International ALA workshop
3-5 October 2016

Collectory module

Marie-Elise Lecoq
dev@gbif.fr

GitHub

- The Github directory is named collectory-hub
- Fork this folder : http://tinyurl.com/h3kdxh5
- **Rename this fork**
  - –On your newly forked folder, go to the sitting tab and rename it
  - –The GBIF France instance is named gbiffrance-collectory for example

Configuration file

Location : /data/<name_of_your_collectory>/config

You can modify the config file directly on the Grails web-app but the file located in the collectory folder has priority on this file.

- Description of the field in the configuration file http://tinyurl.com/h2hqjk6

Mysql (1)

- The database is created automatically at the first launch of the registry

```
environments {
          development {
                    dataSource {
                        dialect =
               org.hibernate.dialect.MySQL5InnoDBDialect
                        dbCreate = "update" // one of 'create',
               'create-drop','update'
                        url =
               "jdbc:mysql://localhost:3306/collectory?autoReconnect
               =true&connectTimeout=0&characterEncoding=UTF-8"
                        username = ""
                        password = ""
                        logSql = true
                    }
          }
}
```

Mysql (2)

http://docs.grails.org/latest/guide/conf.html

- create - Drops the existing schema and creates the schema on startup, dropping existing tables, indexes, etc. first.

- create-drop - Same as create, but also drops the tables when the application shuts down cleanly.

- update - Creates missing tables and indexes, and updates the current schema without dropping any tables or data. Note that this can't properly handle many schema changes like column renames (you're left with the old column containing the existing data).

Webservices

Access to the webservice :

http://URL/collectory/ws

Example	URL
Administration page	http://metadonnee.gbif.fr/ws/
Institution	http://metadonnee.gbif.fr/ws/institution/in1
Total of Institution	http://metadonnee.gbif.fr/ws/institution/count

How to publish data through your instance ?

1. Download DwC-A (1)

All data from one country:

a. Go to /URL_TO_ALF/manage/gbifLoadCountry

a. Choose country and how many datasets you want to download

a. You can also check the “Reload existing resources” if you want to reload everything from this country

Link to the GBIF France page : http://tinyurl.com/juj9hb2

A dataresource is created. Keep the dataresource ID. We will need it during the ingestation.

1. Download DwC-A (2)

One specific dataset:

a. Download a specific dataset from GBIF

a. Copy the first link from the email received from GBIF

a. Go to /URL_TO_ALF/dataResource/gbifUpload

a. Paste the link to the only field of this page

a. If the download works, you will see the new dataresource (with the name of this one)

A dataresource is created. Keep the dataresource ID. We will need it during the ingestation.

1. Download DwC-A (3)

From the IPT:

a. Go to http://URL_TO_ALF/dataProvider/list

a. Add a new dataProvider using the “Add a new dataProvider” button

a. In the IPT Integration part, fill the name and URL of the IPT

a. Click on the “Check endpoint” option in the same part

a. Upload data resources

A dataresource is created. Keep the dataresource ID. We will need it during the ingestation.

2. Get provider codes (1)

Help to map the dataResource to the institution and/or the collection

SOLUTION 1

At the GBIF France, we follow these steps :

1. Download the DwC-A in local

1. Open the resource file using Excel

1. Add filters to the file

1. Go to institutionCode and collectionCode

1. Click on the column name for knowing the different codes used by this dataset

IMPORTANT : works with resources having a number of occurrences lower than the maximum number of lines in Excel (around 1 million)

2. Get provider codes (2)

SOLUTION 2

You can also run the indexation, there is an info line with this information :

1. INFO : [DataLoader] - The current institution codes for the data resource: Set(BRC Small grain cereals, INRA Clermont-Ferrand)

1. INFO : [DataLoader] - The current collection codes for the data resource: Set(Barley French national collection)

1. Enter those codes to your platform

3. Enter metadata

- Fill in the metadata for the newly created dataresource.

- If the dataResource is a link to a collection :
  - if it doesn’t exist: create the collection page.
  - if it does exist: add the collection to the record consumer of this dataResource

- Link the dataResource to an institution :
  - if it doesn’t exist, create the institution page by filling in the metadata.
  - if it exists, add the institution to the record consumer of this dataResource.

- Don’t forget to go back to the dataResource page to fill in the record consumers if the collection or/and institution doesn’t exist.

