



The Encounter 1802

The Encounter of Matthew Flinders and Nicolas Baudin in 1802, by John Ford

Atlas of Living Australia

Key Technical Document

Version: 1.0.4, 6 July 2016

This document is a contribution to the Encounter Bay (ALA node portal internationalization) project, co-funded by the French, Spanish, Portuguese and Belgian GBIF Nodes and GBIF Secretariat through its Capacity Enhancement Support Programme.

Authors (in alphabetical order)

Fabien Cavière, GBIF France, caviere@gbif.fr

Rui Figueira, GBIF Portugal, ruifigueira@isa.ulisboa.pt

André Heughebaert, GBIF Belgium, a.heughebaert@biodiversity.be

Marie-Elise Lecoq, GBIF France, melecoq@gbif.fr

Santiago Martínez de la Riva, GBIF Spain, sama@gbif.es

Audience

IT specialists who wants to get a first understanding of the technology behind the Atlas of Living Australia (ALA) biodiversity data portal. The reader is expected to have at least a moderate understanding of web services, databases and web developments in general.

Aims

This document is intended to capture the essence of the technical documentation about ALA components necessary for evaluators/adopters to discover, assess, understand, install, configure, customize and operate the tools.

License

This public document falls under [CC-BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/) license.



Contact information/disclaimer

This document was elaborated by the early adopters of the ALA technology (the French, Portuguese and Spanish GBIF nodes), not by the Australian development team. Therefore most of the screenshots and examples will not refer to Atlas of Living Australia data portal, but to early adopters portals.

For more information, please contact info@gbif.es.

Endorsement of this documentation by the Atlas of Living Australia

The Atlas of Living Australia (ALA) are pleased to endorse this document which is the result of a collaborative effort from several GBIF Nodes. We consider this a useful introduction to the tools for any national Node, or indeed any group interested in setting up a thematic portal. We see this global collaboration as the first of many joint activities around the Atlas tools.

Table of Contents

1	Introduction.....	5
1.1	How to use this document.....	5
1.2	ALA derived portals.....	6
2	Architecture.....	7
2.1	Rationale of the ALA software design.....	7
2.2	Modules overview.....	9
2.3	Occurrence search.....	10
2.4	Collectory.....	12
2.4.1	Public pages.....	12
2.4.2	Private pages.....	12
2.5	Species page.....	14
2.6	Data Hub.....	15
2.7	Regions.....	19
2.8	Spatial Portal.....	20
2.9	Technical basic principles.....	22
3	Requirements and installation.....	22
3.1	Test server requirements.....	23
3.2	Production server requirements.....	23
3.3	Others example of System Architecture.....	26
3.3.1	Australia.....	26
3.3.2	France.....	27
3.4	Installation process (Test environment).....	27
3.4.1	Vagrant.....	28
3.4.2	Ansible.....	29
3.4.3	Terminating the VM.....	31
3.5	Installation process on Remote servers (Production environment).....	32
3.5.1	Understanding the concept.....	32
3.5.2	Installation steps.....	32
3.5.3	How to install services on several servers using Ansible playbooks.....	33
4	Configuration.....	34
4.1	Github.....	34
4.2	Storage.....	35
4.2.1	MySQL.....	35
4.2.2	Cassandra.....	35
4.3	Files architecture.....	36
4.4	Configuration file.....	37
4.5	Styling the application.....	38
4.6	Internationalization.....	45
5	Loading data.....	45
5.1	Type of data.....	46
5.1.1	Collection.....	47
5.1.2	Data provider.....	47
5.1.3	Institution.....	48

5.1.4 Data Hub.....	48
5.1.5 Contact.....	48
5.2 Configuration data.....	49
5.2.1 Provider Code.....	49
5.2.2 Provider Map.....	49
5.3 Data resource.....	50
5.3.1 External data resource.....	50
5.3.2 GBIF resources.....	51
5.3.3 Using IPT.....	51
6 Backing-up and updating.....	52
6.1 Back-up databases.....	52
6.1.1 Backing up MySQL database.....	52
6.1.2 Backing up Cassandra data store.....	52
6.1.3 Making a snapshot.....	52
6.1.4 Restoring a snapshot.....	53
6.1.5 Backing up SOLR data store.....	54
6.2 Updating tools.....	56
7 Functionalities.....	57
7.1 Data Visualization.....	57
7.1.1 Searching occurrences.....	59
7.1.2 Searching metadata.....	62
7.2 Import features.....	69
7.3 Export features.....	70
7.4 Metadata.....	70
7.4.1 Contacts.....	70
7.4.2 Providers codes.....	71
7.4.3 Providers maps.....	71
7.5 Error detection and reporting.....	72
8 Further readings.....	76
8.1 About Atlas of Living Australia software.....	76
8.2 Development technologies and frameworks.....	76
8.3 Data management.....	77
8.4 Biodiversity data formats.....	77
9 Internationalization.....	78
9.1 About the ALA portal User Interface.....	78
9.2 Step-by-Step guide to UI translation.....	79
9.3 Language configuration.....	83
9.4 Multi-lingual support.....	83
10 Collaboration Channels.....	86

1 Introduction

The Atlas of Living Australia (ALA) project is a National Research Infrastructure for Australia which has built an infrastructure and tools to enable researchers and other users of biodiversity information to find, access, combine and visualize data on Australian plants and animals.

The software produced by the ALA project has progressively been used outside of the original ALA team and became an open source software project.

If you want to get involved and follow future developments, please refer to the Collaboration channels chapter.

This document gives general ideas of the software principles, features and possibilities. The reader will find further details on the more elaborated [wiki documentation](#).

1.1 How to use this document

This key technical document provides an overview of the architecture, the functionalities, the tools/technology dependencies that facilitate the assessment of the ALA software.

Structure of this document:

- Chapter 1: [Introduction](#)
- Chapter 2: [Architecture](#)
- Chapter 3: [Requirements and installation](#)
- Chapter 4: [Configuration](#)
- Chapter 5: [Loading data](#)
- Chapter 6: [Backing-up and updating](#)
- Chapter 7: [Functionalities](#)
- Chapter 8: [Further readings](#)
- Chapter 9: [Internationalization](#)
- Chapter 10: [Collaboration channels](#)

It answers some preliminary questions the reader might have:

- What is the ALA data portal? (read [Functionalities](#) and [Architecture](#))
- How to install it? (read [Requirements & Installation](#))

- How to configure it? How to style/theme it? (read [Configuration](#))
- How to populate it with data? (read [Loading data](#))
- How to update it and keep local development in sync with the global repository? (read [Backing-up and updating](#))
- Where do I get more detailed information? (read [Further readings](#))
- How to translate ALA User Interface (read [Internationalization](#))
- How to join this community? (read [Collaboration channels](#))

1.2 ALA derived portals

You will find below a table with information about nodes that installed and configured data portal using ALA modules. You can contact them directly if you want more information about their installation.

Nodes	Website	Contact	Langue
Atlas of Living Scotland	http://www.als.scot/	info@als.scot	English
GBIF France	http://portail.gbif.fr/	dev@gbif.fr	France
GBIF Spain	http://datos.gbif.es/	info@gbif.es	Spanish
SiBBr	https://portaldabiodiversidade.icmbio.gov.br/portal/		Portuguese
GBIF Portugal	http://dados.gbif.pt	node@gbif.pt	Portuguese

2 Architecture

2.1 Rationale of the ALA software design

The Atlas software has been built upon the following architectural principles:

1. **Service oriented** - all content available to users on the Atlas webpages is available through an underlying web service. These web services make up the public API for the Atlas. This is to promote open data and enable Atlas partners to benefit from these systems to embed elements of the Atlas in their sites and tools.
2. **Modular components** - The Atlas is made up of a set of micro service components, each with a specific role. This helps promote reuse and enables flexibility in deployments.
3. **Re-useable UI modules** - The user interface components are based on a pluggable architecture. This is aimed to enable re-use and enable customization. We recognize that a customized brand is an important part of producing a portal for a given community.
4. **Portability** - The atlas is built exclusively on open source software. All Atlas applications can be installed using a set of maintained scripts to aid adoption of Atlas components.

The Atlas of Living Australia platform (ALA) is divided in different modules that can work together, but can also work apart (e.g. in France, the team just installed three of them). The diagram below shows you the entire architecture of the ALA system.

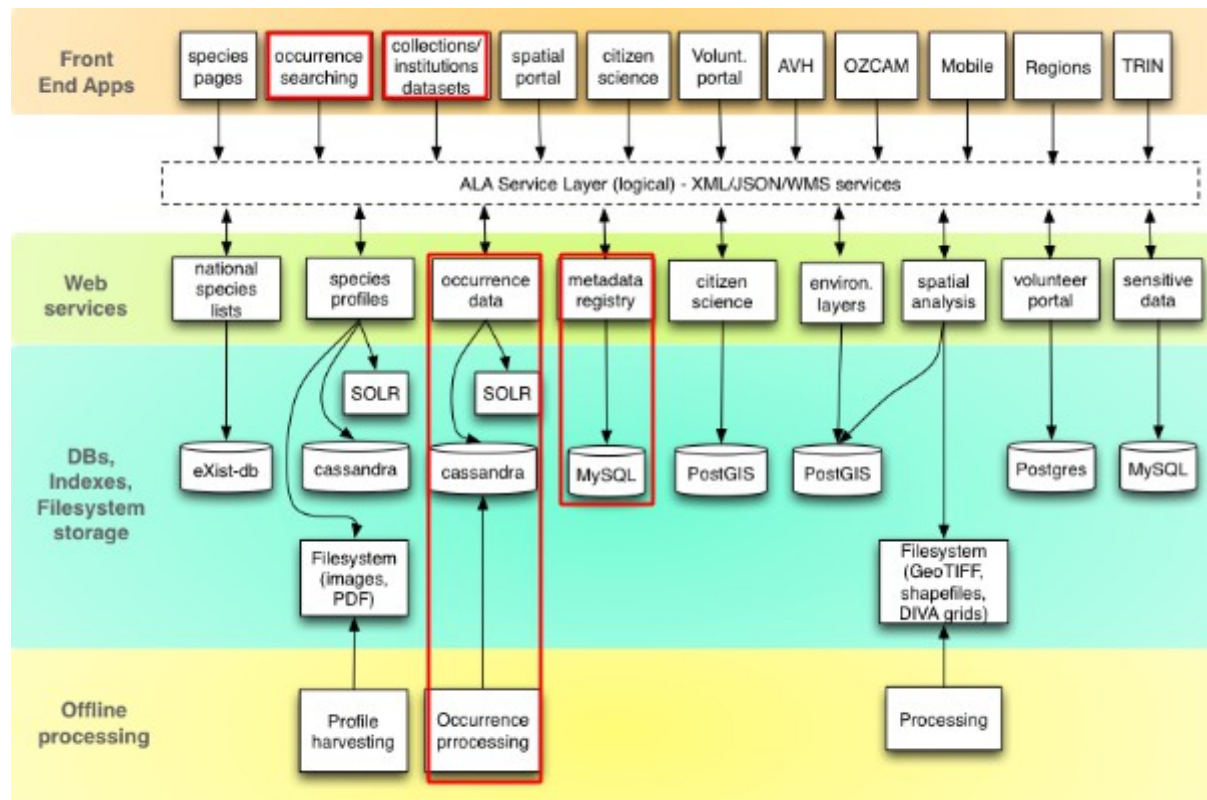


Illustration 1: ALA Architecture

In this chapter, we will focus on the occurrence and collection modules (marked in red in Illustration 1), but we will also make a short description of the other components that the ALA project teams developed. All these items can be installed together or separately. We will also describe the technical basic principles (e.g. language and software used) and advise on the server configuration needed with examples from countries that already installed it.

