

---

# **mg-rest-dm Documentation**

***Release 0.0.1***

**Mark McDowall**

**Oct 31, 2017**



---

## Contents:

---

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Requirements and Installation</b>       | <b>1</b>  |
| 1.1      | Requirements . . . . .                     | 1         |
| 1.2      | Installation . . . . .                     | 1         |
| 1.3      | Testing . . . . .                          | 2         |
| 1.4      | Documentation . . . . .                    | 2         |
| <b>2</b> | <b>mg-rest-dm server</b>                   | <b>3</b>  |
| 2.1      | Methods . . . . .                          | 3         |
| <b>3</b> | <b>Architectural Decision Record (ADR)</b> | <b>9</b>  |
| <b>4</b> | <b>License</b>                             | <b>11</b> |
| <b>5</b> | <b>Indices and tables</b>                  | <b>15</b> |
|          | <b>Python Module Index</b>                 | <b>17</b> |



---

## Requirements and Installation

---

### Requirements

#### Software

- Python 2.7.10+
- pyenv
- pyenv virtualenv
- pip

#### Python Modules

- DMP
- Flask
- Flask-Restful
- pytest
- Waitress
- Sphinx
- sphinx-autobuild

### Installation

#### Basics

Directly from GitHub:

```
1 git clone https://github.com/Multiscale-Genomics/mg-rest-dm.git
2 cd mg-rest-dm/
3 pip install -e .
4 pip install -r requirements.txt
```

Using pip:

```
1 pip install git+https://github.com/Multiscale-Genomics/mg-rest-dm.git
```

## Setting up a server

```
1 git clone https://github.com/Multiscale-Genomics/mg-rest-dm.git
2
3 cd mg-rest-dm
4 pyenv virtualenv 2.7.12 mg-rest-dm
5 pyenv activate mg-rest-dm
6 pip install git+https://github.com/Multiscale-Genomics/mg-dm-api.git
7 pip install -e .
8 pyenv deactivate
```

Requires a file with the name *mongodb.cnf* with the following parameters to define the MongoDB server:

```
1 [dmp]
2 host = localhost
3 port = 27017
4 user = testuser
5 pass = test123
6 db = dmp
```

Customise the *rest/auth\_meta.json* file to locate the authentication server

Starting the service:

```
1 nohup ${PATH_2_PYENV}/versions/2.7.12/envs/mg-rest-dm/bin/waitress-serve --listen=127.
  ↳0.0.1:5002 rest.app:APP &
```

## Testing

Test scripts are located in the *test/* directory. Run *pytest* to from the root repository directory to ensure that the API is working correctly.

The scripts require a valid hdf5 file generated using the scripts from mg-dm-api and a matching datasets.json file located in the *rest/* directory

## Documentation

To build the documentation:

```
1 pip install Sphinx
2 pip install sphinx-autobuild
3 cd docs
4 make html
```

### Methods

#### **class** `rest.app.EndPoints`

Class to handle the http requests for returning information about the end points

#### **get** ()

GET list all end points

List of all of the end points for the current service.

#### Example

```
1 curl -X GET http://localhost:5002/mug/api/dmp
```

#### **class** `rest.app.FileMeta`

Class to handle the http requests for retrieving the data from a file. This class is able to handle big[Bed|Wig] file and serve back the matching region in the relevant format. It is also possible to stream back the whole file of any type for use in other tools.

#### **delete** (\*args, \*\*kwargs)

DELETE Remove a file from the DM API

Function to remove the file from teh DM API. This will result in the file being removed from the records and therefore not available within the VRE or from the RESTful interface

#### Parameters

- **should be passed as the data block with the HTTP request**  
(*This*) –
- **json** (*dict*) –  
**file\_id** [str] ID of the stored file

### Example

```
1 echo '{
2   "file_id": "<file_id>"
3 }' > data.json
4
5 curl -X DELETE
6   -H "Content-Type: application/json"
7   -H "Authorization: Bearer teststring"
8   -d @data.json http://localhost:5002/mug/api/dmp/file
```

**get** (\*args, \*\*kwargs)

GET List values from the file

Call to obtain regions from the compressed index files for Bed, Wig and TSV based file formats that contain genomic information.

Other files can be streamed.

**Parameters** **file\_id** (*str*) – Identifier of the file to retrieve data from

**Returns** Returns a formatted in the relevant file type with any genomic features matching the format of the file.

**Return type** file

### Examples

```
1 curl -X GET http://localhost:5002/mug/api/dmp/track?file_id=test_file
```

**post** (\*args, \*\*kwargs)

POST Add a new file to the DM API

#### Parameters

- **should be passed as the data block with the HTTP request** (*This*) –

- **json** (*dict*) –

**user\_id** [str] User identifier

**file\_path** [str] Location of the file

**file\_type** [str] Tag for the file extension. The valid parameters are defined within the DM API documentation (mg-dm-api)

**data\_type** [str]

**What type of experiment is this data from. Options include** RNA-seq ChIP-seq  
MNase-seq WGBS HiC

**taxon\_id** [int] Taxonomic identifier for a species (Human = 9606)

**compressed** Options of the compression level of the file. If file is not compressed then do not include this parameter

**source\_id** [list] List of file\_ids that were used for generating this file. If this is the root file then do not include this parameter.



