

Documentation

-

compute SQLite databases
-
for each zoom level in GDV

Yohan JAROSZ

Ecole Polytechnique Fédérale de Lausanne
Bioinformatics and Biostatistics Core Facility
Sybit developer

Abstract

This project is intended to be a Daemon (it will run in background). It will takes scores from a SQLite database BBCF formatted representing values for an assembly (score along the genome for different chromosomes) and calculate scores for different zoom level. In result, there will be one new SQLite database for each zoom level per chromosomes

Contents

1	Project	2
1.1	Compiling	2
1.2	Configuration	2
1.3	Running	3
1.4	Logging	3
1.5	Libraries	3
1.6	Post parameters	3
1.7	Warning	3

1 Project

The project is available on https://github.com/bbcf/gdv/conversion/compute_sqlite_scores/

1.1 Compiling

Just download project then run :

```
cd {project directory}
ant jar
```

You should obtain something like :

```
:$ ant jar
Buildfile: {project directory}/build.xml

clean:
    [delete] Deleting directory {project directory}/bin

init:
    [mkdir] Created dir: {project directory}/bin
    [copy] Copying 3 files to {project directory}/bin

compile:
    [echo] conversion: {project directory}build.xml
    [javac] Compiling 13 source files to {project directory}/bin

jar:
    [jar] Building jar: {project directory}/compute_to_sqlite.jar

BUILD SUCCESSFUL
Total time: 2 seconds
```

This will result in the file *compute_to_sqlite.jar*

1.2 Configuration

There is a configuration file under `project_directory/conf` : `conf.yaml` which contains two parameters :

- path to a temporary directory (cannot be `.`) (`tmp_directory`) files will be written in that directory then erased
- an url if you want feedback (`feedback_url`) like `http://myServer.com/jbrowse/post` see [1.6](#) for more.

1.3 Running

There is two scripts in `project_directory` (`start.sh` & `stop.sh`).

Update them with changing the paths to where you put the project.

You can now launch & stop the daemon.

1.4 Logging

A file *compute_scores.log* is created to log & debug the processing.

1.5 Libraries

Some libraries are needed in order to run the daemon. You can put them in your classpath and change a bit `build.xml` if you want.

1.6 Post parameters

This section will describe feedback provided by the daemon with HTTP/POST.

The feedback URL is the one in the configuration file ([1.2](#))

Parameters are :

id : "track_status"

track_id : an identifier for the process

mess : a message (can be "computing", "a percentage of completion" or "completed")

Errors are not send and should be added in the next version.

1.7 Warning

this daemon handle lots of files simultaneously and can resulting with errors (cannot write,read or copy more files during process).

You should increases the limit for the user which handle this process

```
$ulimit -Hn  
$cat /etc/security/limits.conf  
  
40960 is a nice value.
```