2.2 Modules overview

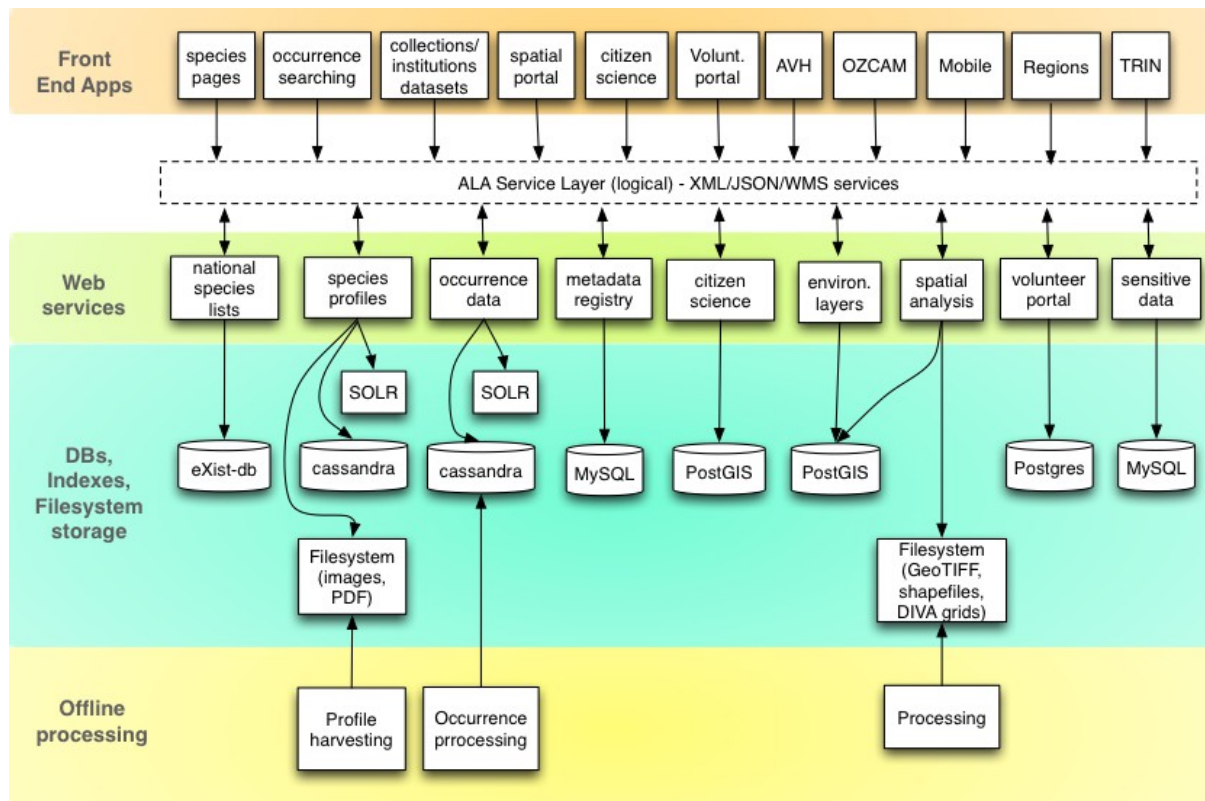


Illustration 2: Modules overview

2.3 Occurrence search

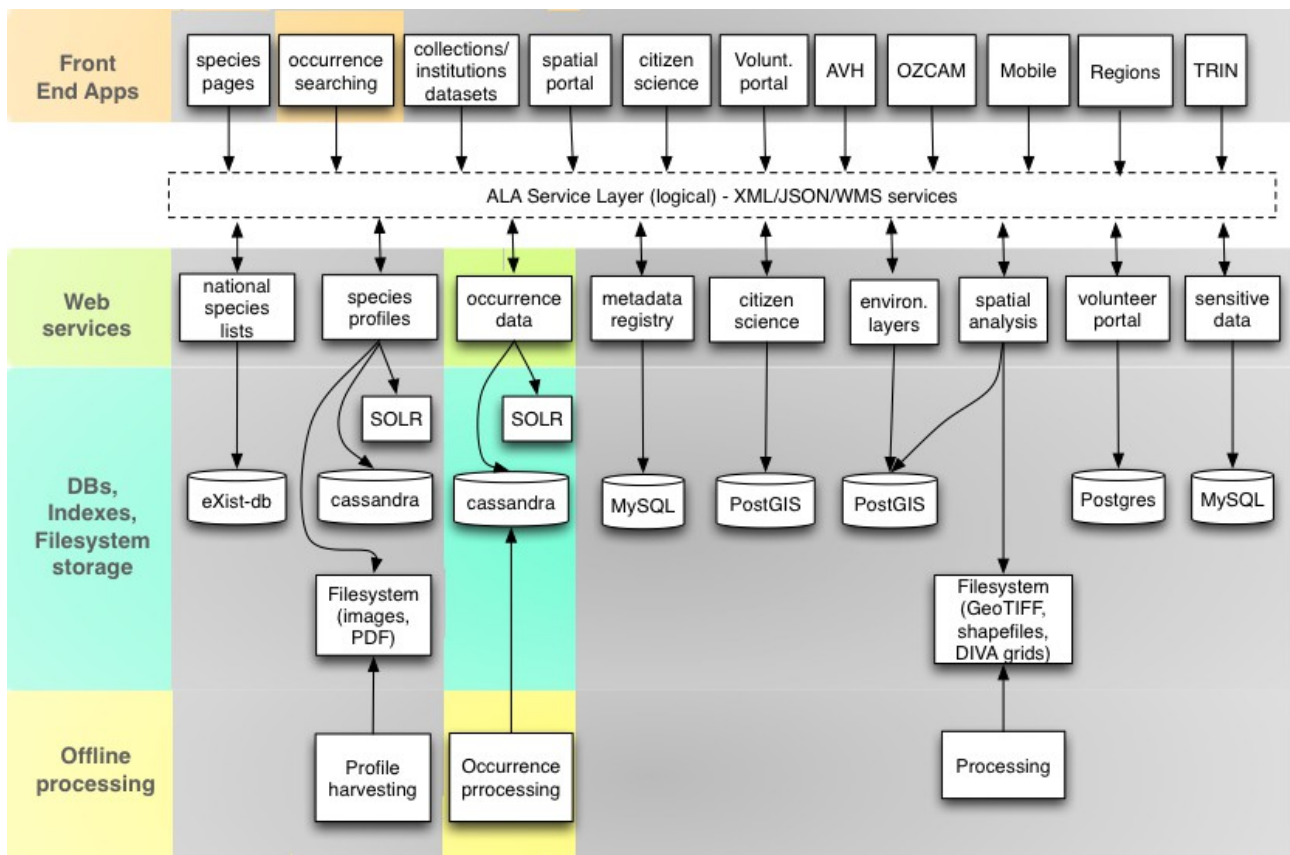


Illustration 3: Occurrences search modules

The occurrence searching is the data search engine of the portal. As you can read on the functionalities chapter, there are different ways to make a search (advanced, by taxon, simple or using bounding box).

The results page shows a list with all the results, diagrams as well as a map with the georeferenced occurrences. In the occurrences page, you will find information about it after the indexing. You will also be able to go the institution and/or collection pages (that part of the metadata registry). It is important to notice that all components of the ALA software are linked.

Technically, you have a web application named biocache-hub that provides a HTML user interface. This component makes use of web services provided by a component called biocache-service (occurrence data on the diagram). Data is stored on a Cassandra database, the file system is called occ and ALA uses Apache SOLR to index the occurrence data.

In order to process data into SOLR, the developer team of ALA project created an indexing tool named Biocache. It also works with the webservice biocache-service.

Basically, this code base manages the loading, sampling, processing and indexing of occurrence records for the ALA platform. There are additional tools to support outlier detection, duplicate detection and identifying extra-limit outliers based on authoritative distribution polygons for taxa.

2.4 Collectory

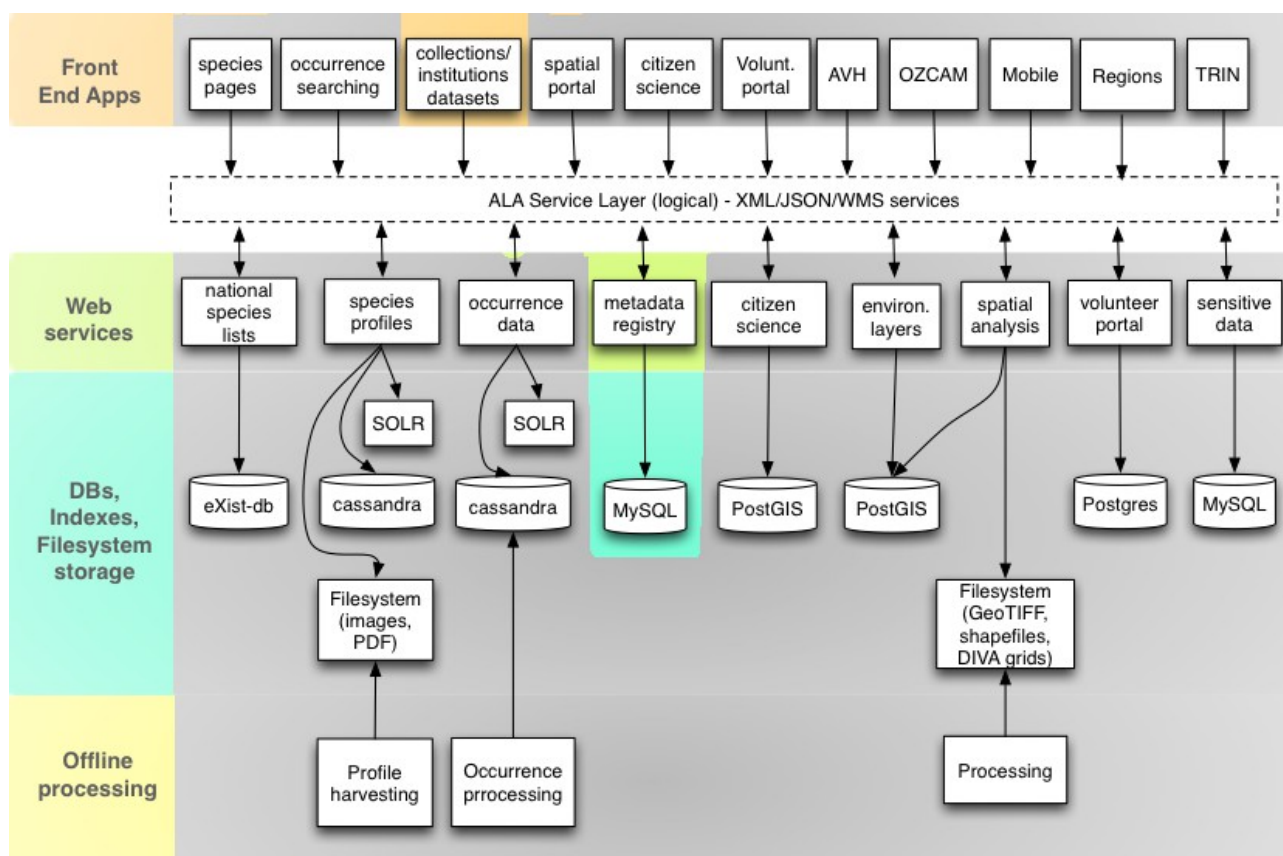


Illustration 4: Collectory modules

The collectory-plugin is the module that manages the registry metadata (information about institutions, collections, data resources, contacts, etc.). All data are stored on a MySQL database named collectory. This module is divided in two parts: public and private pages.

