Table of Contents

[RESOURCES 3](#_Toc377022914)

[Access GitHub’s IHTSDO/ISAAC project 3](#_Toc377022915)

[Ensure have JDK 1.7 or JDK 1.8 installed on system 3](#_Toc377022916)

[Ensure have Maven 3.0.5 installed on system 3](#_Toc377022917)

[Subfolders found under IHTSDO/ISAAC project on GitHub to download: 3](#_Toc377022918)

[Berkeley Database 3](#_Toc377022919)

[General Setup 3](#_Toc377022920)

[Setup your settings.xml file 3](#_Toc377022921)

[Setup Project in IDE 4](#_Toc377022922)

[Eclipse 4](#_Toc377022923)

[NetBeans 5](#_Toc377022924)

[IntelliJ 5](#_Toc377022925)

[Command Line 5](#_Toc377022926)

[Create a Git local repository of ISAAC 5](#_Toc377022927)

[Desktop Client 5](#_Toc377022928)

[Command Line 6](#_Toc377022929)

[Import local Git repository into IDE 6](#_Toc377022930)

[Eclipse 6](#_Toc377022931)

[NetBeans 8](#_Toc377022932)

[IntelliJ 8](#_Toc377022933)

[Command Line 8](#_Toc377022934)

[Build project 8](#_Toc377022935)

[Eclipse 8](#_Toc377022936)

[NetBeans 9](#_Toc377022937)

[IntelliJ 9](#_Toc377022938)

[Command Line 9](#_Toc377022939)

[Deploy Berkeley Database 9](#_Toc377022940)

[Eclipse 9](#_Toc377022941)

[NetBeans 10](#_Toc377022942)

[IntelliJ 10](#_Toc377022943)

[Command Line 10](#_Toc377022944)

[Run application 10](#_Toc377022945)

[Eclipse 11](#_Toc377022946)

[NetBeans 11](#_Toc377022947)

[IntelliJ 11](#_Toc377022948)

[Command Line 11](#_Toc377022949)

# RESOURCES

## Access GitHub’s IHTSDO/ISAAC project

1. Create a GitHub account at <http://github.com>
2. Request access to the IHTSDO /ISAAC Project on GitHub from a project administrator
   1. Currently, only Rory Davidson may provide you with permissions to the project
3. Access IHTSDO/ISAAC project at: <https://github.com/IHTSDO/ISAAC>

## Ensure have JDK 1.7 or JDK 1.8 installed on system

1. Download from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
2. Install as instructed

## Ensure have Maven 3.0.5 installed on system

1. Download from <http://maven.apache.org/download.cgi>
2. Install as instructed

## Subfolders found under IHTSDO/ISAAC project on GitHub to download:

1. TK3 (Toolkit 3)
2. Contains the refactored core libraries
3. Contains updated Icons libraries
4. Does not contain GUI elements as the existing workbench GUI has been removed based on the architecture review’s recommendations
5. (SIM) Simple Integrated Model
6. Contains a simple event-driven document model for representing patient records

## Berkeley Database

1. Log into VA-Archiva at <https://mgr.servers.aceworkspace.net/apps/va-archiva/index.action>
2. Download tcc-test-data-3.0.zip file from <https://mgr.servers.aceworkspace.net/apps/va-archiva/repository/all/org/ihtsdo/otf/tcc-test-data/3.0/>

# General Setup

## Setup your settings.xml file

1. Pull settings.xml file from <https://github.com/IHTSDO/ISAAC/tree/master/resources> folder.
2. Replace <username> and <password> with your VA-Archiva credentials:

…..

<servers>

<server>

<id>va-maestro</id>

<username>VA-ARCHIVA\_USER\_NAME</username>

<password>VA-ARCHIVA\_USER \_PASSWORD</password>

</server>

</servers>

…..

1. Define your repositories

…..

