# GTAS ETL Jobs and ETL Scheduler Deployment Guide (Linux)

## 1. Required tools

- Java 1.8
- Pentaho Data Integration tool (See separate installation guide)
- Elasticsearch and Kibana should be up and running (for the report ETL)

#### 2. Create Local ETL Environment

a. Create your local ETL parent directory gtas-etl in the root directory (/) Note: Although the ETL jobs and scheduler can be installed under any directory, the following instruction show the installations under the root directory (/). If you choose to create this parent directory at a different location, make sue that you review and update the properties file listed in the next section.

b. Pull the latest from the GTAS GitHub repository. Go to the gtas-etl-main directory and copy ETL and configuration files with their directories to your local ETL parent directory, gtas-etl

Copy from	Copy to
GTAS/gtas-etl-main/ (all files and directories)	/gtas-et1/

- c. Go to the GTAS/gtas-etl-scheduler project and build it as follows:
  - mvn clean install

After a successful build, copy the gtas-etl-scheduler.jar file from *GTAS/gtas-etl-scheduler/target* to your ETL parent directory /gtas-etl

d. Verify that your local ETL environment looks as follows:

∠ gtas-eti
config
report-etl (job.properties)
scheduler (scheduler.properties)
neo4j-etl (all .ktr and .kjb files for neo4j-etl)
temp (can be empty. Temporary files during runtime will be created here)
report-etl (all .ktr and .kjb files for report-etl)
index (one or more index.properites files)
temp (can be empty. Temporary files during runtime will be created here)
template (One or more json template files and one or more template.properties files)
workpad (One or more json workpad files and one or more workpad.properties files)
neo4j-etl (can be empty. The neo4j-etl logs will be created during runtime)
report-etl (can be empty. The report-etl logs will be created during runtime)
scheduler (can be empty. The scheduler logs will be created during runtime)
gtas-etl-scheduler.jar

#### 3. Update Property Files

**Note:** If you <u>did not</u> create gtas-etl in the root directory (/), please review each property file carefully. <u>In addition to the following properties</u>, you will have to update properties with values pointing to a file or a directory structure.

## a. /gtas-etl/config/scheduler/scheduler.properties

• No update is required

## b. /gtas-etl/config/neo4j-etl/job.properties

- EXT\_VAR\_GTAS\_DB\_USER\_NAME
- EXT\_VAR\_GTAS\_DB\_PASSWORD
- EXT VAR NEO4J DB USER NAME
- EXT\_VAR\_NEO4J\_DB\_PASSWORD

#### c. /gtas-etl/config/report-etl/job.properties

- EXT\_VAR\_GTAS\_DB\_USER\_NAME
- EXT\_VAR\_GTAS\_DB\_PASSWORD
- EXT\_VAR\_JAVA\_TRUST\_STORE
- EXT\_VAR\_JAVA\_TRUST\_STORE\_PASSWORD
- EXT\_VAR\_ELSEARCH\_USERNAME
- EXT\_VAR\_ELSEARCH\_PASSWORD
- EXT\_VAR\_KIBANA\_USERNAME
- EXT\_VAR\_KIBANA\_PASSWORD

## 4. Launch the gtas-etl-scheduler

- a. Start the scheduler as follows
  - java -jar gtas-etl-scheduler.jar