2.4.1 Public pages

You will find all of the public page of institutions or collections, but also a search engine with a map focus on the collection.

2.4.2 Private pages

This is the administration part. You will be able to add, modify or delete institutions, collections, data resources or contacts. You will also be able to see a report on your portal. If you want to add data resources indexed by the global GBIF portal GBIF.org, you will find all of the tools needed in those private pages.

This application can be hosted in an independent server.

Technically, there is just one web application and one MySQL instance. This is one of the simplest architecture of the entire system. You will find highlighted the elements used.

2.5 Species page

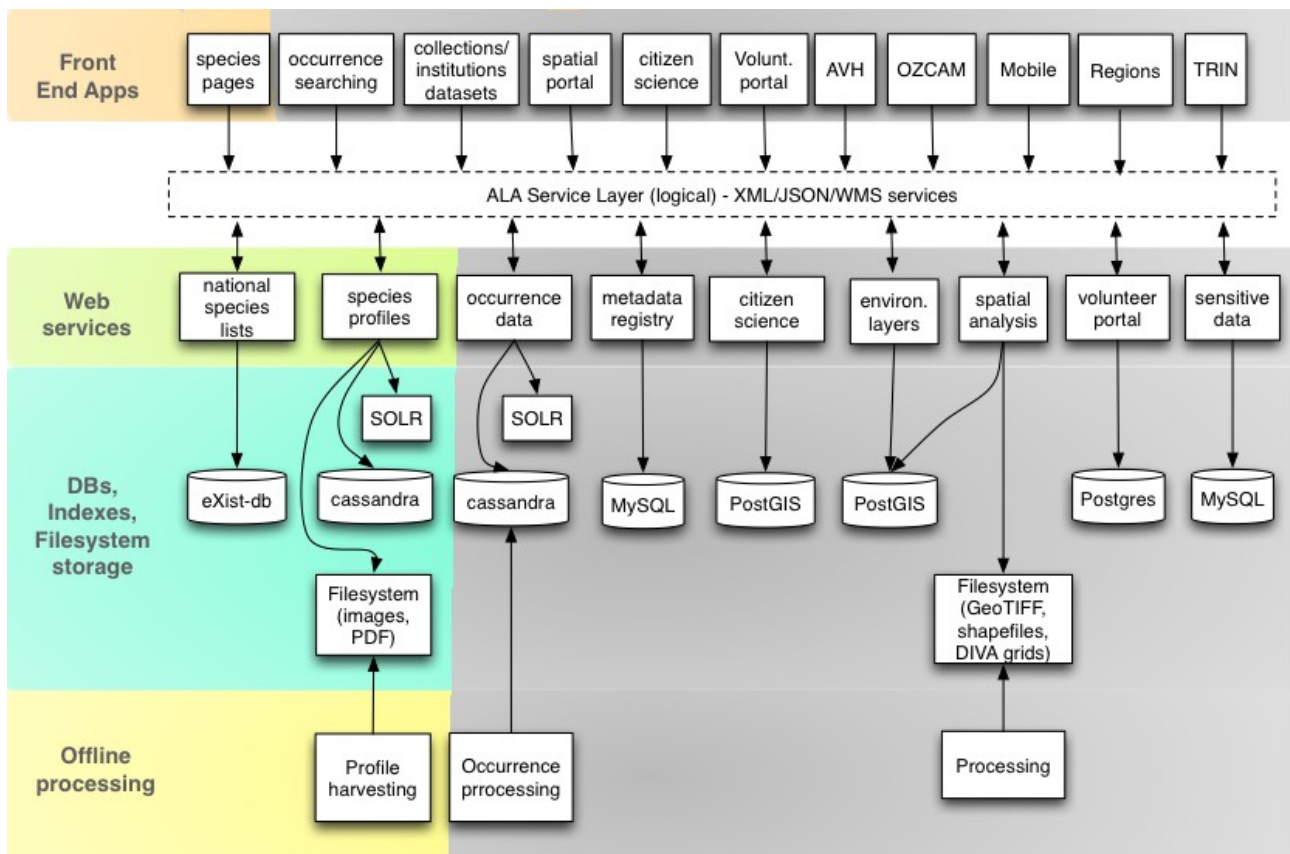


Illustration 5: Species modules

This page allows the user to browse existing species that are hosted in the data portal. You can filter the list of species by Section, LifeForm, Species groups and Conservation status. For each species page the user can view the information about the Names, Classification, Records, Literature and Sequences. The system creates a vinculation between the occurrences and

their species page, in this way in you are in one specific species page you can view the records related with that species. You can find examples here:

<http://bie.ala.org.au/> and

<http://bie.ala.org.au/species/Carcharhinus+amblyrhynchos>.

Technically, it is developed with Grails Framework and it is using the REST Web Service of your portal.

2.6 Data Hub

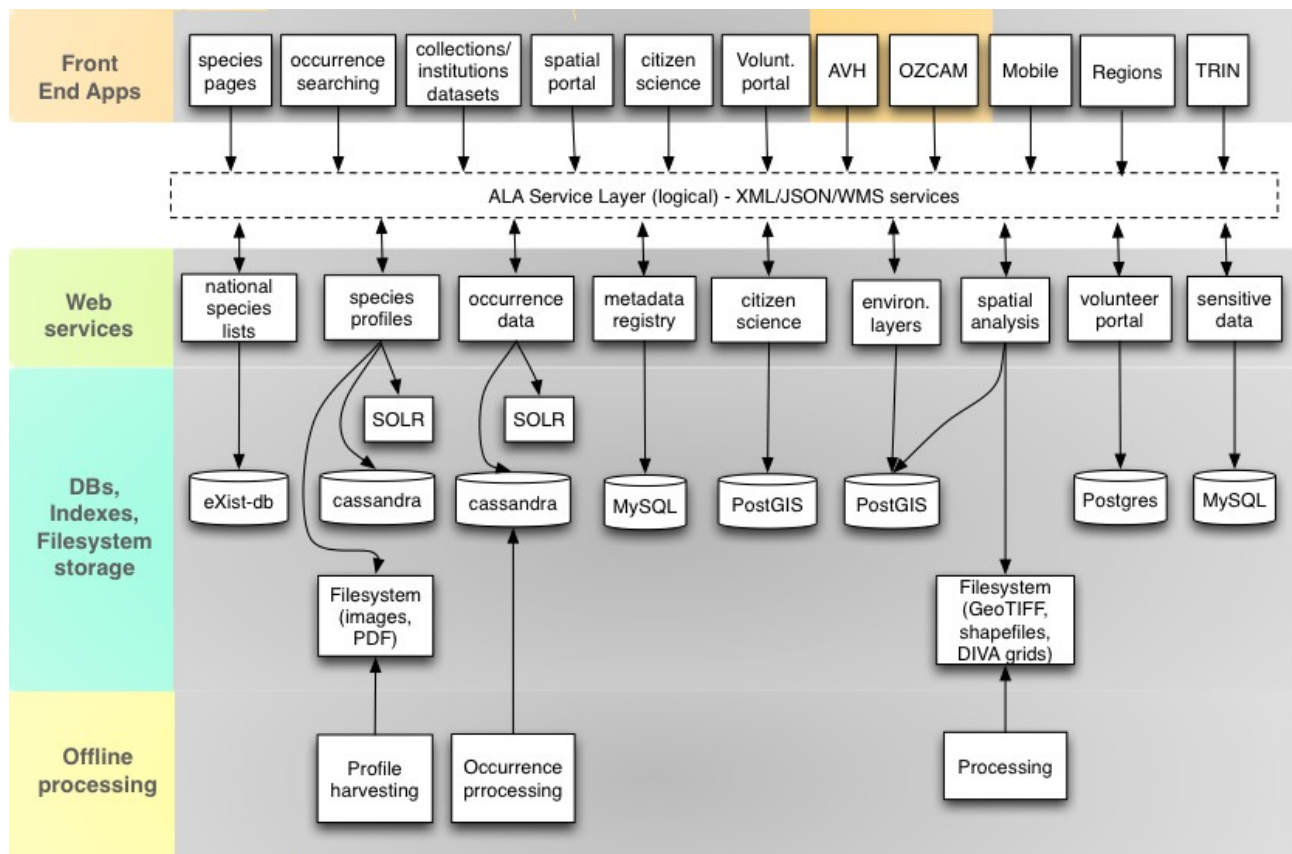


Illustration 6: Datahub modules

The first thing that you have to understand is the concept of Data Hub and you can understand it very easily with one graphic example.

In GBIF Spain node, you have the main data portal: <http://datos.gbif.es>, through this portal users can see all the records that spanish providers publish.

What happens if one of these providers wants to show a subgroup of these records in their own institutional web?

For these cases, the provider can use the Data Hub web application as you can see here:

http://161.111.171.47:8080/myxo-hub/search#tab_simpleSearch

Español | English

MYXOTROPIC

[Inicio](#) [Buscar](#) [Mapa](#) [Colección](#) [Juego de datos](#) [Datos GBIF.es](#)

Buscador de registros

Mediante este Portal el usuario podrá consultar información de registros de Myxomycetes que viven en el Neotrópico. Para realizar las búsquedas puede hacer uso de los caracteres comodín: * que reemplaza un conjunto de caracteres e ? para reemplazar sólo un carácter.

