Firefox Release History](#) on Wikipedia and KumaScript macro [CompatGeckoDesktop](#)
  - Microsoft Internet Explorer - [Release History of IE](#) on Wikipedia
  - Opera - [Current Opera version history](#) and [Presto history](#) on opera.com
  - Safari - [Safari version history](#) on Wikipedia



### 2.6.3 Future Tasks

Once MDN is using the API as the primary data source for compatibility data, we can move on to other goals.

This includes, but is not limited to:

- Supporting other relations between resources. For example, the CSS property `color` can be assigned a CSS color type such as `color=red`. The CSS property `background-color` can also be assigned CSS color types. Different browsers may support a different set of named colors. This relation can't be directly represented in the API, but can be partially modelled with HTML links.
- Support test-driven compatibility data. This is generated by browser vendors and standardization supporters, and can be implemented as separate applications using the compatibility API.

## 2.7 History

The BrowserCompat API is still in beta, and true versioning won't start until it is shipped to the general public. The issues are tracked on [Bugzilla](#).

When code is merged to master (once or twice a week), it is deployed to Heroku at <https://browsercompat.herokuapp.com>.

Here are some development milestones:

- 2015-12-01 - MDN beta users can see API-backed tables on select pages
- 2015-09-11 - 3rd re-write of MDN importer ships, 82% of MDN can be imported
- 2015-02 - Added MDN importer
- 2014-12 - Added rest of resources, sample displays. Dropped versioning pre-release.
- 0.2.0 - 2014-10-13 - Add features, supports, pagination
- 0.1.0d - 2014-09-29 - Add resource-level caching
- 0.1.0c - 2014-09-16 - Add sample feature view, simplify draft API
- 0.1.0b - 2014-09-05 - Add filtering, more JSON API tuning.
- 0.1.0a - 2014-09-02 - First Heroku deployment. Browser and Version data.

## 2.8 Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)