4. Create the mapping

- If the specific providerCodes do not exist, enter them using “Manage provider code” page :

```
http://URL_TO_ALF/providerCode/list
```

- Create the provider map between the data resource, the institution and/or the collection in the following page :

```
http://URL_TO_ALF/providerMap/list
```

- Example : http://tinyurl.com/jbzndcb

4. Ingest the dataset (1)

IN CASE OF A SMALL DATARESOURCE (< 50 000 records)

1. Connect to the server where your biocache tool is hosted

1. Connect to biocache

```
$ sudo biocache
```

1. Run the following command line:

```
$ ingest -dr <dataresource_id>
```

- you can also run directly on the terminal

```
$ sudo biocache ingest -dr <dataresource_id>
```

but you will not have logs if you don’t specify the out file.

4. Ingest the dataset (2)

IN CASE OF A BIG DATARESOURCE (> 50 000 records and < 8 millions)

- Connect to the server where your biocache tool is hosted
- Connect to biocache : $ sudo biocache
- Run the following command lines:

```
1. $ load <dataResource_id>
2. $ process -dr <dataResource_id>
3. $ sample -dr <dataResource_id>
4. $ index -dr <dataResource_id>
```

- You can also run directly on the terminal, the command lines above with `$ sudo biocache` before it as explained in the slide before.

4. Ingest the dataset (3)

IN CASE OF A REALLY REALLY BIG DATARESOURCE (DwC-A size > 1 Go)

1. Upload a modified DwC-Archive with 15 occurrences in order to create the dataset into the system.

1. Copy the real DwC-Archive instead of the modified one on the `/collectory/upload/` folder

1. Then run the load, process and index commands.

- We need to do step 1 because the ZipFile library used by the biocache-store can't open a file bigger than 1 GO

- You need to have a server with at least the size of your DwC-Archive in RAM.

4. Ingest all of your resources

- You can run one command line (as sudo user):

```
1. $ nohup biocache ingest -all >
   /tmp/load.log &
```

- You can run three different command lines directly on the terminal (as sudo user):

```
1. $ biocache bulk-processor load -t 7 >
   data/output_load.log
2. $ biocache bulk-processor process -t 6
   > data/output_process.log
3. $ biocache bulk-processor index -ps
   1000 -t 8 > /data/output_index.log
```

- With the -t option, you will give the number of CPU you want to use for the processus.
- With the -ps option, you will give the number of occurrences per pages on SOLR.

GBIF
www.gbif.org

gbif France

Customization

Add filters (1)

- At GBIF France, we had to add filters to the collectory public page: paleontology and fungi

- Modification made on the following files:

File name	Folder
map3.gsp	/views/public
message.properties	/i18n/
message_fr.properties	/i18n/
generic.css	/web-app/css/
show.gsp	/views/collection/
map.js	/web-app/js/

Add filters (2)

map3.gsp

```
<div class="fungis" id="fungis"
  onclick="toggleButton(this);return false;">
      <h2>
          <a href="" class="filtre-meta">
               <g:message code="public.map3.link.fungus" />
               <span style="">
                    <g:message
                      code="public.map3.link.fungi" />.
               </span>
          </a>
     </h2>
</div>
```

Access to the file in the GBIF France Github
http://tinyurl.com/jsy2bvv

Add filters (3)

generic.css

```
filter-buttons .fungis a {
        background: transparent
           url(../images/ala/mycetes_button_bgwd.png) no-
           repeat;
        background-size: 100%;
}
.filter-buttons .paleontology a {
        background: transparent
           url(../images/ala/paleontology_button_bgwd.png) no-
           repeat;
        background-size: 100%;
}
```

Access to the file in the GBIF France Github
http://tinyurl.com/h5rjust

Add filters (4)

show.gsp

```
<g:if test="${!instance.listKeywords().contains('paleo')}">
    (<g:link action="addKeyword" id="${instance.id}"
    params="[keyword:'paleo']">
        <g:message code="collection.show.link.addfungis"
        />
    </g:link>)
</g:if>

<g:if test="${!instance.listKeywords().contains('fungi')}">
    (<g:link action="addKeyword" id="${instance.id}"
    params="[keyword:'fungi']">
        <g:message code="collection.show.link.addpaleo" />
    </g:link>)
</g:if>
```

Access to the file in the GBIF France Github
http://tinyurl.com/ztr4bzg

Add filters (5)

map.js

```
if (checked ==
'fauna,entomology,microbes,plants,paleontology,fungis') {
        checked = 'all';
}
```