Búsqueda simple

Búsqueda avanzada


Búsqueda por áreas

Buscar

Nota: la búsqueda simple intenta hacer coincidir una **especie/taxon** conocido - por su nombre científico. Si no hay ninguna coincidencia de nombre, en la consulta se realizará una búsqueda de **texto completo**

REAL JARDÍN BOTÁNICO, CSIC
Plaza de Murillo, 2. Madrid (ESPAÑA)
Teléfono: +34 91 4203017
Fax: +34 91 4200157
Email: Carlos Lado
Web: Myxotropic





Copyright

Proyecto financiado por la Dirección General de Investigación Científica y Técnica de España (CGL2011-22684)

Diseño Raúl de la Fuente | Textos Carlos Lado | Textos en inglés Diana Wrigley de Basanta | Imágenes Carlos de Mier y Carlos Lado | Gestión de base de datos Juan Carlos Hernández |
Desarrollo Santiago Martínez de la Riva

Illustration 7: Myxotropic Datahub

In this Data hub when the users do a query, the search will only find the records that are in the subgroup defined by the provider, in this case, this projects is about the Myxomycetes that lives in the Neotropics, and this group of records is defined in the main data portal, specifically in the data resource below:

<http://datos.gbif.es/collectory/public/showDataResource/dr190>

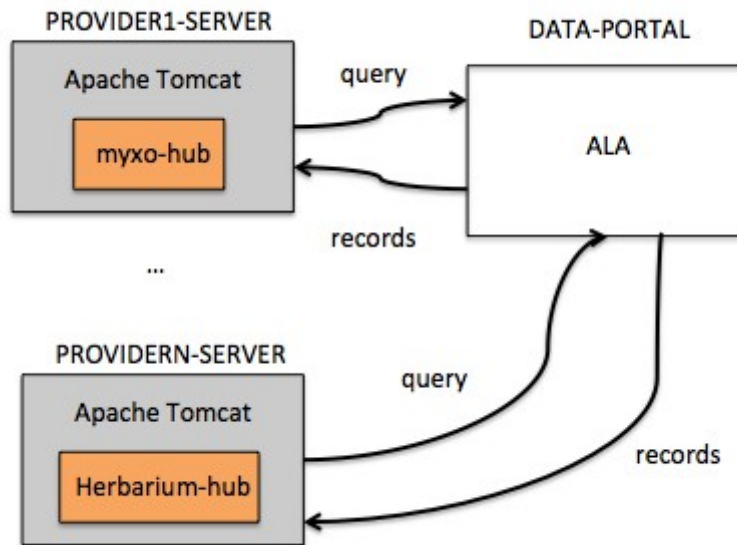


Illustration 9: Data Hubs

You can read how to configure our own Data Hub here:

<https://github.com/AtlasOfLivingAustralia/documentation/wiki/Configuring-generic-hub>

Note: You can access to AVH datahub (<http://avh.chah.org.au/>) and to OZCAM datahub (<http://ozcam.ala.org.au/>)

2.7 Regions

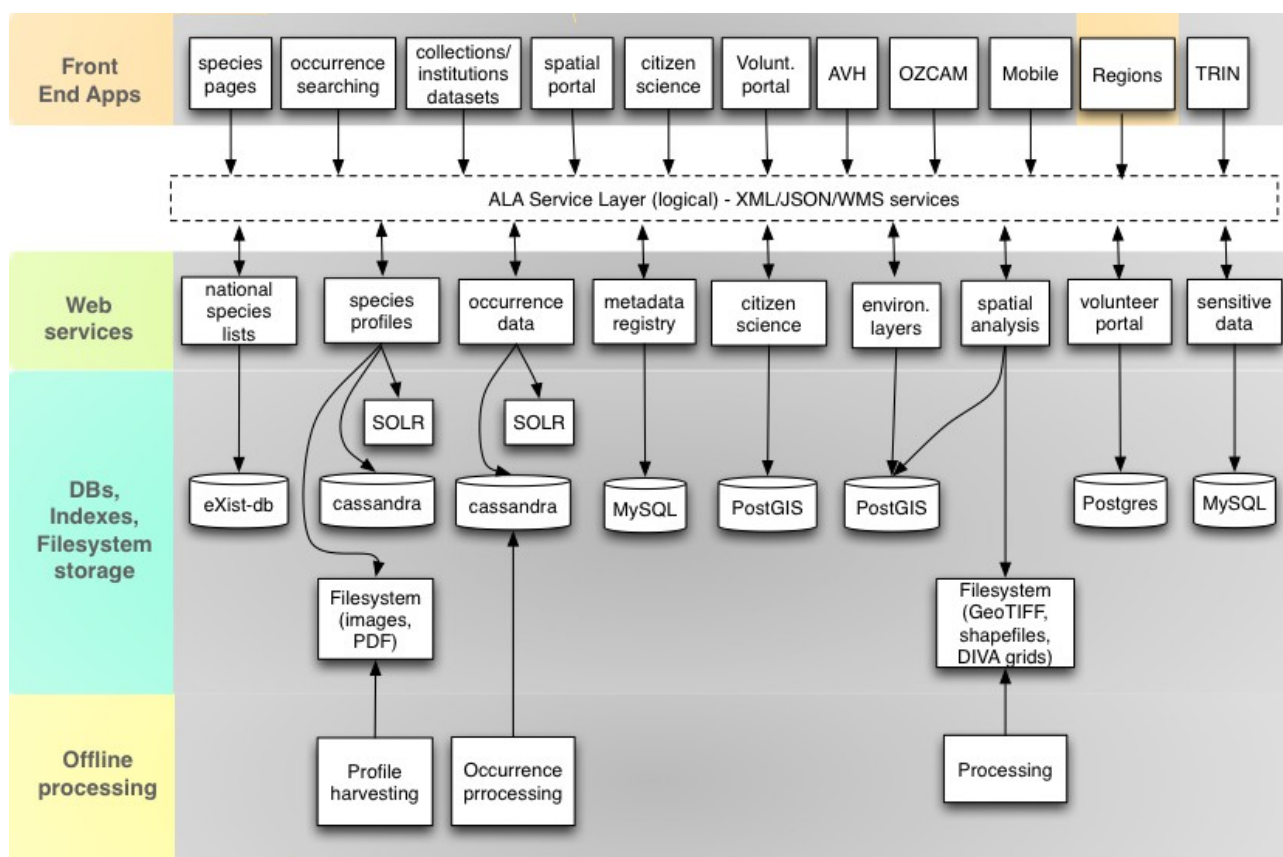


Illustration 10: Regions modules

This project allows the user to conveniently browse existing ALA portal records that have been previously filtered by the user selecting a wide range of meaningful regions displayed on a map.

You can find examples here:

<http://regions.ala.org.au/> and <http://regions.als.scot/>.

Technically, it is developed with Grails Framework and it is using the REST Web Service of your portal. It is also using Google Maps Web API and Google Charts.

It is easier to install than the spatial portal. We invite you to discuss about this with the community if you decide to implement a spatial tool to your portal.

2.8 Spatial Portal

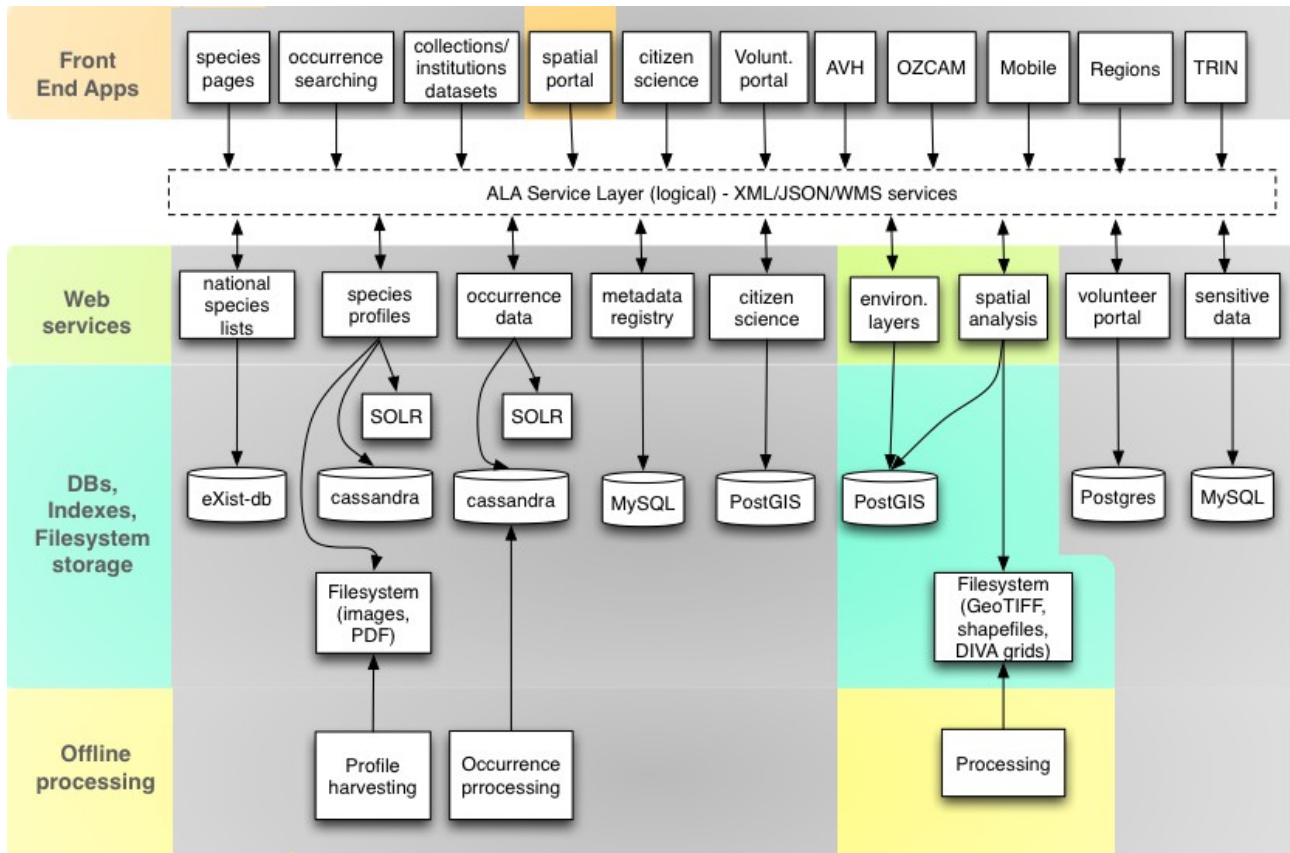


Illustration 11: Spatial portal modules

The ALA Spatial Portal is made up of 4 web applications. These web applications are separated based on their task and utilize web services to communicate with each other when necessary. However, data is shared between each of the web application as necessary, so a pointer to a global storage is necessary.

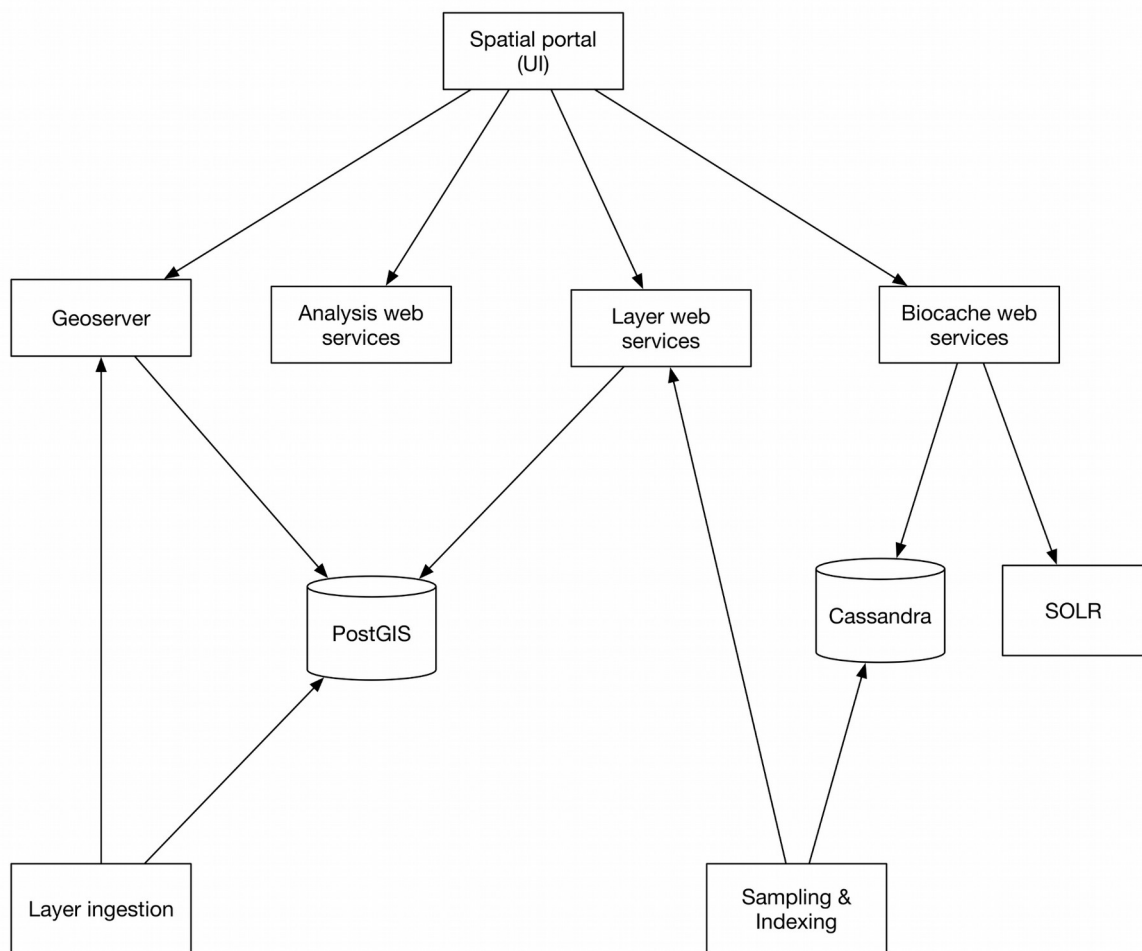


Illustration 12: Spatial Portal

The web applications, components, that make up the whole Spatial Portal experience are:

- ALA Spatial Portal UI (webportal)
- ALA Spatial Analysis Service (alaspacial)
- ALA Spatial Layers Service (layers-service)
- ALA Spatial Actions (actions)

There are a few utilities that are built to be shared between each of the above components:

- ALA Spatial Layers Store (layers-store)
- ALA Spatial Utilities (utilities)

The Portal also utilizes other ALA web services, such as:

- BIE
- Biocache
- Collectory

The SP uses a standard release of Geoserver (version 2.8+) to serve environmental, contextual and dynamic analysis output data via WMS protocol. Please refer to Geoserver for more information at <http://www.geoserver.com/>. A custom, simplified WMS service has been created to serve biodiversity data. This serves the occurrence data as points or grid cells (e.g. 1 degree, latitude/longitude), with some expert distributions for species served as polygons. Please refer to <http://spatial.ala.org.au/ws/> for more information. Example usage of loading biodiversity data is available at <http://spatial.ala.org.au/ws/examples/>.

More information on the GitHub of ALA project (wiki page dedicated to the spatial portal)

2.9 Technical basic principles

For the collectory and the occurrence research, ALA uses Grails, a web framework as main tool. It is based on the Groovy language and it works with Java and hibernate among others. Modules are Web application Archive (WAR) and we use Tomcat 7 as web server.

The database used by the collectory module is MySQL. The occurrence search use Apache SOLR for the indexation on top of the Apache Cassandra for the storage. You will find more information in the [Requirements and installation](#) and [Configuration](#) chapters.

In order to configure or styling web applications used in this project, you should have a reasonable level of knowledge of HTML and CSS as well as a basic understanding of Grails and Bootstrap.

3 Requirements and installation

The ALA Portal requires several components like Java, Tomcat and Cassandra, as well as web applications that composes the ALA Portal. These softwares can be automatically installed and configured by an [Ansible](#) playbook, a type of script, developed together with the project. So if you've got an Ubuntu Linux instance up-and-running, you can use Ansible playbooks in the [ala-install](#) project to automatically set up and configure the Linux instance into an ALA Portal.

Besides Ansible playbooks, if you don't have a Linux instance ready, or just want to set up a clean one for this ALA Portal, you can consider the following two tools to create a clean instance:

1. [Vagrant](#): Create and configure a virtual machine to host the ALA portal;
2. [VirtualBox](#): The container of virtual machines;

3.1 Test server requirements

- Virtual machine with Ubuntu 12 or 14.
- 100GB of free disk storage (ideally SSD), the indexing and processing of data consumes space very fast.
- 32 GB RAM.
- 2 CPUs.

The [installation guide](#) assumes one wants to create a Linux instance and configure an ALA portal from scratch. If you want to configure an existing Linux instance (Ubuntu), you may go straight to the [Ansible section](#) of the installation guide. This tutorial can only be run on UNIX systems (Linux/Mac OS X), given the fact that Ansible is not yet available on Microsoft Windows as 20 May 2016. (http://docs.ansible.com/ansible/intro_installation.html)

Using a Macintosh as an example, here are steps to get these tools ready:

Download [Vagrant](#) and install the downloaded package.

Download [VirtualBox](#) and install the downloaded package.

To install Ansible (version 2.0.1.0+), the easiest way is to install via [homebrew](#). It's also handy to have the [Command Line Tools for Mac OS X](#) installed. Once ready, the following commands will install Ansible:

```
$ brew update
$ brew install ansible
```

However, in this guide, the Vagrantfile will determine the server configurations for you. Now you're ready to start the actual installation of the ALA portal.

3.2 Production server requirements

If you want to deploy the ALA portal on a complex infrastructure, you can use several Virtual Machines to organize the different services: Tomcat, Solr, Mysql, Cassandra, Image service...

Note: You can see the [Installation](#) chapter to understand how to install playbooks in different servers.

In the below image, you see what could be the requirements that you need in each Virtual Machine (Spanish infrastructure):

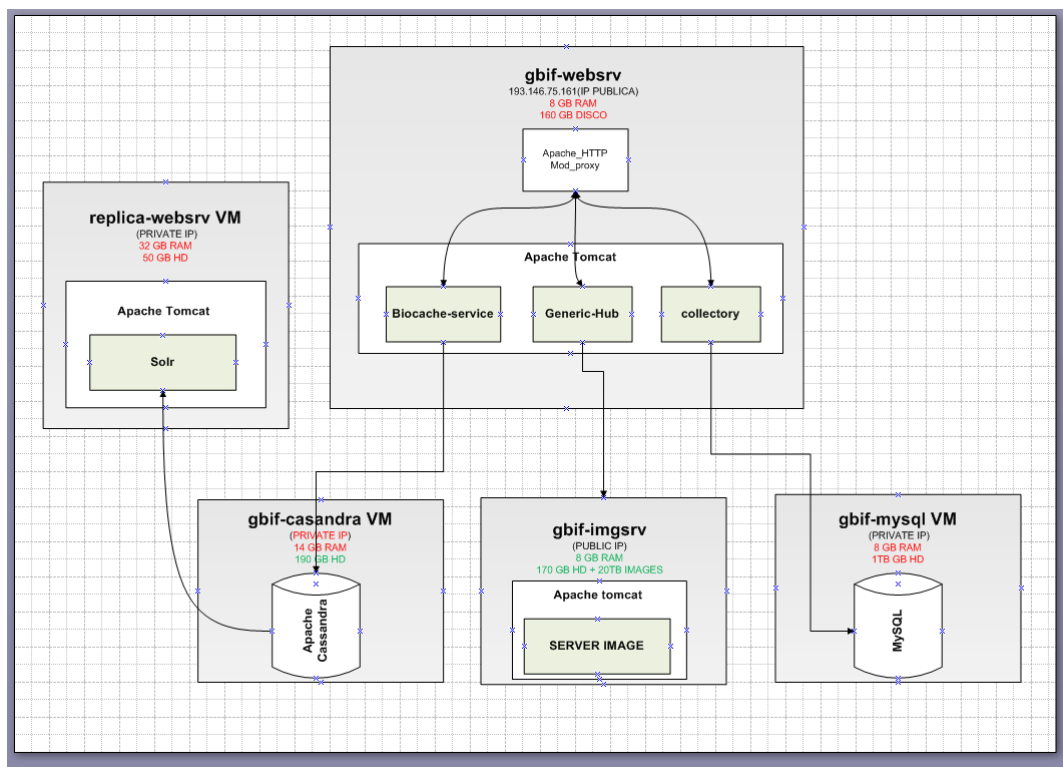


Illustration 13: Spanish data portal infrastructure

The requirements are closely related with the number of records, that you want to publish, and obviously it will be related with the number of visits that you could have in your data portal.

In the case of Spanish node, that at the moment of writing this text has 12 millions of records, 189 datasets and 89 institutions, the main virtual machine is "gbif-websrv" that contains the service of Apache and Tomcat.

This server shows us the home page <http://datos.gbif.es> and redirects the calls on tomcat service to port :80.

Tomcat service contains the web applications of ALA portal:

- generic-hub: http://datos.gbif.es/generic-hub/search#tab_simpleSearch
- collectory: <http://datos.gbif.es/collectory/public/showDataResource/dr190>
- biocache-service (API service):

- <http://datos.gbif.es/biocache-service/occurrences/collections/co11>

The “gbif-websrv” needs a good capacity of storage, because the datasets that you are going to publish in the data portal are stored in this server.

You have to keep in mind that all users that are visiting the data portal, they are using the resources of this Virtual Machine, and if you want to add, update or delete records in the system through the biocache process, you are spending part of these resources.

Then if you need more resources because you have a lot of visits or you have a lot of records, you can add RAM in this Virtual Machine and also you can add other Virtual Machine to host, for example, the collectory module, that only shows us the metainformation of Institutions, Collections and Data resources, that is stored in the “gbif-mysql” server.

In this case the “gbif-mysql” server has 8 GB RAM and 1 TB storage. Below you have 3 links where you can see the kind of information that you have stored in the mysql server.

- Resource: <http://datos.gbif.es/collectory/public/showDataResource/dr190>
- Collection: <http://datos.gbif.es/collectory/public/show/co188>
- Institution: <http://datos.gbif.es/collectory/public/show/in61>

“gbif-websrv” does calls to “replica-websrv” Virtual Machine, because in this server you can have hosted the SOLR service, some fields of the records that are show in the data portal are indexed with this system, in this way, when the user search this indexed information is found quicker.

This service is used in each query that user does, then you need a considerable amount of RAM, not only to run the queries, also to index the number of records that you have. e.g:

http://datos.gbif.es/generic-hub/occurrences/searchq=data_resource_uid:dr190#tab_mapView

http://datos.gbif.es/generic-hub/occurrences/search?q=data_resource_uid%3A%22dr190&fq=country%3A%22Chile%22&fq=taxon_name:%22Arcyodes+incarnata%22

To understand it better with these examples, dataresource_uid, country and taxon_name are facets/fields that are indexed in the solr service. For each of these urls you are using resources of “replica-webbsrv” server.

In other hand, if you want to view the information of one specific record,

<http://datos.gbif.es/generic-hub/occurrences/92fad279-47cc-4bf3-9d3d-f0644ad8fe79>

In this case the information of this fields are hosted in the gbif-cassandra VM, it's a No-SQL server, and it's enough if you have 8GB RAM to manage it. Again this data is closely related with the amount of information that you have to store and with the number of requests.

In each case you have to estimate the amount of memory that you could need according to the number of users and the information that you are going to manage in the data portal.

3.3 Others example of System Architecture

For now, the Atlas of Living Australia platform has been installed and configured in several countries as Argentina, Brazil, France, Portugal, Scotland and Spain. You will find below some examples of system architecture set up by those countries. You will find more information in the [Configuration](#) chapter.