**meta\_data** [dict] Hash array describing the relevant metadata for the file, including the assembly if relevant

**Returns** Returns the id of the stored file

**Return type** file\_id

### Example

```

1 echo '{
2   "data_type": "RNA-seq",
3   "file_type": "fastq",
4   "meta_data": {
5     "assembly" : "GRCh38"
6   }, "taxon_id": 9606,
7   "file_path": "/tmp/test/path/RNA-seq/testing_123.fastq",
8   "parent_dir" "/tmp/test/path/RNA-seq/",
9   "size": 64000,
10  }' > data.json
11
12 curl -X POST
13   -H "Content-Type: application/json"
14   -H "Authorization: Bearer teststring"
15   -d @data.json http://localhost:5002/mug/api/dmp/file

```

**put** (\*args, \*\*kwargs)

PUT Update meta data

Request to update the meta data for a given file. This allows for the adding or removal of key-value pairs from the meta data.

#### Parameters

- **should be passed as the data block with the HTTP request** (*This*) –

- **json** (dict) –

**user\_id** [str] User identifier

**file\_id** [str] ID of the stored file

**type** [str] Options are ‘add\_meta’ or ‘remove\_meta’ to modify they key-value pairs for the file entry. Minimum sets of pairs are defined within the DM API (mg-dm-api)

**meta\_data** [dict] Hash array describing the relevant metadata key-value pairs that are to be added

**Returns** Returns the id of the stored file

**Return type** file\_id

### Example

To add a new key value pair:

```
1 echo '{
2     "type": "add_meta",
3     "file_id": "<file_id>",
4     "meta_data": {"citation": "PMID:1234567890"}
5 }' > data.json
6
7 curl -X PUT
8     -H "Content-Type: application/json"
9     -H "Authorization: Bearer teststring"
10    -d @data.json http://localhost:5002/mug/api/dmp/file
```

To remove a key value pair:

```
1 echo '{
2     "type": "remove_meta",
3     "file_id": "<file_id>",
4     "meta_data": ["citation"]
5 }' > data.json
6
7 curl -X PUT
8     -H "Content-Type: application/json"
9     -H "Authorization: Bearer teststring"
10    -d @data.json http://localhost:5002/mug/api/dmp/file
```

To modify a column value (file size):

```
1 echo '{
2     "type": "modify_column",
3     "file_id": "<file_id>",
4     "key": "<column_key>"
5     "value": "<new_value>"
6 }' > data.json
7
8 curl -X PUT
9     -H "Content-Type: application/json"
10    -H "Authorization: Bearer teststring"
11    -d @data.json http://localhost:5002/mug/api/dmp/file
```

### **class** rest.app.Files

Class to handle the http requests for retrieving the list of files for a given user handle

**get** (\*args, \*\*kwargs)

GET List user tracks

Function to list the files that are owned by a single user. It is possible to filter by assembly, file or data type, or to find track files that contain data for a given region

#### **Parameters**

- **assembly** (*str*) – Genome assembly accession
- **region** (*str*) – <chromosome>:<start\_pos>:<end\_pos>
- **file\_type** (*str*) –
- **data\_type** (*str*) –

### Example

```
1 curl -X GET http://localhost:5002/mug/api/dmp/Files?>
```

#### **class** rest.app.**FileHistory**

Class to handle the http requests for retrieving the list of file history of a given file for a given user handle

**get** (\*args, \*\*kwargs)

GET the list of files that were used for generating the defined file

### Example

```
1 curl -X GET http://localhost:5002/mug/api/dmp/file_history?file_id=<file_id>
```

#### **class** rest.app.**Ping**

Class to handle the http requests to ping a service

**get** ()

GET Status

List the current status of the service along with the relevant information about the version.

### Example

```
1 curl -X GET http://localhost:5002/mug/api/dmp/ping
```



## CHAPTER 3

---

### Architectural Decision Record (ADR)

---

This file is a record of the choices that have been made about the choice of software, packages, pipelines and data structures that have been made in this repository. This document should serve the help future developers (including the original authors) understand what certain choices were made.



## CHAPTER 4

---

### License

---

#### Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

#### 1. Definitions.

“License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

“Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

“Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, “control” means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

“You” (or “Your”) shall mean an individual or Legal Entity exercising permissions granted by this License.

“Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

“Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

“Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other

modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

“Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
  - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
  - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
  - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
  - (d) If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as



modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

## END OF TERMS AND CONDITIONS

### APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets “{}” replaced with your own identifying information. (Don’t include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright 2016 EMBL-European Bioinformatics Institute

Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## CHAPTER 5

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**r**

`rest.app`, 3



### D

`delete()` (`rest.app.FileMeta` method), 3

### E

`EndPoints` (class in `rest.app`), 3

### F

`FileHistory` (class in `rest.app`), 7

`FileMeta` (class in `rest.app`), 3

`Files` (class in `rest.app`), 6

### G

`get()` (`rest.app.EndPoints` method), 3

`get()` (`rest.app.FileHistory` method), 7

`get()` (`rest.app.FileMeta` method), 4

`get()` (`rest.app.Files` method), 6

`get()` (`rest.app.Ping` method), 7

### P

`Ping` (class in `rest.app`), 7

`post()` (`rest.app.FileMeta` method), 4

`put()` (`rest.app.FileMeta` method), 5

### R

`rest.app` (module), 3