Access to the file in the GBIF France Github
http://tinyurl.com/jep6pt5

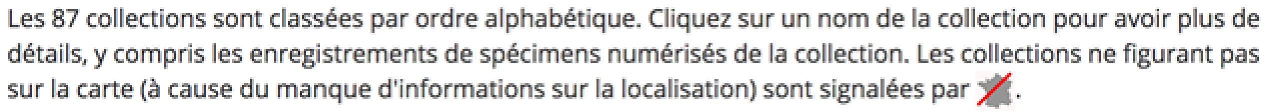
Internationalization (1)

- There are differences between the English sentence structuration and the French one.
- Modifications needed directly on the code and creation of new terms in messages.properties
- Possibility to add a multi-languages selector

Internationalization (2)

Example of modification linked to the internationalization

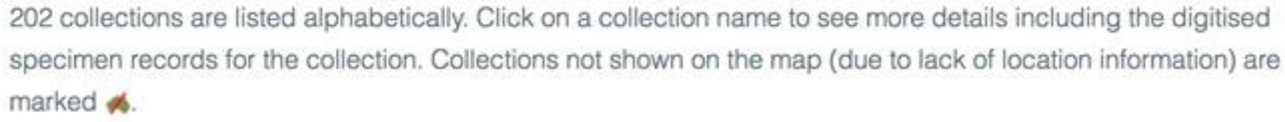
- French version (on ALF)



Les 87 collections sont classées par ordre alphabétique. Cliquez sur un nom de la collection pour avoir plus de détails, y compris les enregistrements de spécimens numérisés de la collection. Les collections ne figurant pas sur la carte (à cause du manque d'informations sur la localisation) sont signalées par .

```
case 0: innerFeatures = jQuery.i18n.prop('map.js.nocollectionsareselected');  break;
case 1: innerFeatures = features.length  + " " + jQuery.i18n.prop('map.js.collectionislisted');
break;
default: innerFeatures  = "Les "+ features.length  + " " +
jQuery.i18n.prop('map.js.collectionsarelistedalphabetically');  break;
```

- English version (on ALA)



202 collections are listed alphabetically. Click on a collection name to see more details including the digitised specimen records for the collection. Collections not shown on the map (due to lack of location information) are marked .

```
case 0: innerFeatures = jQuery.i18n.prop('map.js.nocollectionsareselected');   break;
case 1: innerFeatures = jQuery.i18n.prop('map.js.onecollectionisselected');   break;
default: innerFeatures  = features.length  + " "+ selectedFrom + "."; break;
```

Internationalization (3)

Example of a multi-language selector (1)

1. Include in BuildConfig.groovy the next code (Source: http://grails.org/plugin/lang-selector)

```
runtime ":lang-selector:0.3"
```

1. Add the tag to your gsp (in the header section of your layout if you want to be seen on all of the pages):

```
<ul class="nav pull-right">
    <li class="dropdown">
      <a class="dropdown-toggle" data-toggle="dropdown"
href="#">Language<b class="caret"></b></a>
      <ul class="dropdown-menu">
                <li><langs:selector langs="ca"/></li>
                <li><langs:selector langs="en"/></li>
      </ul>
    </li>
</ul>
```

1. Optionally you can add this property to the `/grails-app/conf/Config.groovy` to tell the plugin which flag display for the language :

```
com.mfelix.grails.plugins.langSelector.lang.flags = ["es":"es",
"en":"gb", "fr":"fr", "da":"dk", "de":"de", "it":"it",
"ja":"jp", "nl":"nl", "ru":"ru", "th":"th", "zh":"cn",
"pt":"pt", "ca":"catalonia"]
```

Internationalization (4)

Example of a multi-language selector (2)

1. Include in BuildConfig.groovy the next code (Source: http://grails.org/plugin/lang-selector)

```
      runtime ":lang-selector:0.3"
```

1. Add the code from this gist file in `/lang-selector-0.3/grails-app/views/langSelector/_selector.gsp`

http://tinyurl.com/zn4h5pv

- These two examples are based on the Encounter Bay document written by GBIF Spain, GBIF Portugal, GBIF Belgium and GBIF France during a mentoring.

Pratical work

Merci !

GBIF France’s IT Team

dev@gbif.fr

Fabien Caviere
caviere@gbif.fr
https://github.com/CaviereFabien

Marie-Elise Lecoq
melecoq@gbif.fr
https://github.com/melecoq
@bouhoub

Github account

https://github.com/gbiffrance

GBIF
www.gbif.org

gbif
France