3.3.1 Australia

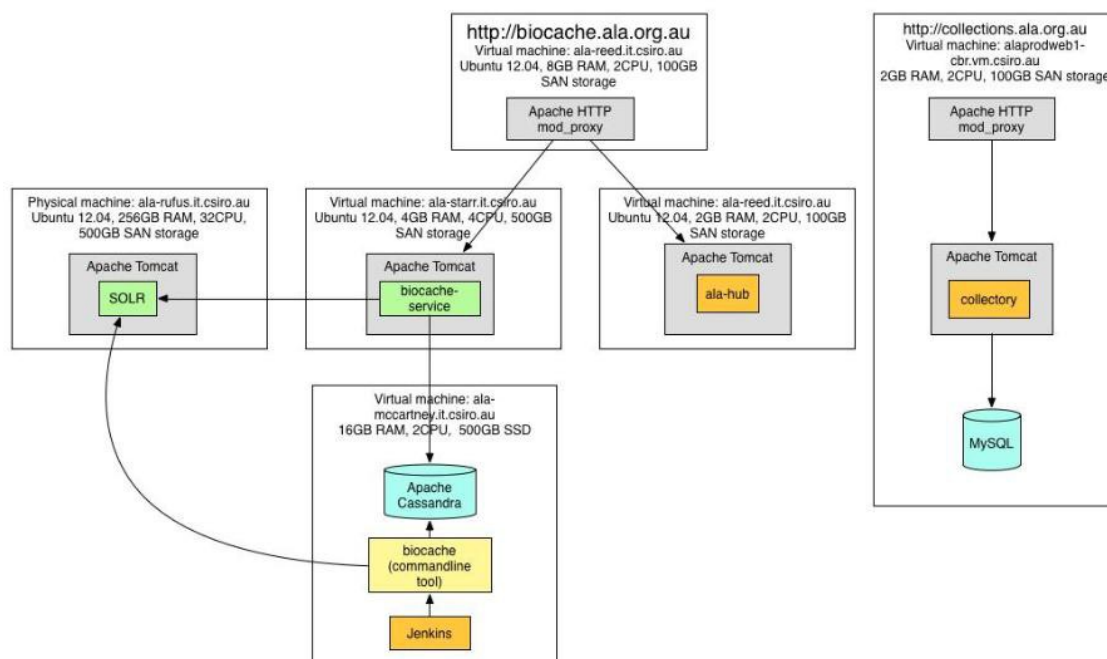


Illustration 14: Australian Infrastructure

3.3.2 France

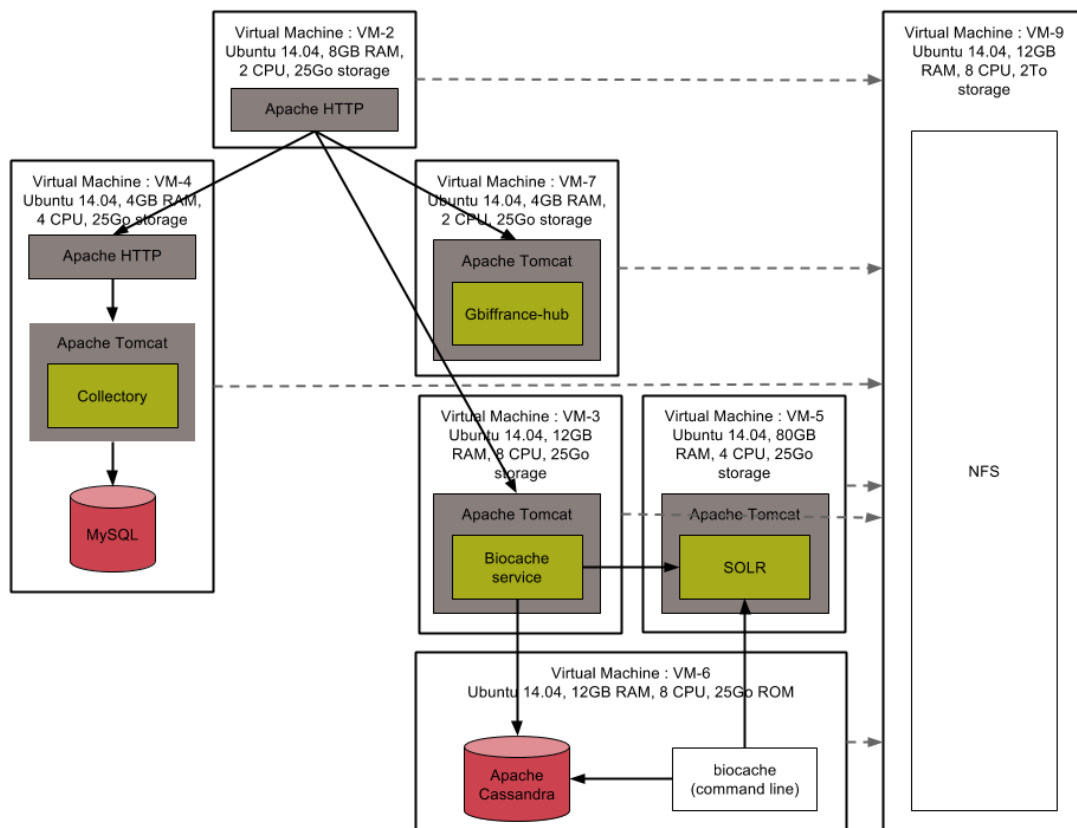


Illustration 15: French Infrastructure

For more information about the system, we invite you to contact directly the IT team of each country. As a turnover in the development team can occur, you can find the newer information about the persons in charge directly on their own website:

- Australia: <http://www.ala.org.au/>
- France: <http://www.gbif.fr>
- Scotland: <http://www.als.scot/>
- Spain: <http://www.gbif.es>

3.4 Installation process (Test environment)

You need to have a Vagrant file and an Ansible playbook in order to guide Vagrant and Ansible for the automation.

The whole installation, on an INTEL i7 2.66Ghz MacBook Pro (late 2010) with SSD, takes about 30 mins to finish. This includes time for transferring war files over the internet.

To get them, clone the GitHub ALA-install repo at:

<https://github.com/AtlasOfLivingAustralia/ala-install>

```
$ git clone https://github.com/AtlasOfLivingAustralia/ala-install.git  
$ cd ala-install
```

3.4.1 Vagrant

```
$ cd vagrant/ubuntu/  
$ vagrant up
```

At this time, you should see an ubuntu instance up and running if you open VirtualBox:

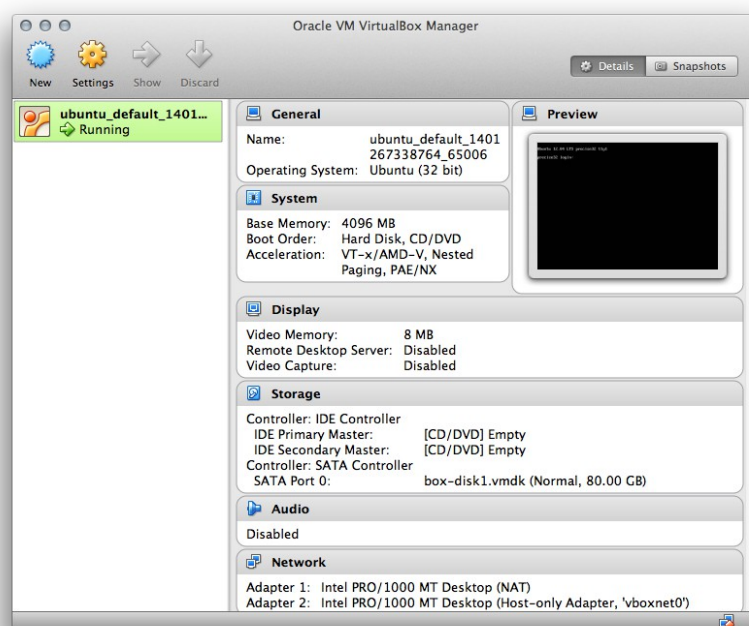
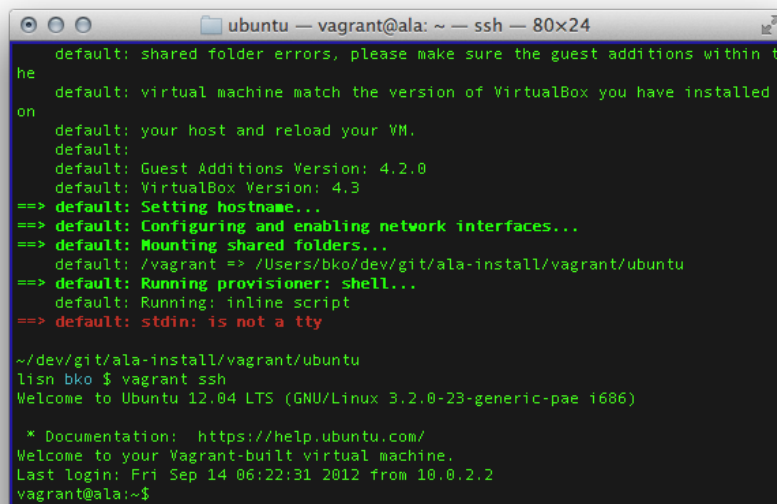


Illustration 16: Ubuntu

As of 28 May 2014, you might see "default: stdin: is not a tty" in red. This doesn't harm because if you do:

```
$ vagrant ssh
```

You can login to the ubuntu instance you've just set up:



```
ubuntu — vagrant@ala: ~ — ssh — 80x24
default: shared folder errors, please make sure the guest additions within t
he
default: virtual machine match the version of VirtualBox you have installed
on
default: your host and reload your VM.
default:
default: Guest Additions Version: 4.2.0
default: VirtualBox Version: 4.3
==> default: Setting hostname...
==> default: Configuring and enabling network interfaces...
==> default: Mounting shared folders...
default: /vagrant => /Users/bko/dev/git/ala-install/vagrant/ubuntu
==> default: Running provisioner: shell...
default: Running: inline script
==> default: stdin: is not a tty

~/dev/git/ala-install/vagrant/ubuntu
lisa bko $ vagrant ssh
Welcome to Ubuntu 12.04 LTS (GNU/Linux 3.2.0-23-generic-pae i686)

 * Documentation:  https://help.ubuntu.com/
Welcome to your Vagrant-built virtual machine.
Last login: Fri Sep 14 06:22:31 2012 from 10.0.2.2
vagrant@ala:~$
```

