



gbif-indexingtoolkit

The GBIF Harvesting and Indexing Toolkit (HIT)

 Search projects

[Project Home](#)
[Downloads](#)
[Wiki](#)
[Issues](#)
[Source](#)
[Export to GitHub](#)

Search Current pages ▼ for

Installation

General installation instructions

Featured

Updated Nov 8, 2010 by kyle.br...@gmail.com

General Installation Guide

For new installations, please review that that the [requirements](#) have been met before proceeding.

Requirements

The HIT will run on any of the most widespread Operating Systems (Linux, Mac OS (TM), MS Windows (TM)) provided that:

- Java (TM) runtime environment (<http://java.sun.com/>) version 6 or higher is installed. See the section entitled [Java](#) for help.
- A web server with a servlet container, such as Apache Tomcat (<http://tomcat.apache.org/>) or Jetty (<http://jetty.mortbay.com/>), is installed and connected to the internet. See the section entitled [Application Server](#) for help.
- MySQL 5.1 or higher is installed; See the section entitled [MySQL](#) for more help.

HIT Installation

Simple enough, the installation requires the following steps:

1. [Set up the harvesting \('hit'\) database](#)
2. [Set up the indexing \('portal'\) database](#)
3. [Install the application](#)

Then it is necessary to 4. [Configure the application](#), to tailor the application to your own environment.

Then it is highly recommended to 5. [Configure the environment](#), to tailor the application to your running environment.

Once everything has been properly installed and configured, you can 6. [Access the application](#)

Set up the harvesting ('hit') database

- Create the harvesting database

If you have not already installed MySQL, instructions on how to install it can be found here: [Installing MySQL](#).

Assuming you already have MySQL installed, you need to create a database (The name I have used, and that you will encounter in these instructions is 'hit'. Feel free to use this name, or call your database something else if you wish). The command typed from inside a MySQL prompt would look like this:

```
mysql>create database hit DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci;
```

- Load the database schema

Download the harvesting database's schema file (most recent at time of writing called located [here](#)

Switch to use the database

```
mysql>use hit;
```

Load the schema

```
mysql>source ${download_location}/hit.sql
```

Alternatively from a 'terminal window'(Linux, Mac OS (TM))/'command prompt'(MS Windows(TM)) you could also load the schema by typing the following command:

```
$ mysql -u root -Dhit -p < ${download_location}/hit.sql
```

Set up the indexing ('portal') database

- Create the indexing database

This database does not necessarily have to reside on the same database management system as the harvesting database. Wherever you decide to put it, you must first create the database obviously. (The name I have used, and that you will encounter in these instructions is 'portal'. Feel free to use this name, or call your indexing database something else if you wish). The command typed from inside a MySQL prompt would look like this:

```
mysql>create database portal DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci;
```

- Load the portal database schema

Download the GBIF Portal database's schema file (called portal.ddl) located [here](#)

Switch to use the database

```
mysql>use portal;
```

Load the schema

```
mysql>source ${download_location}/portal.ddl
```

Alternatively from a 'terminal window' (Linux, Mac OS (TM)) / 'command prompt'(MS Windows(TM)) you could also load the schema by typing the following command:

```
$ mysql -u root -Dportal -p < ${download_location}/portal.ddl
```

- Populate the portal database with the minimum required data

Download the script responsible for populating the portal database (called initPortal.data) located [here](#)

Similar to how the portal schema was loaded, type

```
mysql>source ${download_location}/initPortal.data
```

Alternatively from a 'terminal window' (Linux, Mac OS (TM)) / 'command prompt'(MS Windows(TM)) you could also load the data by typing the following command:

```
$ mysql -u root -Dportal -p < ${download_location}/initPortal.data
```

Install the application

Note: these instructions are specific for Tomcat

- Download the web archive file (hit-with-synchroniser-1.13-RC1.war at the time of writing) from the project downloads page: <http://code.google.com/p/gbif-indexingtoolkit/downloads/list>
- Rename it to hit.war and copy it into the webapps folder of the Tomcat installation directory.
- Launch Tomcat:

Open a 'command prompt' (MS Windows(TM)) / 'terminal window' (Linux, Mac OS (TM))

Navigate to the Tomcat installation folder

Go to the bin folder and execute startup.bat (MS Windows(TM)) or startup.sh (Linux, Mac OS (TM))

- Configure the JVM settings for Tomcat (Tomcat's default settings are inadequate to be able to run the application) See the section [Tomcat configuration](#) for guidance.

