genomehubs

Release 2.7.29

genomehubs

Sep 15, 2023

CONTENTS

CHAPTER

ONE

GENOMEHUBS

1.1 About

GenomeHubs comprises a set of tools to parse index and search and display genomic metadata, assembly features and sequencing status for projects under the Earth BioGenome Project umbrella that aim to sequence all described eukaryotic species over a period of 10 years.

Genomehubs builds on legacy code that supported taxon-oriented databases of butterflies & moths (lepbase.org), molluscs (molluscdb.org), mealybugs (mealybug.org) and more. Genomehubs is now search-oriented and positioned to scale to the challenges of mining data across almost 2 million species.

The first output from the new search-oriented GenomeHubs is Genomes on a Tree (GoaT, goat.genomehubs.org), which has been opublised in: Challis *et al*. 2023, **Genomes on a Tree (GoaT): A versatile, scalable search engine for genomic and sequencing project metadata across the eukaryotic tree of life**. Wellcome Open Research, **8**:24 doi:10.12688/wellcomeopenres.18658.1

The goat.genomehubs.org website is freely available with no logins or restrictions, and is being widely used by the academic community and especially by the Earth BioGenome Project to plan and coordinate efforts to sequence all described eukaryotic species.

The core GoaT/Genomehubs components are available as a set of Docker containers:

1.1.1 GoaT UI

A bundled web server to run a GoaT-specific instance of the GenomeHubs UI, as used at goat.genomehubs.org.

Usage

```
docker pull genomehubs/goat:latest

docker run -d --restart always \
    --net net-es -p 8880:8880 \
    --user $UID:$GROUPS \
    -e GH_CLIENT_PORT=8880 \
    -e GH_API_URL=https://goat.genomehubs.org/api/v2 \
    -e GH_SUGGESTED_TERM=Canidae \
    --name goat-ui \
    genomehubs/goat:latest
```

1.1.2 Genomehubs UI

A bundled web server to run an instance of the GenomeHubs UI, such as goat.genomehubs.org.

Usage

```
docker pull genomehubs/genomehubs-ui:latest

docker run -d --restart always \
    --net net-es -p 8880:8880 \
    --user $UID:$GROUPS \
    -e GH_CLIENT_PORT=8880 \
    -e GH_API_URL=https://goat.genomehubs.org/api/v2 \
    -e GH_SUGGESTED_TERM=Canidae \
    --name gh-ui \
    genomehubs/genomehubs-ui:latest
```

1.1.3 Genomehubs API

A bundled web server to run an instance of the GenomeHubs API. The GenomeHubs API underpins all search functionality for Genomes on a Tree (GoaT) goat.genomehubs.org. OpenAPI documentation for the GenomeHubs API instance used by GoaT is available at goat.genomehubs.org/api-docs.

Usage

```
docker pull genomehubs/genomehubs-api:latest

docker run -d \
    --restart always \
    --net net-es -p 3000:3000 \
    --user $UID:$GROUPS \
    -e GH_ORIGINS="https://goat.genomehubs.org null" \
    -e GH_HUBNAME=goat \
    -e GH_HUBPATH="/genomehubs/resources/" \
    -e GH_NODE="http://es1:9200" \
    -e GH_API_URL=https://goat.genomehubs.org/api/v2 \
    -e GH_RELEASE=$RELEASE \
    -e GH_SOURCE=https://github.com/genomehubs/goat-data \
    -e GH_ACCESS_LOG=/genomehubs/logs/access.log \
    -e GH_ERROR_LOG=/genomehubs/logs/error.log \
    -v /volumes/docker/logs/$RELEASE:/genomehubs/logs \
    -v /volumes/docker/resources:/genomehubs/resources \
    --name goat-api \
genomehubs/genomehubs-api:latest;
```

1.1.4 Genomehubs CLI

command line tool to process and index genomic metadata for GenomeHubs. Used to build and update GenomeHubs instances such as Genomes on a Tree goat.genomehubs.org.