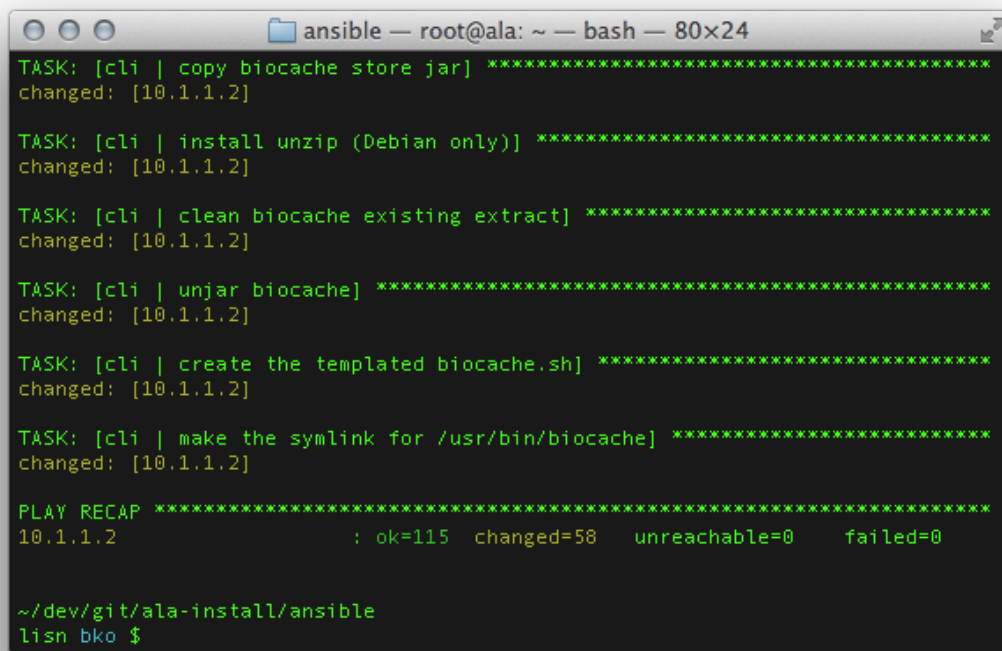
Illustration 17: Ubuntu login

3.4.2 Ansible

Now you're ready to configure the ubuntu server with Ansible. To run the Ansible playbook:

```
$ cd ../../ansible/
$ ansible-playbook -i inventories/vagrant/demo-vagrant ala-demo.yml
--private-key ~/.vagrant.d/insecure_private_key -u vagrant -s
```

The playbook should be finished. There might be some minor non-aborting errors which don't harm. In that case, please file an issue so the developers can investigate. Thanks!



```
ansible — root@ala: ~ — bash — 80x24
TASK: [cli | copy biocache store jar] *****
changed: [10.1.1.2]

TASK: [cli | install unzip (Debian only)] *****
changed: [10.1.1.2]

TASK: [cli | clean biocache existing extract] *****
changed: [10.1.1.2]

TASK: [cli | unjar biocache] *****
changed: [10.1.1.2]

TASK: [cli | create the templated biocache.sh] *****
changed: [10.1.1.2]

TASK: [cli | make the symlink for /usr/bin/biocache] *****
changed: [10.1.1.2]

PLAY RECAP *****
10.1.1.2                : ok=115  changed=58  unreachable=0    failed=0

~/dev/git/ala-install/ansible
lism bko $
```

Illustration 18: Ansible

The ALA demo portal should be accessible now. For convenience, as in the Vagrantfile the hostname is set as ala.vagrant.dev and it has an IP address

```
10.1.1.2, adding a line in /etc/hosts:
10.1.1.2  ala.vagrant.dev
```

...will allow you to visit the ALA demo portal from the hosting machine.

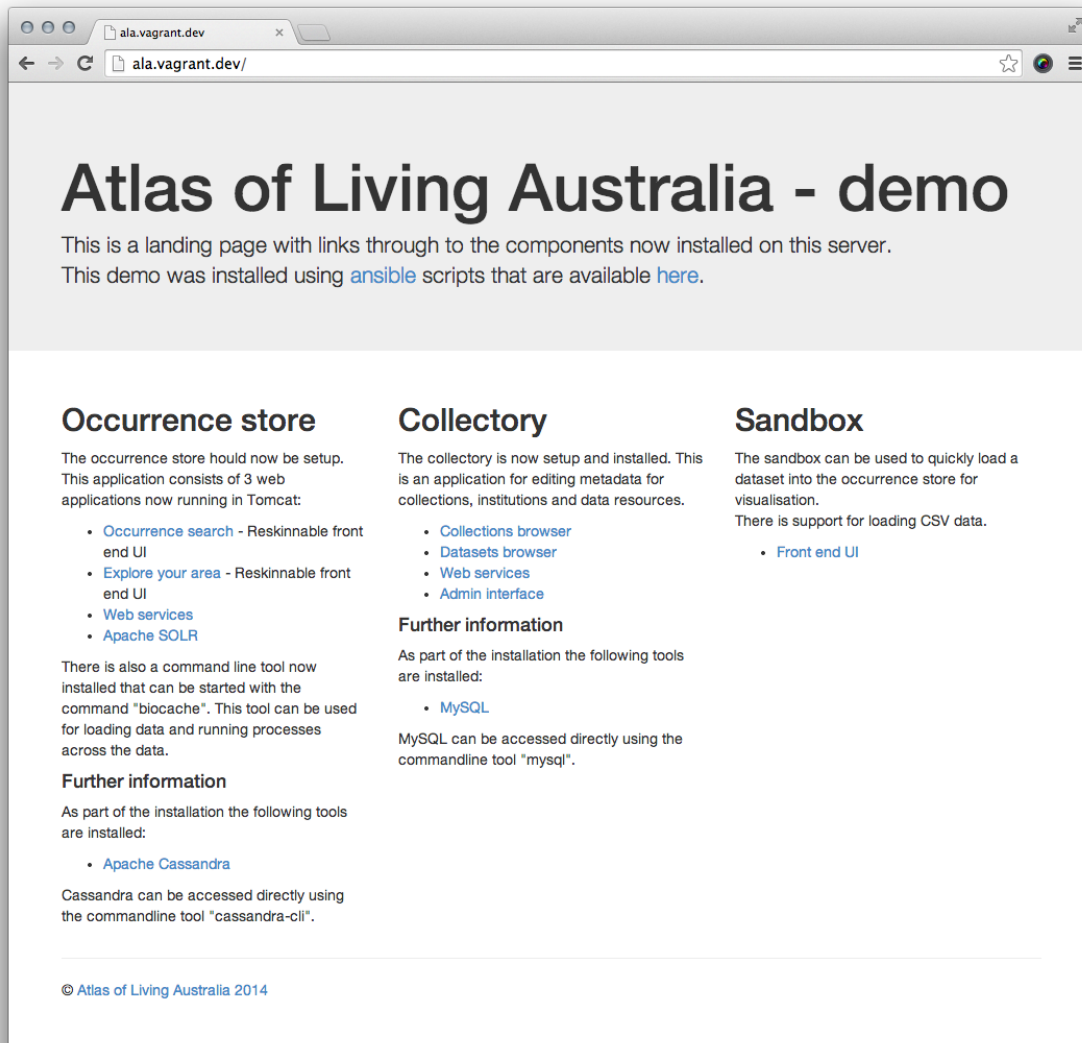


Illustration 19: ALA Demo

Congratulations! Your demo ALA portal is up and running.

3.4.3 Terminating the VM

When you're happy with the test installation, chances are you need a break so you need to stop the virtual machine. To do so:

```
$ cd ../vagrant/ubuntu/  
$ vagrant halt
```

This command shuts down the running machine Vagrant is managing. If you want to remove this test instance, at the same directory, do:

```
$ vagrant destroy
```

This will wipe out the virtual machine and all the configuration you have done with Ansible.

3.5 Installation process on Remote servers (Production environment)

3.5.1 Understanding the concept

This process is very similar that the previous explanation with the difference that in this case, you don't need a virtual machine, you don't need the Vagrant software, you only have to run ansible pointing to a remote server, how it's explaining in this section of the wiki documentation of ala-install:

<https://github.com/AtlasOfLivingAustralia/ala-install#installing-the-ala-demo-on-ec2-or-other-cloud-providers>

Flow of information:

1. In your local machine, you have to install git to clone the remote repository of ala-install
2. You have to create a particular inventory
3. Finally you can run the ansible-playbook on your server.

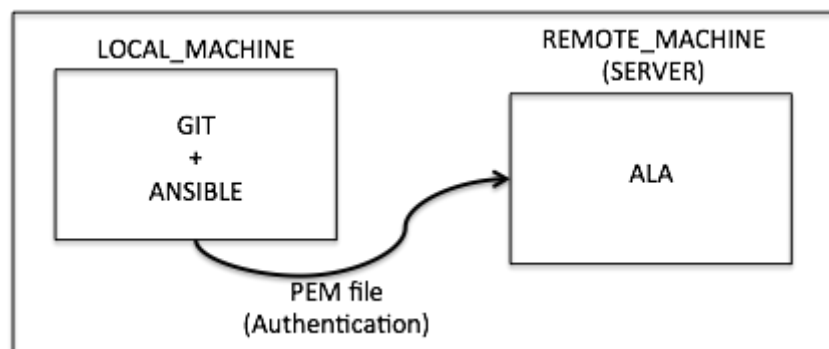


Illustration 20: Flow of information

How can you do it with commands? Very easily, just follow these installation steps.

3.5.2 Installation steps

Once you understand this way to install the software of ALA, you can copy ala-install in your local_machine through git:

```
$ git clone https://github.com/AtlasOfLivingAustralia/ala-install.git
$ cd ala-install
```

Now you have to create a new inventory, your own inventory, to customize the installation in your server. To do that, you can copy demo-ec2 inventory and replace the name.

```
$ cd ansible/inventories
$ cp demo-ec2 ala-server
```

After that, you have to edit the new inventory and replace the IP directions by the IP that your server has. You can do it through Vim editor and with this command:

```
%s/old_direction/new_direction/
$ vim ala-server
```

and in this particular case you have to use the follow command:

```
:/%s/ala-demo.org/myserver.com/
```

At this moment, you can save the file and you can use it to run ansible. Run it in the ansible directory:

```
$cd ..
```

But before that you have to do the next. Copy authorized_key on /root/.ssh/ directory:

```
server$ cp /home/ubuntu/.ssh/authorized_keys /root/.ssh/
```

Finally, you can install your particular playbook pointing to your remote server:

```
$ansible-playbook -i inventories/ala-server ala-demo.yml -private -key
/tmp/xx.pem -u root
```

3.5.3 How to install services on several servers using Ansible playbooks

With this idea you can modify other inventories to install each service that you want in one specific server. For example:

```
$../ansible-playbook -i inventories/cassandra-server cas-standalone.yml
--private-key /tmp/yy.pem -v root
```

cassandra-server inventory should contain the commands needed to install the folow roles:

- common
- java
- tomcat

- mysql
- apache
- cas

The unique thing that you have to know is that if you want to install services in other servers, you have to modify the config file of biocache-hub or collectory to point the following variables (cassandra_hosts, solr_home, list.tool.url, layer.service.url, service.bie.url, datasource.url...) of these service to the new servers.

In this moment you can create a more complex infrastructure, installing the services in different virtual machines and trying to create an optimized Data Portal.

4 Configuration

4.1 Github

All the software modules are stored in the Atlas of Living Australia Github Account. We will present below a table with the module's name and the github directory where you can find the code of the specific element.

You need to follow this step when you begin to install and configure a new item:

- Go to the specific directory in the ALA GitHub account
- Fork the directory in your own account
- Change the name (for example, generic-hub became gbiffrance-hub in France)
- Clone the new directory in your local machine
- Have fun configuring and installing new component.