<profiles>

<profile>

<id>my-profile</id>

<activation>

<activeByDefault>true</activeByDefault>

</activation>

<repositories>

<repository>

<id>va-maestro</id>

<name>VA Maestro</name>

<url>https://mgr.servers.aceworkspace.net/apps/va-archiva/repository/all/</url>

</repository>

</repositories>

<pluginRepositories>

<pluginRepository>

<id>va-maestro</id>

<name>VA Maestro</name>

<url>https://mgr.servers.aceworkspace.net/apps/va-archiva/repository/all/</url>

</pluginRepository>

</pluginRepositories>

</profile>

</profiles>

…..

# Setup Project in IDE

## Eclipse

1. Via **Windows-Preferences**, ensure **Java-Installed JREs** version is using latest jdk
2. Via **Windows-Preferences**, ensure **Maven-Installations** version is using local maven 3.0.5
3. Via **Windows-Preferences**, ensure **Maven-User Settings** has the User Settings pointing to the proper settings.xml file updated in General Setup portion of this document

## NetBeans

## IntelliJ

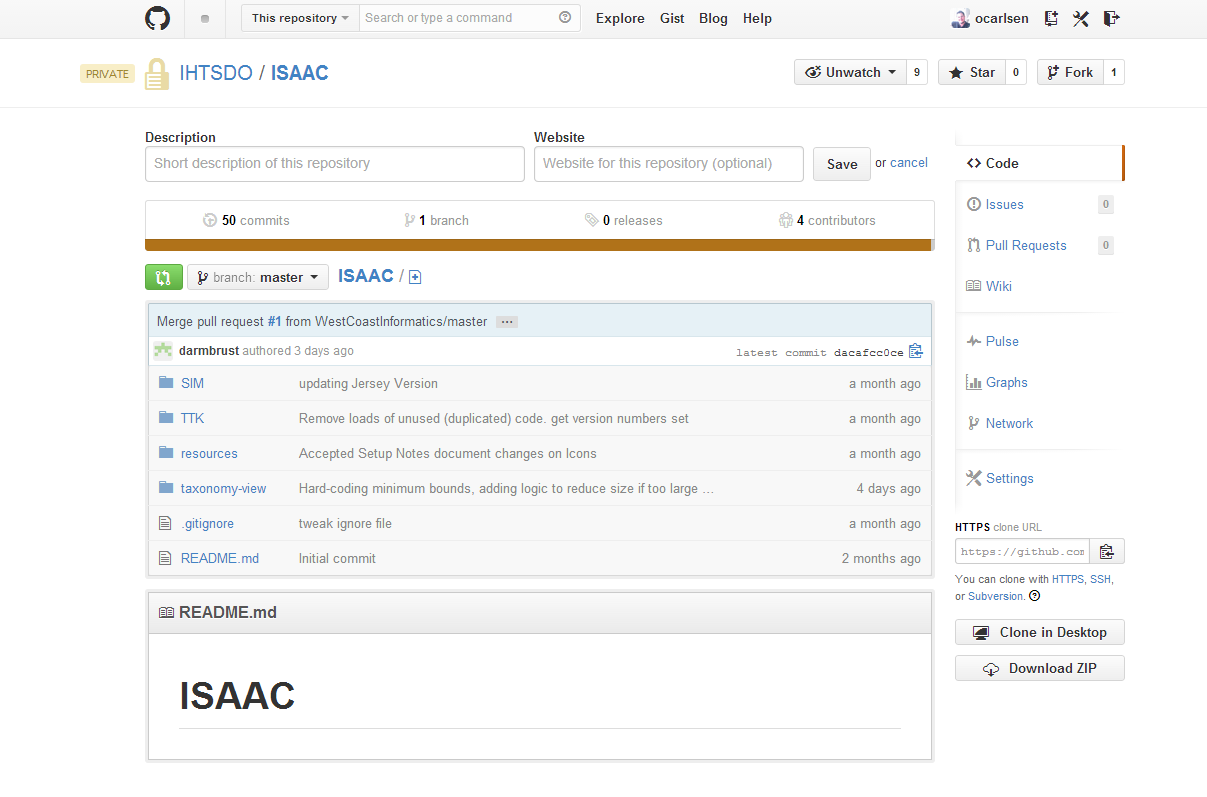
## Command Line

# Create a Git local repository of ISAAC

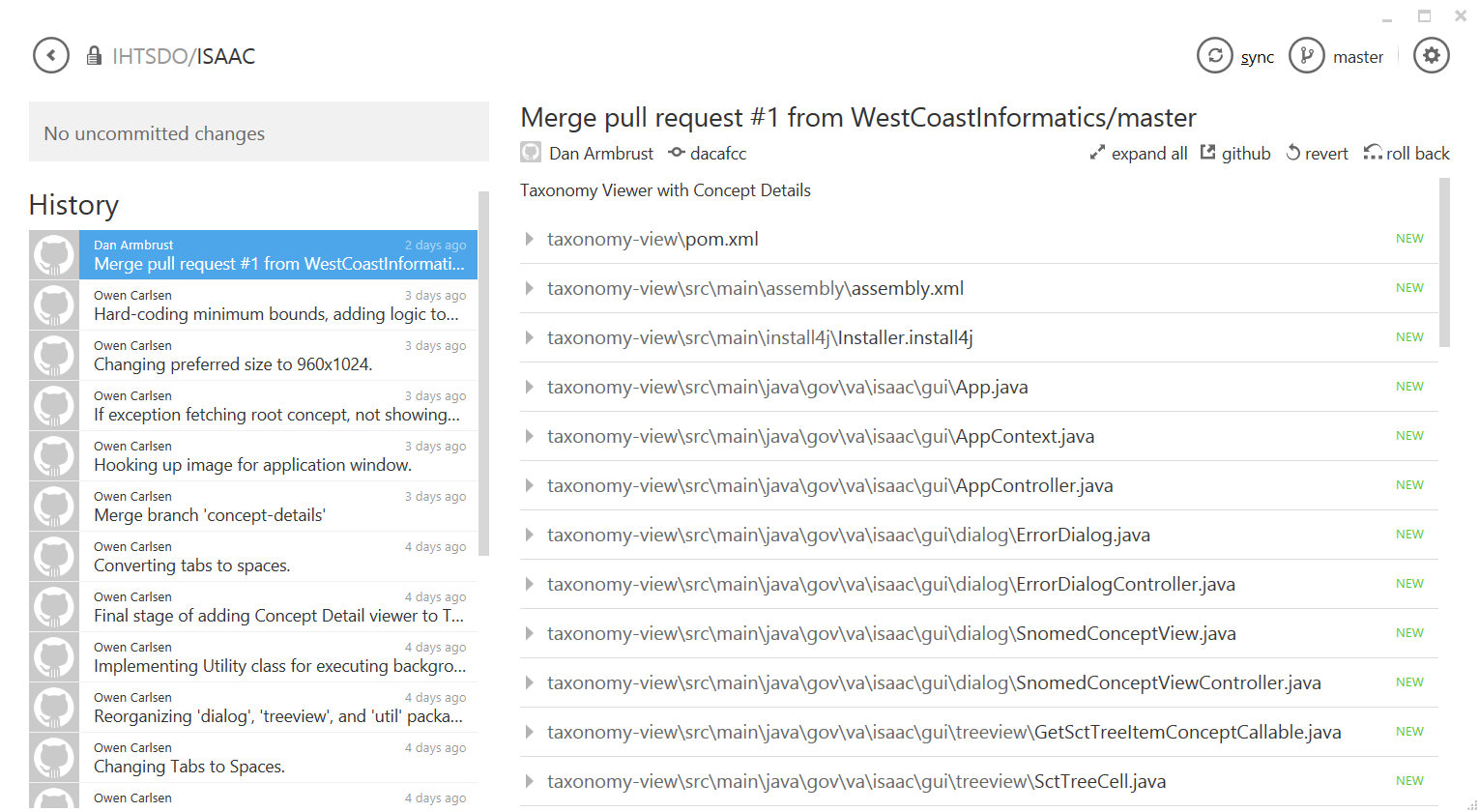
## Desktop Client

Demonstrated for Windows via Windows GitHub client

1. Go to <https://github.com/IHTSDO/ISAAC>
