




Opentheso v4

Par Miled Rousset `miled[dot]rousset[.]mom.fr`

Installation sous Debian

téléchargez la dernière version de Debian sur le site (<http://www.debian.org/>)

-  Pour pouvoir installer les packages ci-dessous, il faut être root ou avoir le droit (sudo).
- installation effectuée avec (Debian GNU/Linux 8.7.1 _Jessie_)
 - java 7
 - tomcat 8

Installation de Postgresql

1. Installez le serveur Postgresql

- `apt-get install postgresql`

pour contrôler la version : `psql --version`

`psql (PostgreSQL) 9.4.10`

2. ensuite il faut changer le mot de passe de Postgres

- `passwd postgres`
- Saisissez le nouveau mot de passe :
- Saisissez-le à nouveau :

Installation de Tomcat

1. pour installer tomcat, il faut repasser en root et procéder comme suit :
- `aptitude install tomcat8`

l'installation de tomcat installe en même temps java, si ce n'est pas le cas, il faut installer la version 7 de java.

Installation de Java 7

Il faut tester d'abord si java est installé :

- `java -version`
`java version « 1.7.0_121 »`

sinon,

- `aptitude install openjdk-7-jre`

pour passer de la version 6 à 7 de java :

- `update-alternatives --config java`

Il existe 2 choix pour l'alternative java (qui fournit /usr/bin/java).

Sélection	Chemin	Priorité	État
* 0	/usr/lib/jvm/java-6-openjdk-amd64/jre/bin/java	1061	mode automatique
1	/usr/lib/jvm/java-6-openjdk-amd64/jre/bin/java	1061	mode manuel
2	/usr/lib/jvm/java-7-openjdk-amd64/jre/bin/java	1051	mode manuel

Appuyez sur <Entrée> pour conserver la valeur par défaut[*] ou choisissez le numéro sélectionné :2

Installation du module AJP (pont de connexion entre apache et tomcat)

- `aptitude install libapache2-mod-jk`
- `vi /var/lib/tomcat8/conf/server.xml`

il faut chercher le texte suivant :

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<!--
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
-->
```

puis enlever le commentaire autour du connector et ajouter l'encodage UTF-8 :

```
<Connector port="8009" URIEncoding="UTF-8" protocol="AJP/1.3" redirectPort="8443"/>
```

- `/etc/init.d/tomcat8 restart`

Modifier les chemins dans le fichier de conf :

`/etc/libapache2-mod-jk/workers.properties`

Il faut donner les bons chemins et le nom à utiliser avec apache (exp: `ajp13_worker`)

Parametrage de Apache (Virtual Host)

(ceci est un exemple de paramétrage dans `/etc/apache2/sites-enabled/`)

il faut créer un fichier de conf propre à Opentheso, voici un exemple suivant l'installation décrite ici :

- `cd /etc/apache2/sites-enabled/`
- `vi opentheso.conf`

```
<VirtualHost *:80>
    ServerAdmin toto@mondomaine.fr

    JkMount /opentheso/* ajp13_worker
    JKMount /opentheso ajp13_worker

    ServerName opentheso.mondomaine.fr

    DocumentRoot /var/lib/tomcat8/webapps/opentheso
    ErrorLog /var/lib/tomcat8/logs/error.log
    CustomLog /var/lib/tomcat8/logs/access.log common
    <Directory /var/lib/tomcat8/webapps/opentheso>
        Options -Indexes +FollowSymLinks +MultiViews
        AllowOverride All
        Order allow,deny
        Allow from all
    </Directory>

    <Location "/META-INF/">
        deny from all
    </Location>
    <Location "/WEB-INF/">
        deny from all
    </Location>

    <IfModule mod_rewrite.c>
        RewriteEngine On
        RewriteRule ^/$ /opentheso/$1 [R]
    </IfModule>

    # pour gérer les images, il faut créer le dossier avant
    <Directory /var/www/images>
        Options -Indexes +FollowSymLinks +MultiViews
        AllowOverride All
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

Il faut maintenant activer le module « rewrite » dans apache2 :

- `a2enmod rewrite`

puis relancer apache

- `/etc/init.d/apache2 restart`

paramétrage de Tomcat 7

Il faut ajouter la ligne suivante dans /etc/init.d/tomcat7 :
`export CATALINA_OPTS="-Xms256M -Xmx1024M"`
sinon, le serveur tomcat produit une erreur de type (Out Of Memory)


Si nécessaire, il faut modifier cette ligne pour pointer vers le bon java
`JDK_DIRS="/usr/lib/jvm/java-7-openjdk-amd64"`

Installation d'Opentheso :

Version 4.2.3 (c'est la version actuelle à la rédaction de la doc)

Déploiement du war Opentheso :

pour déployer le « war » sur le serveur que vous venez d'installer, il faut déposer le fichier « opentheso.war » dans le dossier /var/lib/tomcat8/webapps/

 Attention, il faut renommer le war qui est fourni sur GitHub pour avoir la même configuration que la doc exp : (opentheso.war au lieu de opentheso-4.2.3.war).