Usage

```
docker pull genomehubs/genomehubs:latest
```

Parse [NCBI datasets](https://www.ncbi.nlm.nih.gov/datasets/) genome assembly metadata:

```
docker run --rm --network=host \
    -v `pwd`/sources:/genomehubs/sources \
     genomehubs/genomehubs:latest bash -c \
        "genomehubs parse \
            --ncbi-datasets-genome sources/assembly-data \
            --outfile sources/assembly-data/ncbi_datasets_eukaryota.tsv.gz"
```

Initialise a set of ElasticSearch indexes with [NCBI taxonomy](https://www.ncbi.nlm.nih.gov/taxonomy/) data for all eukaryotes:

```
docker run --rm --network=host \
    -v `pwd`/sources:/genomehubs/sources \
     genomehubs/genomehubs:latest bash -c \
        "genomehubs init \
            --es-host http://es1:9200 \
            --taxonomy-source ncbi \
            --config-file sources/goat.yaml \
            --taxonomy-jsonl sources/ena-taxonomy/ena-taxonomy.extra.jsonl.gz \
            --taxonomy-ncbi-root 2759 \
            --taxon-preload"
```

Index assembly metadata:

```
docker run --rm --network=host \
    -v `pwd`/sources:/genomehubs/sources \
     genomehubs/genomehubs:latest bash -c \
        "genomehubs index \
            --es-host http://es1:9200 \
            --taxonomy-source ncbi \
            --config-file sources/goat.yaml \
            --assembly-dir sources/assembly-data"
```

Fill taxon attribute values across the tree of life:

```
docker run --rm --network=host \
    -v `pwd`/sources:/genomehubs/sources \
     genomehubs/genomehubs:latest bash -c \
        "genomehubs fill \
            --es-host http://es1:9200 \
            --taxonomy-source ncbi \
            --config-file sources/goat.yaml \
```

(continues on next page)

(continued from previous page)

```
            --traverse-root 2759 \
            --traverse-infer-both"
```

1.2 Related projects

Some GenomeHubs components are hosted in separate open source repositories (all under MIT licenses), including:

1.2.1 BlobToolKit

Interactive quality assessment of genome assemblies.

Explore analysed public assemblies at blobtoolkit.genomehubs.org/view

1.2.2 GoaT CLI

A command line interface for GoaT.

The GoaT CLI builds URLs to query the Goat API, removing some of the complexity of the GoaT API. for the end user.

CHAPTER

TWO

INSTALLATION

At the command line:

```
pip install genomehubs
```

CHAPTER

THREE

USAGE

To use genomehubs in a project:

```
import genomehubs
```

CHAPTER

FOUR

REFERENCE

4.1 init

4.2 parse

4.3 index

4.4 fill

CHAPTER

FIVE

CONTRIBUTING

5.1 Bug reports

When reporting a bug please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.2 Documentation improvements

Contributions to the official genomehubs docs and internal docstrings are always welcome.

5.3 Feature requests and feedback

The best way to send feedback is to file an issue at https://github.com/genomehubs/genomehubs/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that code contributions are welcome

5.4 Development

To install the development version of *genomehubs*:

1. Clone the *genomehubs* repository:

```
git clone https://github.com/genomehubs/genomehubs
```

2. Install the dependencies using pip:

```
cd genomehubs
pip install -r requirements.txt
```

3. Build and install the *genomehubs* package:

```
python3 setup.py sdist bdist_wheel \
&& echo y | pip uninstall genomehubs \
&& pip install dist/genomehubs-2.0.0-py3-none-any.whl
```

To set up *genomehubs* for local development:

1. Fork *genomehubs <https://github.com/genomehubs/genomehubs>* - (look for the "Fork" button).
2. Clone your fork locally:

```
git clone git@github.com:USERNAME/genomehubs.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

4. When you're done making changes run all the checks and docs builder with tox one command:

```
tox
```

5. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

5.4.1 Pull Request Guidelines

If you need some code review or feedback while you're developing the code just make the pull request.

For merging, you should:

1. Include passing tests (run `tox`)[1].
2. Update documentation when there's new API, functionality etc.
3. Add a note to `CHANGELOG.rst` about the changes.
4. Add yourself to `AUTHORS.rst`.

[1] If you don't have all the necessary python versions available locally you can rely on Travis - it will run the tests for each change you add in the pull request.
It will be slower though ...

5.4.2 Tips

To run a subset of tests:

```
tox -e envname -- pytest -k test_myfeature
```

To run all the test environments in *parallel*:

```
tox -p
```

CHAPTER

SIX

AUTHORS

- Richard Challis - https://twitter.com/rjchallis
- Sujai Kumar - https://twitter.com/sujaik

CHAPTER

SEVEN

CHANGELOG

7.1 2.0.0 (2020-07-02)

- First release on PyPI.

CHAPTER

EIGHT

INDICES AND TABLES

- genindex
- modindex
- search