2. Select ‘Clone in Desktop’ button at lower-right



1. Opens GitHub client
2. Content will download
3. Once done, GitHub client will look like this



1. Locate local repository
   1. Click on settings button
   2. Select **open in explorer** option
   3. Explorer opens in directory containing local Git repository

## Command Line

Demonstrated in a Unix/Linux terminal

1. Change directory to the location you wish to clone the repository.

cd /Projects/ISAAC

1. Download the project with the Git “clone” command.

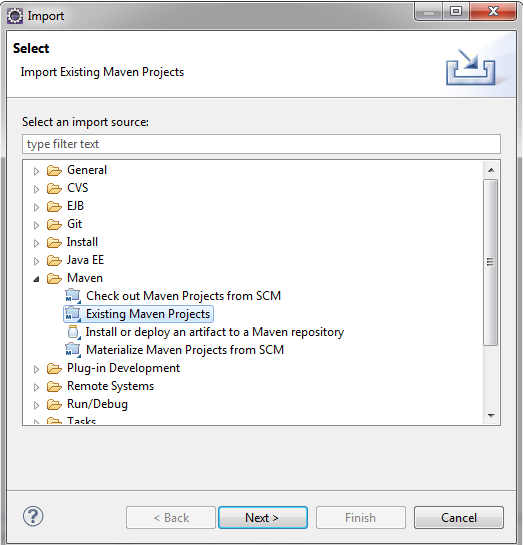
git clone https://github.com/IHTSDO/ISAAC

