

Hawk Installation Guide

Monitoring Tool of MEASURE Platform

.....

Hardware Requirements

- Recommended memory capacity: 8 Go
- OS: Hawk should work on Linux, Windows and Mac

Prerequisite

- Java must be installed. We recommend the 1.8 version.

Hawk Installation

- Download hawk-server-nogpl_XXXXXXX-linux.gtk.x86_64.zip the latest version of "Hawk Server" at the link:

<https://github.com/mondo-project/mondo-hawk/releases>

- Extract the archive in your computer
- Provide configuration files for the instances to be executed on Hawk Server. These instances are described in XML files and must be added in the configuration directory at the root of hawk server before running hawk-server.

Configuration

- You must provide a configuration file for each instance you want to execute on Hawk server.
- The configuration files must be located in the configuration folder at the root of the hawk server project. The configuration files can have any name and must be in XML format.
- You can run many instances in Hawk Server at the same time and must add configuration files for each of these instances.
- In the configuration file we specify
 - The name of the Hawk instance
 - The backend (database engine) in which the instance will be running
 - "org.hawk.orientdb.OrientDatabase" for normal queries.
 - "org.hawk.greycat.LevelDBGreycatDatabase" for temporal queries
 - Delay for automatic updating (synchronization) of the model in the database. You can specify Min and Max values for Delay in milliseconds. If Max=0, it means the updating of the indexed model will be manual.
 - Plugins needed to execute the instance. We must specify the necessary plugins depending on the used backend (see the examples below).

- The path of Modelio metamodel descriptor. The modelio metamodel can be found in Modelio model repository in "admin" folder.
- The monitored repositories: they can be local (type="org.hawk.localfolder.LocalFolder") or SVN links (type="org.hawk.svn.SvnManager"). See the example below for more information.
- More details on configuration file can be found in this link: <https://github.com/mondo-project/mondo-hawk/wiki/File-based-instance-configuration>
- Here is an example of configuration file for an instance running on time aware backend in order to perform temporal queries.

Databio_instance.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<hawk backend="org.hawk.greycat.LevelDBGreycatDatabase" name="DataBio"
factory="org.hawk.timeaware.factory.TimeAwareHawkFactory">
  <delay max="0" min="0"/>
  <plugins>
    <plugin name="org.hawk.modelio.exml.listeners.ModelioGraphChangeListener"/>
    <plugin name="org.hawk.modelio.exml.metamodel.ModelioMetaModelResourceFactory"/>
    <plugin name="org.hawk.modelio.exml.model.ModelioModelResourceFactory"/>
    <plugin name="org.hawk.timeaware.graph.TimeAwareModelUpdater"/>
  </plugins>
  <metamodels>
    <metamodel location="/home/.../metamodel_descriptor.xml" uri=""/>
  </metamodels>
  <repositories>
    <repository frozen="false"
location="https://rd.constellation.modeliosoft.com/svn/.../trunk/model" pass="****"
type="org.hawk.svn.SvnManager" user="****"/>
  </repositories>
</hawk>
```

- Here is an example of configuration file for an instance with non-temporal queries

Project2_instance.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<hawk backend="org.hawk.orientdb.OrientDatabase" name="Measure">
  <delay max="0" min="0"/>
  <plugins>
    <plugin name="org.hawk.modelio.exml.listeners.ModelioGraphChangeListener"/>
    <plugin name="org.hawk.modelio.exml.metamodel.ModelioMetaModelResourceFactory"/>
    <plugin name="org.hawk.modelio.exml.model.ModelioModelResourceFactory"/>
    <plugin name="org.hawk.graph.updater.GraphModelUpdater"/>
  </plugins>
  <metamodels>
    <metamodel location="/home/.../metamodel_descriptor.xml" uri=""/>
  </metamodels>
  <repositories>
    <repository frozen="false" location="file:///home/.../design/model/"
type="org.hawk.localfolder.LocalFolder" user="" pass=""/>
    <repository frozen="false" location="file:///home/.../model2/model/"
type="org.hawk.localfolder.LocalFolder" user="" pass=""/>
  </repositories>
</hawk>
```

Running Hawk Server

- To run Hawk Server in Linux, execute in the terminal the command:

```
./run-server.sh
```

- Before executing any queries wait for the indexation process to be finished. A message indicating the completion of this task will be displayed:

```
Updated Hawk instance (success)
```

- If you haven't specified an automatic update in the configuration file, you can run the command:

```
./sync-server.sh
```

- You can stop Hawk server by typing "exit" in the console or by pressing: CTRL + C