à partir de ce moment là, on peut modifier les fichiers de paramètres décrits ci-dessous :
les fichiers de conf se trouvent dans :
/var/lib/tomcat8/webapps/opentheso/WEB-INF/classes

Modification du fichier « preferences.properties » :

(avant tout, il faut renommer le fichier, il est fourni par défaut sous le nom : preferences_dist.properties)

c'est la langue de base du site dans laquelle les thésaurus seront générés par défaut.

#Preferred language
workLanguage=fr

choix de l'attribution d'identifiants pour les concepts : numériques ou alpha-numériques ; ce paramètre peut être changer en cours de route sans incident

#identifiant 1=alphanumeric or 2=numeric
identifierType=2

ce paramètre permet de lancer Opentheso au démarrage avec un thesaurus en particulier, ici « 1 » est l'identifiant du trésaurus que nous souhaitons avoir par défaut

#l'identifiant du thésaurus par défaut à charger
defaultThesaurusId=1

Opentheso prend en charge la gestion des identifiants type ARK, mais pour cela, il faut disposer d'un serveur ou webservices qui fourni des identifiants Ark. Actuellement, Opentheso utilise le WebServices proposé par le PSIR de la MOM qui fourni des identifiants ARK « Projet Arkéo » ; avec Arkéo, il est possible de gérer plusieurs réseaux, projets, équipes indépendamment).

Donc nous avons la possibilité d'activer ou désactiver cette fonctionnalité.

(serverArk : c'est l'alias vers le serveur Ark

paramètre pour utiliser un serveur Ark ou non (true / false)
useArk=true

serverArk=http://ark.frantiq.fr/ark:/

Pour pouvoir écrire les images, il faut que votre utilisateur apache puisse avoir le droit d'écriture sur ce dossier « pathImage ».

paramètre du dossier de stockage des images et vignettes

pathImage=/var/www/images/

il faut ensuite créer le dossier « resize » à l'intérieur du dossier « /var/www/images » ou pathImage.

dossierResize=resize

Adresse web du site pour la création automatique de liens vers les concepts

paramètre de l'URL de votre Site

cheminSite=http://opentheso.votredomaine.fr/

paramètre pour lier le thésaurus à une base de données par URL

on peut se connecter à une base distante par une URL pré-formatée

il y a juste à laisser (terme) comme variable qui prend la valeur du concept sélectionné.

exp :

notice.url=http://www.test.fr/fr/recherche?field=field_link_mots_cles_code&search=terme

bdd.active=false

url.bdd=http://www.test.fr/fr/recherche?field=field_link_mots_cles_code&search=terme

(Il est indispensable de déposer les deux fichiers de paramétrage (repositories.xml et SchemaMappings.xml) dans le bon dossier et régler les préférences suivant votre base de données. Ces deux fichiers sont livrés dans le War d'Opentheso à la racine du Site.

paramètre de la connexion avec un SIGB type KOHA

en utilisant jzkit

ce paramètre permet d'activer ou désactiver la recherche z3950

z3950.aktif=true

collection.adresse=KOHA/biblios

notice.url=http://catalogue.votredomaine.fr/cgi-bin/koha/opac-search.pl?

type=opac&op=do_search&q=an=terme

url.encode=UTF-8

pathNotice1=/var/www/notices/repositories.xml

pathNotice2=/var/www/notices/SchemaMappings.xml

Modification du fichier de connexion à la base de données Postgresql

« hikari.properties » :

(avant tout, il faut renommer le fichier, il est fourni par défaut sous le nom : hikari_dist.properties)

minimumIdle=1

autoCommit=true

setMaximumPoolSize=1000

idleTimeout=30000

connectionTimeout=30000

connectionTestQuery=SELECT 1

dataSourceClassName=org.postgresql.ds.PGSimpleDataSource

Ces paramètres sont à adapter suivant l'installation de la base de données d'Opentheso sous Postgresql.