Configure the application

Configurations take place in two files: `${tomcat.directory}/webapps/hit/WEB-INF/classes/application.properties` and `${tomcat.directory}/webapps/hit/WEB-INF/classes/applicationContext-security.xml`

Please note that some configurations vary depending on the operating system used.

- Inside `${tomcat.directory}/webapps/hit/WEB-INF/classes/application.properties` - Configure the application's root directory (where files will be saved)

IMPORTANT: This directory must have sufficient space to write to, and TOMCAT must have permission to write to it.

Modify property 'harvest.directory'

```
For Windows
e.g. harvest.directory=\\opt\\hit\\
```

```
All other Operating Systems
e.g. harvest.directory=/opt/hit/
```

- Inside `${tomcat.directory}/webapps/hit/WEB-INF/classes/application.properties` - Configure the application's backup directory (where files will be backed up)

IMPORTANT: This directory must have sufficient space to write to, and TOMCAT must have permission to write to it.

Modify property 'backup.directory'

For Windows

e.g. `backup.directory=\\opt\\hit-backup\\`

All other Operating Systems

e.g. `backup.directory=/opt/hit-backup/`

- Configure the harvesting database ('hit')

Modify the following properties as necessary:

```
e.g. dataSource.servername=localhost
dataSource.name=hit
dataSource.username=root
dataSource.password=password
dataSource.url=jdbc:mysql://localhost:3306/hit?autoReconnect=true&useUnicode=true&characterEncoding=UTF-8
```

- Inside `$(tomcat.directory)/webapps/hit/WEB-INF/classes/application.properties` - Configure the indexing database ('portal')

Modify the following properties as necessary:

```
e.g. portalDataSource.servername=localhost
portalDataSource.name=portal
portalDataSource.username=root
portalDataSource.password=password
portalDataSource.url=jdbc:mysql://localhost:3306/portal?autoReconnect=true&useUnicode=true&characterEncoding=UTF-8
```

- Inside `$(tomcat.directory)/webapps/hit/WEB-INF/classes/applicationContext-security.xml` - Configure the user management

The default username and password is (admin/koala). To modify this password, replace the password as seen below with another one, that has been md5 encoded (you could use an md5 encoder online to do this, like [this one](#) for example)

```
<authentication-manager>
  <authentication-provider>
    <password-encoder hash="md5"/>
    <user-service>
      <user name="admin" password="a564de63c2d0da68cf47586ee05984d7" authorities="ROLE_ADMIN" />
    </user-service>
  </authentication-provider>
</authentication-manager>
```

Once all configurations have been completed, it's necessary for Tomcat to be restarted in order for the changes to take effect.

Configure the environment

- Configure log rotation

Since running in debug mode might cause the application logs (e.g. `catalina.out`) to fill up your storage it might be a good idea to have some log rotation/compression schema in place. Many modern Linux distributions have a "logrotate" utility which will conveniently perform logs maintenance.

Logrotate typically reads its configuration from the command line; On modern RedHat compatible systems, by default one gets a "logrotate" script in the `cron.daily` folder which in turn calls logrotate with `/etc/logrotate.conf` as an argument. In a situation like this all you have to do is to create/modify a file in `/etc/logrotate.d/` e.g.

`/etc/logrotate.d/tomcat`

```
/var/log/tomcat5/catalina.out {
copytruncate
size 500M
rotate 5
compress
missingok
}
```

Notice the "size 500M" which will force the rotation if the log is larger than 500M; The file `/etc/logrotate.d/tomcat` might be already there if you have installed Tomcat from an RPM package.

