htsget Documentation

Release 0.1.0

Jerome Kelleher

Nov 07, 2022

Contents

This package is a client implementation of the GA4GH htsget protocol. It provides a simple and reliable way to retrieve genomic data from servers supporting the protocol.

Slightly confusingly, this package and the protocol that it implements are both called "htsget". As a member of the GA4GH Streaming API group, I developed this client as part of the process of contributing to the design and evaluation of the protocol. I named the Python package "htsget", which was subsequently also adopted as the name of the protocol. Since no one objected to me continuing to use the name for my package there didn't seem to be much point in renaming it.

This is not an "official" GA4GH client for the protocol.

Please report any issues or features requests on GitHub

CHAPTER 1

Features:

- Thoroughly tested, production ready implementation.
- Robust to transient network errors (failed transfers are retried).
- Easy to *install* (pure Python implementation, minimal dependencies).
- Powerful *command line interface*.
- Simple *Python API*.

CHAPTER 2

Contents

2.1 Installation

To install `htsget`, simply run:

```
$ pip install htsget
```

If you wish to install `htsget` into a your local Python installation, use:

```
$ pip install htsget --user
```

However, you will need to ensure that the local binary directory (usually something like `$HOME/.local/bin`) is in your PATH.

2.2 Quickstart

2.2.1 Installation

Install from PyPI using

```
$ pip install htsget
```

See the *Installation* section for more details.

2.2.2 CLI Usage

The `htsget` command line downloads data from a URL as follows:

```
$ htsget http://htsnexus.rnd.dnanex.us/v1/reads/BroadHiSeqX_b37/NA12878 \
    --reference-name=2 --start=1000 --end=20000 -O NA12878_2.bam
```

Full documentation on the command line options is available via `htsget --help` or the *Command line interface* section.

2.2.3 API Usage

The Python API provides a single function *get()* which supports all of the arguments provided in the protocol. For example, to duplicate the example above, we can use the following code:

```
import htsget

url = "http://htsnexus.rnd.dnanex.us/v1/reads/BroadHiSeqX_b37/NA12878"
with open("NA12878_2.bam", "wb") as output:
    htsget.get(url, output, reference_name="2", start=1000, end=20000)
```

See the *API Documentation* section for full details.

2.3 API Documentation

htsget.**get**(*url*, *output*, *reference_name=None*, *reference_md5=None*, *start=None*, *end=None*, *fields=None*, *tags=None*, *notags=None*, *data_format=None*, *max_retries=5*, *retry_wait=5*, *timeout=120*, *bearer_token=None*, *headers=None*)

Runs a request to the specified URL and write the resulting data to the specified file-like object.

Parameters

- **url** (*str*) – The URL of the data to retrieve. This may be composed of a prefix such as `http://example.com/reads/` and an ID suffix such as `NA12878`. The full URL must be supplied here, i.e., in this example `http://example.com/reads/NA12878`.
- **output** (*file*) – A file-like object to write the downloaded data to. To support retrying of failed transfers, this file must be seekable. For this reason, retry will fail if `stdout` is provided.
- **reference_name** (*str*) – The reference sequence name, for example "chr1", "1", or "chrX". If unspecified, all data is returned.
- **reference_md5** (*str*) – The MD5 checksum uniquely representing the reference sequence as a lower-case hexadecimal string, calculated as the MD5 of the upper-case sequence excluding all whitespace characters (this is equivalent to SQ:M5 in SAM).
- **start** (*int*) – The start position of the range on the reference, 0-based, inclusive. If specified, `reference_name` or `reference_md5` must also be specified.
- **end** (*int*) – The end position of the range on the reference, 0-based exclusive. If specified, `reference_name` or `reference_md5` must also be specified.
- **data_format** (*str*) – The requested format of the returned data.
- **max_retries** (*int*) – The maximum number of times that an individual transfer will be retried.
- **retry_wait** (*float*) – The amount of time in seconds to wait before retrying a failed transfer.
- **timeout** (*float*) – The socket timeout for I/O operations.
- **bearer_token** – The OAuth2 Bearer token to present to the htsget ticket server. If this value is specified, the token is provided to the ticket server using the `Authorization:`

`Bearer [token]` header. If `bearer_token` is None or not specified, no Authorization header is sent to the server. Obtaining the bearer token is beyond the scope of htsget; consult the documentation for your server for information on authentication and how to obtain a valid token.

- **`headers`** – Additional headers needed for the requests to the htsget service.

2.4 Command line interface

CHAPTER 3

Indices and tables

- genindex
- modindex
- search

Index

G