#mon thesaurus

dataSource.serverName=localhost

dataSource.user=opentheso

dataSource.password=opentheso

dataSource.databaseName=opentheso

Installation de la base de données pour Opentheso :

Pour pouvoir installer la base de données sous postgresql, il faut récupérer le script SQL fourni, voici les étapes :

pour réaliser cette partie, il faut connaître le fonctionnement d'un serveur postgresql (création d'utilisateurs, de bases de données, lancement de scripts etc) il y a pas mal de documentations sur internet à ce sujet.

2. création du rôle et de la base de données *(ne pas saisir le texte en vert)*

- *su postgres*
- *psql*
- *postgres=# CREATE USER opentheso PASSWORD 'opentheso';*
- *postgres=# create Database opentheso OWNER opentheso;*
- *\q*

3. installation du script SQL pour préparer la base de données (*script fourni dans opentheso/src/main/resources/install/opentheso_current.sql*)

- *psql -U opentheso -h localhost -p 5432 opentheso < /home/toto/opentheso_current.sql*
- *entrer le mot de passe de l'utilisateur que vous venez de créer (opentheso d'après l'exemple ci-dessus) :*



maintenant, la base de données est installée.

si tout va bien, alors on doit pouvoir relancer tomcat et apache2 :

- */etc/init.d/tomcat8 restart*
- */etc/init.d/apache2 restart*

Maintenant, dernière étape, il faut aller sur le domaine de votre installation pour tester si tout fonctionne :

<http://opentheso.votredomaine.fr/>

c'est le paramètre que vous avez défini dans le virtualHost avec la commande « ServerName opentheso.votredomaine.fr »

Première connexion à Opentheseo :

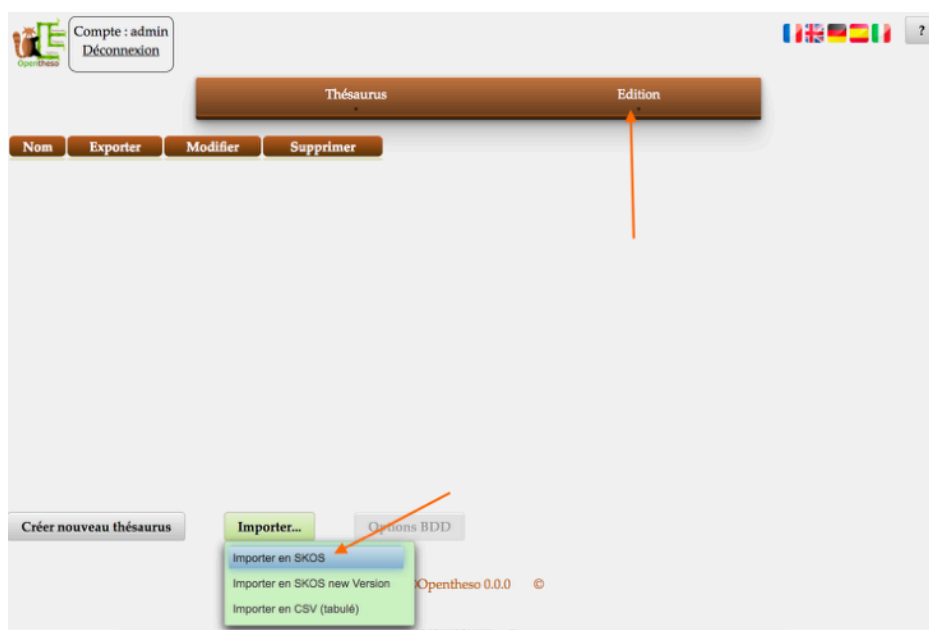
Pour se connecter la première fois, il y a un utilisateur super-admin qui est défini par défaut :