1. The Taxonomy Viewer code will be in the “taxonomy-view” folder.

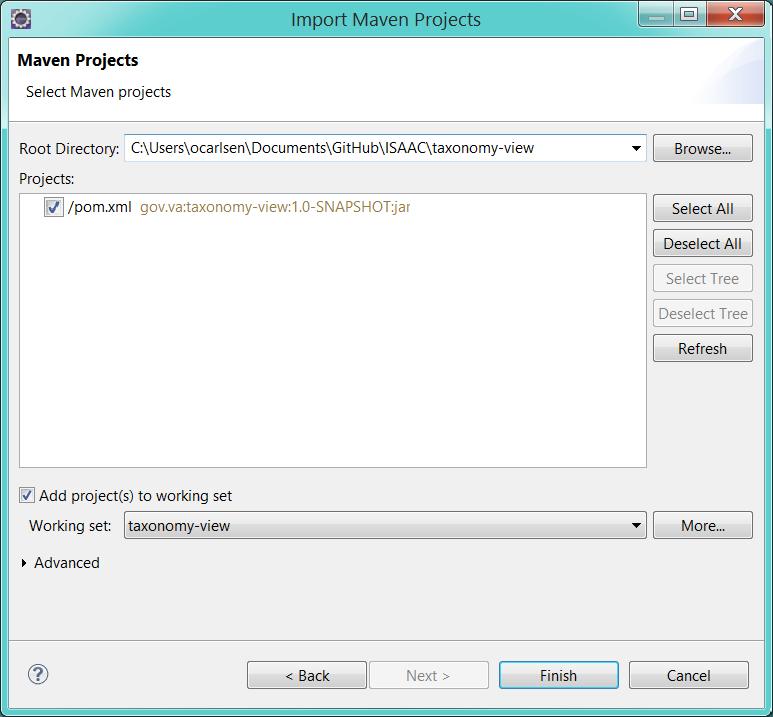
# Import local Git repository into IDE

## Eclipse

1. Select **File-Import**
2. Select **Maven-Existing** Maven Project option



1. Point **Root Directory** to the “taxonomy-view” folder in ISAAC’s local GitHub repository



1. Click the **Finish** button when done. When the project is done importing, it will be called “taxonomy-view”.

## NetBeans

## IntelliJ

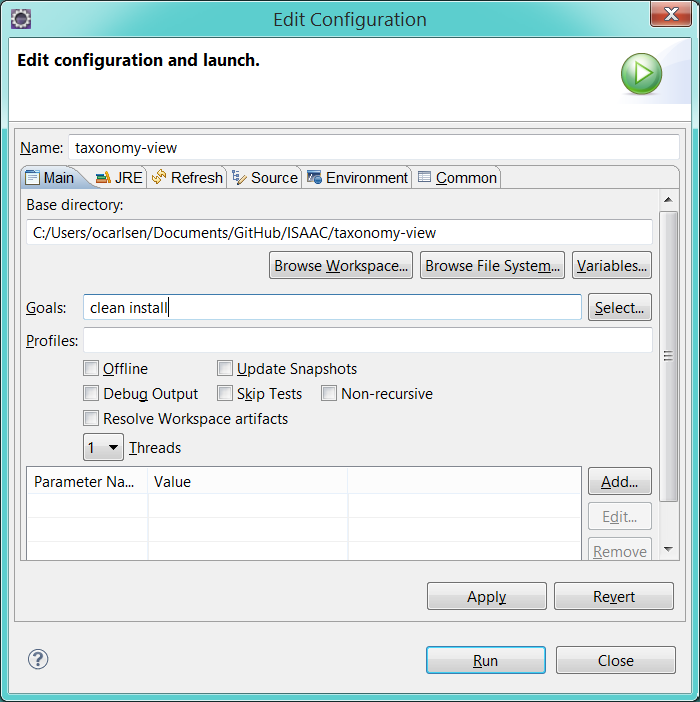
## Command Line

# Build project

The following sections describe how to build the Taxonomy Viewer app.

## Eclipse

1. Create a Run Configuration for the project
   1. Right-click the “taxonomy-view” project and select **Run As-Maven Build** from the menu
   2. Type “clean install” in the **Goals** field of the **Edit Configuration** dialog:



1. Click the **Run** button to build the Taxonomy Viewer app.

## NetBeans

## IntelliJ

## Command Line

Demonstrated in a Unix/Linux terminal

1. Change directory to the location containing the “taxonomy-view” folder in the ISAAC repository cloned in [Create a Git local repository of ISAAC](#_Create_a_Git).