- Configure disk clean up

Since the HIT is saving large quantities of data to disk, it might be a good idea to have some disk clean up schema in place. Many modern Linux distributions have tasks listed in `/etc/cron.x` folders. There is a `cron.daily` folder where you can have a script which is run daily. All you have to do is create a file in `/etc/cron.daily` e.g.

`/etc/cron.daily/hit_cleanup`

```
#!/bin/bash
# removes all inventory and search request and response files older than 3 months as requested
# HITLOC must be equal to environment variable harvest.directory as set in the HIT's application.properties file

IFS=`echo -en "\n\b"`
AGE=90
HITLOC=/mnt/fiber/hit/

for k in inventory_request.* inventory_response.* search_request.* search_response.*; do
    for i in `find $HITLOC -type f -mtime +$AGE -name $k`; do rm -f $i; done
done

unset IFS
```

Access the application

The HIT should now be accessible in any browser at <http://localhost:8080/hit/>

Should the application not appear, please check Tomcat's log information for indications of what might be amiss.

Some helpful sections for reference

Java

Mac OS X (TM) includes a fully configured and ready-to-use Java runtime environment. For MS Windows (TM) machines, follow these steps to install:

1. Download the Java 2 Standard Edition Runtime Environment (JRE), release version 6.0 or later, from <http://java.sun.com/j2se>.
2. Install the JRE according to the instructions included with the release.
3. Set an environment variable named JRE_HOME to the pathname of the directory into which you installed the JRE, e.g. c:\jre5.0:
 1. Open the Control Panel and click the System icon.
 2. Go to the Advanced pane and click the Environment variables button.
 3. In the System variables section, click New. In the Variable name box, type JRE_HOME; in the Variable value box, type the path to the JRE e.g. C:\Program Files\Java\jre6

Application Server

The HIT should run in any compliant java application server using Java 1.6. Tomcat 5.5 is recommended for the installation. It was also successfully tested on Tomcat 6 and Jetty.

The application server will require a rather large amount of memory allocated as the HIT runs multithreaded operations. It is recommended to increase the server's available memory to at least 1GB (see general instructions below).

Tomcat installation

1. Download a binary distribution of Tomcat from: <http://tomcat.apache.org/download-55.cgi>
2. Unpack the binary distribution into a location of your choice (e.g. C:\apache-tomcat-5.5.27)
3. Open a command prompt window and navigate to the installation folder.
4. Mac OS only: some files need to have their permissions adapted: type 'sudo chmod 775' at the command line. You will be asked for your administrative password.
5. Go to the bin folder and execute startut.bat (Windows (TM)) or startup.sh (Linux, Mac OS (TM)). A new Tomcat window will open. You need to keep this window open.

Tomcat configuration

Tomcat comes with very little allocated memory. In order to improve performance levels, this amount should be increased depending on the physical memory on the server. If the server has at least 2GB of RAM, the available memory should be increased to at least 1GB. The Java PermGen space should also be increased to at least 128M (the default value is 64M).

The settings used will depend on each server specifications, but a suggested setup for a server running with 2GB RAM is as follows:

For **Linux, Mac OS (TM)**, add/edit the following line in the catalina.sh file:

```
# to set minimum memory to 256MB, maximum memory to 1024MB, minimum PermGen space to 128Mb, maximum PermGen space to 256Mb
JAVA_OPTS="$JAVA_OPTS -Xms256m -Xmx1024m -XX:PermSize=128m -XX:MaxPermSize=256m -Dfile.encoding=UTF-8"
```

For **Windows**, add/edit the following line in the catalina.bat file:

```
# to set minimum memory to 256MB, maximum memory to 1024MB, minimum PermGen space to 128Mb, maximum PermGen space to 256Mb
Set JAVA_OPTS=-Xms256m -Xmx1024m -XX:PermSize=128m -XX:MaxPermSize=256m -Dfile.encoding=UTF-8
```

MySQL

The MySQL installation guide for version 5.1 can be found here: <http://dev.mysql.com/doc/refman/5.1/en/installing.html>

▶ [Sign in](#) to add a comment

[Terms](#) - [Privacy](#) - [Project Hosting Help](#)

Powered by [Google Project Hosting](#)