user : admin

pass : admin

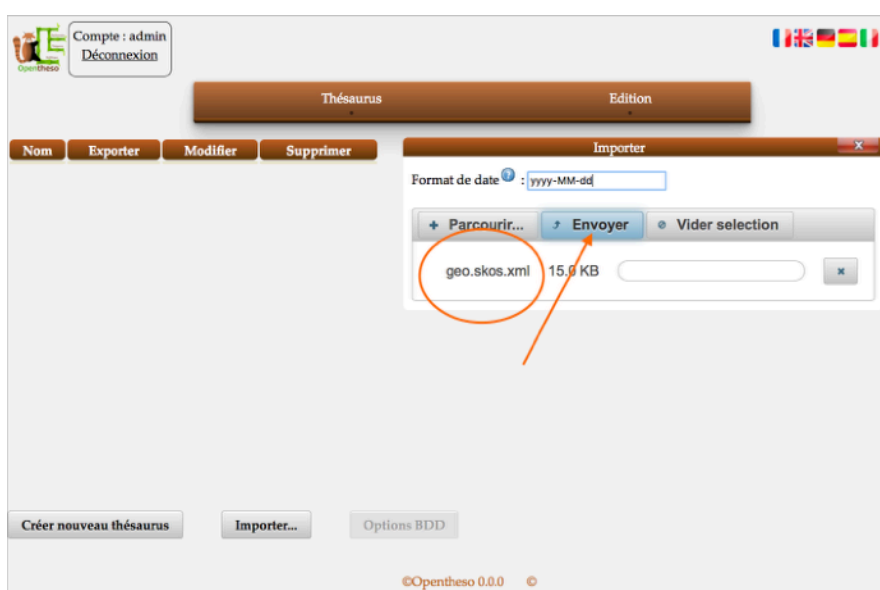
ensuite, il faut importer le thesaurus de test fourni dans le package : « geo.skos.xml »

pour cela, allez dans (**Edition**) -> puis en bas, il y a (**Importer en SKOS**)

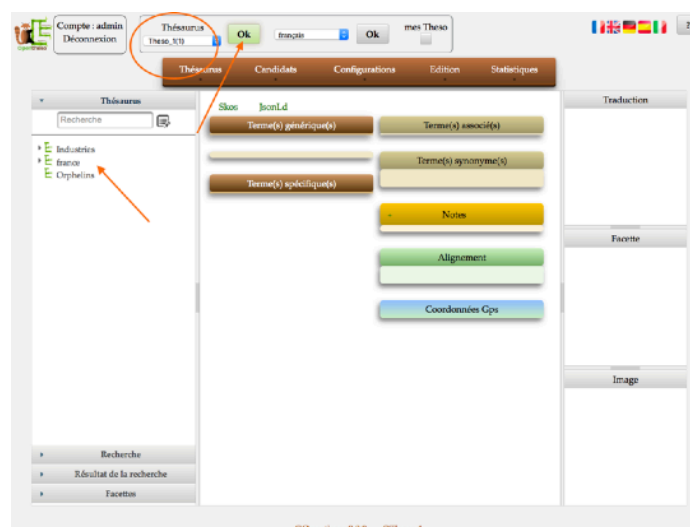


pour le choix de la date : il faut taper : (yyyy-MM-dd) ce qui correspond au format de la date dans le fichier Skos (2014-07-21).

puis cliquez sur (**Envoyer**)



ensuite Il faut recharger la page et vous devriez avoir votre thésaurus de test



Pour accéder au WebServices, il faut aller à cette adresse :

<http://votredomaine.fr/opentheso/copyright.xhtml>

vous avez des exemples prédéfinis concernant le fonctionnement de ce service.

💡 **Attention, pour pouvoir tester votre webservices, il faut d'abord naviguer dans votre thesaurus et se positionner sur un concept en particulier, ensuite, allez sur la page ci-dessus et testez.**

exemple :

http://votredomaine.fr/opentheso/webresources/rest/skos/concept/id=C_26984&th=1
devrait produire :

```
- <rdf:RDF>
  - <skos:Concept rdf:about="http://193.48.137.89/opentheso/?id=C_26984&idt=1">
    <skos:prefLabel xml:lang="fr">productions</skos:prefLabel>
    <skos:inScheme rdf:resource="http://193.48.137.89/opentheso/?id=MT_18&idt=1"/>
    <skos:broader rdf:resource="http://193.48.137.89/opentheso/?id=C_26975&idt=1"/>
    <skos:narrower rdf:resource="http://193.48.137.89/opentheso/?id=C_26985&idt=1"/>
    <skos:narrower rdf:resource="http://193.48.137.89/opentheso/?id=C_26986&idt=1"/>
    <skos:narrower rdf:resource="http://193.48.137.89/opentheso/?id=C_26987&idt=1"/>
    <dcterms:created>2014-07-21</dcterms:created>
    <dcterms:modified>2014-07-21</dcterms:modified>
    <dcterms:identifier>C_26984</dcterms:identifier>
  </skos:Concept>
</rdf:RDF>
```