cd /Projects/ISAAC/taxonomy-view

1. Build the project with the Maven as usual.

mvn clean install

1. The Taxonomy Viewer app will be built into a ZIP file in the “target” folder.

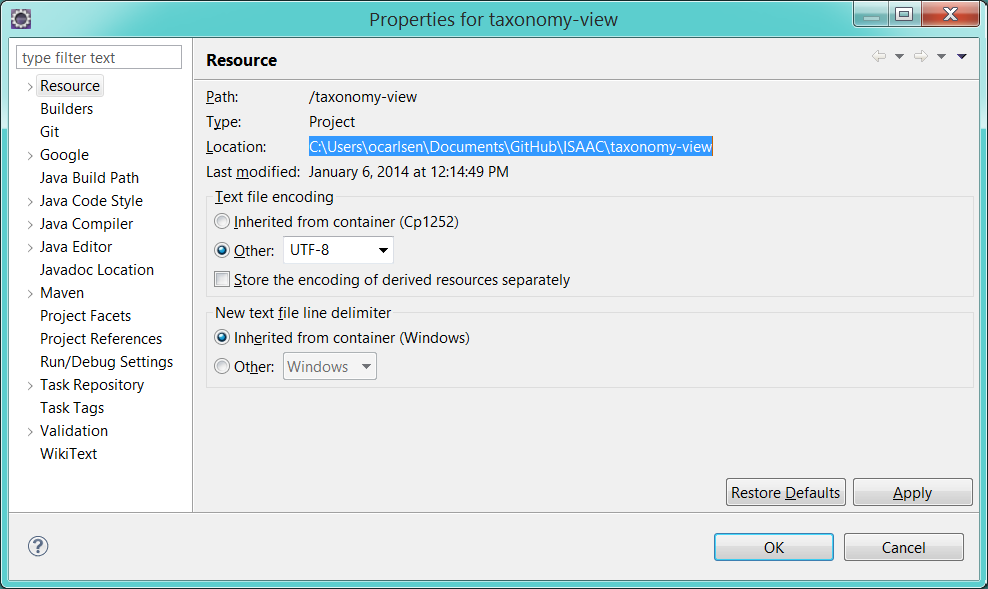
# Deploy Berkeley Database

The Taxonomy Viewer will need a database to connect to. The following steps describe how to deploy the database described in the [Berkeley Database](#_Berkeley_Database) section.

1. Locate the tcc-test-data-3.0.zip file you downloaded.
2. Extract the ZIP file into a directory called tcc-test-data-3.0.
3. Note the extracted “berkeley-db” folder inside. This contains the database content.

## Eclipse

1. Right-click the “taxonomy-view” project and select **Properties**.
2. In the Properties dialog, make a note of the **Location** field:



1. Copy the “berkeley-db” folder you extracted above into this directory.
2. Run the Taxonomy Viewer again and notice there are no more errors.

## NetBeans

## IntelliJ

## Command Line

Demonstrated in a Unix/Linux terminal

1. Return to the directory containing the “target” folder of the “taxonomy-view” project.

cd /Projects/ISAAC/taxonomy-view/target

1. Copy the “berkeley-db” folder you extracted above into this directory.

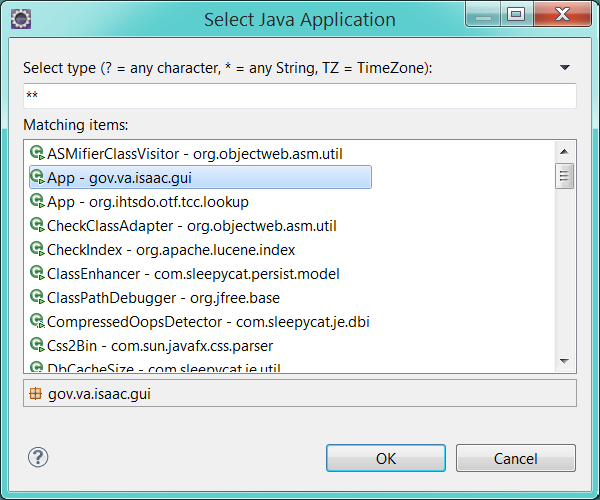
cp -r /tcc-test-data-3.0/berkeley-db .

# Run application

The following sections describe how to run the Taxonomy Viewer app.

## Eclipse

1. Create a Run Configuration for the app
   1. Right-click the “taxonomy-view” project and select **Run As-Java Application** from the menu.
   2. Select “App – gov.va.isaac.gui” from the **Matching Items** list of the **Select Java Application** dialog:



1. Click the **Ok** button to start the Taxonomy Viewer app. An Eclipse run configuration will be automatically created for you.

## NetBeans

## IntelliJ

## Command Line

Demonstrated in a Unix/Linux terminal

1. Change directory to the location containing the “target” folder of the “taxonomy-view” project you built in [Build project](#_Build_project).

cd /Projects/ISAAC/taxonomy-view/target

1. Unzip the Taxonomy Viewer ZIP file.

jar xvf taxonomy-view.zip

1. This will extract several scripts and an executable JAR into the “target” folder. Locate the appropriate script and execute it.

./taxonomy-view64.sh