Module	directory	github's link
Occurrence Research	Generic-hub	https://github.com/AtlasOfLivingAustralia/generic-hub
Collectory	Generic-collectory	https://github.com/AtlasOfLivingAustralia/generic-collectory
Data Hub	Generic-hub	https://github.com/AtlasOfLivingAustralia/generic-hub
Spatial Portal	Spatial-portal	https://github.com/AtlasOfLivingAustralia/spatial-

		portal
Species list	specieslist-webapps	https://github.com/AtlasOfLivingAustralia/specieslist-webapp
Species page	generic-bie	https://github.com/AtlasOfLivingAustralia/generic-bie

Note: This chapter will focus on the collectory and biocache-hub configuration and styling.

4.2 Storage

4.2.1 MySQL

The database's name is "collectory" and you can find below its scheme. Be careful with the encoding, in France, we had issues with link between the local variable of our server and the encoding of the database. You need to configure them as UTF-8 if you don't want to have errors.

To access to the database, you need to connect to your server (as sudo user) hosting the MySQL and then follow this:

```
$ mysql
[] use collectory; //see the database in order to query it
[] show tables; // see all tables of the database
```

You can download the DIAGRAM ER from this link:

https://github.com/AtlasOfLivingAustralia/collectory-plugin/raw/master/Collectory_schema_20140916.pdf

Note: On the configuration file `collectory-config.properties` of the collectory (`/data/collectory/config/`), check if the MySQL connexion is well encoded in UTF-8

```
dataSource.url=jdbc:mysql://localhost:3306/collectory?
autoReconnect=true&connectTimeout=0&useUnicode=true&characterEncoding=UTF-8
```

4.2.2 Cassandra

To access to Cassandra, you need to connect to your server hosting Cassandra and then follow the command lines below:

```
//where occurrence data is stored
$ cd /data/cassandra/data/occ

//connect to cassandra
$ cassandra-cli
//see elements stored
```

```
[default@unknown] use occ;  
// see one elements (below)  
[default@occ] list occ limit 1;  
// It's also possible if you want to print out a specific record  
[default@occ] get occ where uuid = 'e47e0e31-ff9c-4f31-b598-34f452cb023f';
```

In the last line an UUID is generated automatically by Cassandra. As long as a record is contained by the same data resource determined by druid, the UUID will be stable. You will find more information about this on the official website.

4.3 Files architecture

Generic-hub and Generic-collectory are both based on their own plugins (biocache-hub and collectory). In order to keep all of your work when they are a plugin update, they create a system that will use your modification before the plugin modification. You can modify any files (e.g. controllers, services, etc.) but you need to follow the step.

Keep the same tree view as the plugin one.

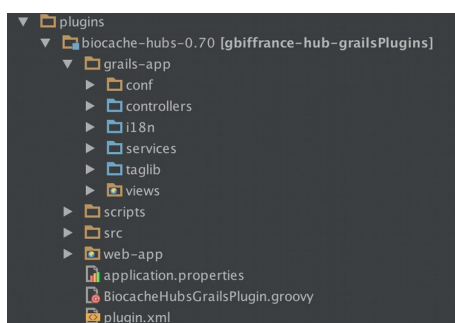


Illustration 21: Plug tree view

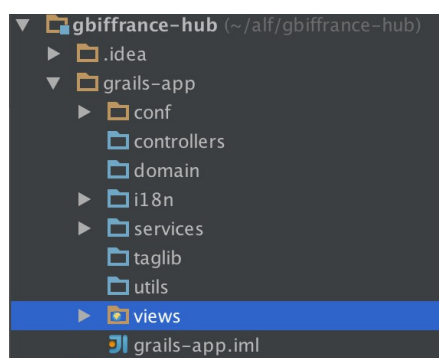


Illustration 22: GBIFrance-hub tree view

If you want to modify the file `list.gsp` located in `grails-app/views/occurrence`, you need to copy this file on the right folder in the gbiffrance-hub's grails-app tree view. You need to follow the same logic for every file that you modify (view, controller, domaine, etc.).

Note: You don't need to copy all of the file but just the ones that you want to modify. If a folder is empty, you can delete this one on your one tree view.

Note: if you find a bug, you can correct it on the plugin and make a pull request in order that your correction help the entire community.

4.4 Configuration file

They are several configuration files that you can modify in order to customize your portal:

Name	Location	Description
<code>basisOfRecord.txt</code>	<code>/data/biocache/vocab</code>	Vocabularies used during the indexation of data resource. If you have some quality errors during the indexation, you may need to look at this file to be sure that there is no missing information e. g. <code>FossilRecord</code> <code>FossilSpecimen</code> When during the indexation, the system find <code>FossilRecord</code> , he need to interpretate this as <code>FossilSpecimen</code>
<code>stateProvince2Countries.txt</code>	<code>/data/biocache/vocab</code>	List of state province with country name
<code>stateProvince</code>	<code>/data/biocache/vocab</code>	List of the state province
<code>biocache-config.properties</code>	<code>/data/biocache/config</code>	Configuration file of the biocache
<code>collectory-config.properties</code>	<code>/data/collectory/config</code>	Configuration file of the collectory
<code>generic-hub-config.properties</code>	<code>/data/yourOrg-hub/config</code>	Configuration file of the yourOrg-hub

4.5 Styling the application

Note: It is the same logic for the collectory and the generic-hub. We will focus our description on the generic-hub

The basic steps are:

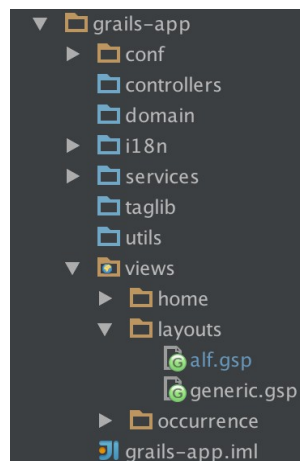
Get the code from GitHub

1. Fork the generic-hub project and rename it to reflect your organisation (e.g. : gbiffrance-hub).
2. Go to generic-hub page and click the "Fork" button in the top right corner of the page.
3. Select your repository and click OK.
4. To rename the project, click the "Settings" icon on the right side and edit the project name (e.g. yourOrg-hub).
5. Than Clone/checkout the project to your local PC.

Create your specific layout

Copy the generic.gsp layout file (`grails-app/views/layouts/generic.gsp`) to a new file (e.g. `yourOrg.gsp`) in the same location

Directly on your IDE (e. g.: IntelliJ)



*Illustration 23:
Create your
specific layout*

or using command line:

```
$ cd yourOrg-hub/grails-app/views/layout  
$ cp generic.gsp yourOrg.gsp
```

Edit Config.groovy to use the new layout file

There are two mechanisms for configuring the web app:

- external config file (properties file) - default location:
/data/appName/conf/appName-config.properties
- internal Grails config file at:
/grails-app/conf/Config.groovy

Note : External config values will take precedence over those in Config.groovy. The ala-demo uses the external config file and this is the recommended way of setting config values.

Change the following vars (in either external or internal file):

```
skin.layout = 'generic'  
skin.orgNameLong = 'Generic Data Portal'
```

to this:

```
skin.layout = 'yourOrg'  
skin.orgNameLong = 'Your Org Name'
```

Example from the internal file:

```
/*  
 * SKINNING  
 */  
skin.layout = 'alf'  
skin.orgNameLong = "Atlas of Living France"  
skin.orgNameShort = "Occurrences"  
// whether crumb trail should include a home link that is external to this webapp  
skin.includeBaseUrl = true  
//skin.headerUrl = "classpath:resources/generic-header.jsp" // can be external  
//skin.footerUrl = "classpath:resources/generic-footer.jsp" // can be external  
skin.fluidLayout = true // true or false  
chartsBgColour = "#FFFFFF"  
// 3rd part WMS layer to show on maps  
map.overlay.url = ""  
map.overlay.name = ""  
biocache.baseUrl="http://recherche.gbif.fr/biocache-service"
```

Illustration 24: Internal file

Note : you may want to point the app at a local version of biocache-service by adding/editing the line biocache.baseUrl

```
biocache.baseUrl = "http://yourOrg.org/biocache-service"
```

Note : if using external properties file, then omit the quote characters around string values.

Edit the layout yourOrg.gsp file

The layout file is a GSP file, which is similar to the Java JSP file format, with some minor differences (see Grails docs). Grails uses the SiteMesh library to provide common HTML component to pages.

Note : The generic-hub (~ biocache-hubs plugin) uses the Bootstrap CSS framework, so there are some HTML elements that are required to be present for all pages to render properly.

Create custom CSS (and optional JS) files

Grails provides a mechanism to manage static resources (CSS, JS and images), called the Resources plugin, and we recommend using this feature.

Resources are declared in the file :

```
yourOrg-hub/grails-app/conf/ApplicationResources.groovy
```

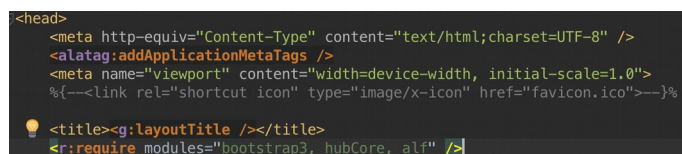
with related files being managed as modules. Modules can dependOn other modules, which provides a way of making sure dependent files appear before the calling file (e.g. a jQuery plugin will dependOn jQuery and thus jQuery will be printed to the page before the plugin).

The layout file contains a require tag that states which modules are required for all pages that use that particular layout file. E.g.

```
<r:require modules="bootstrap2, hubCore" />
```

To add a new module, simply add a reference to the module:

```
<r:require modules="bootstrap2, hubCore, youOrg" />
```



```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<al:tag:addApplicationMetaTags />
<meta name="viewport" content="width=device-width, initial-scale=1.0">
%{--<link rel="shortcut icon" type="image/x-icon" href="favicon.ico">--}%
<title><g:layoutTitle /></title>
<r:require modules="bootstrap3, hubCore, alf" />
```

Illustration 25: Add a new module

and then add that module to ApplicationResources.groovy:

```
yourOrg {
    dependsOn 'bootstrap2', 'hubCore' //
    resource url: [dir:'css', file:'yourOrg.css']
    resource url: [dir:'js', file:'yourOrg.js']
    resource url: [dir:'images', file:'twitter.png']
    resource url: [dir:'css/images', file:'arrow_stat_grey_right.pgn']
}
```

As an example, see how the institution page as been changed:

American Museum of Natural History

Acronym: AMNH

Collections




- 1. American Museum of Natural History Palaeontology Collections

Usage statistics

This month			117 records downloaded from 8 downloads.		
Biosecurity management/planning	0 events	0 records			
Citizen science	0 events	0 records			
Collection management	0 events	0 records			
Conservation management/planning	0 events	0 records			
Ecological research	3 events	27 records			
Education	1 events	13 records			
Environmental assessment	0 events	0 records			
Other	1 events	13 records			
Other scientific research	0 events	0 records			
Restoration/remediation	0 events	0 records			
Scientific research	0 events	0 records			
Systematic research/taxonomy	1 events	38 records			
Testing	1 events	3 records			
*The testing statistics are not included in the total count of downloads.					
Unclassified	2 events	26 records			

Data access

187 records

 View records
 Download usage stats
 Alert me about new records
 Alert me about annotations

Location

Central Park West at 79th Street
New York
NY 10024-5192
US

212-769-5100

Web site

[Visit the institution's website](#)

Illustration 26: Institution page (Australia)

Afficher les occurrences
1,004 enregistrements

VUE D'ENSEMBLE

STATISTIQUES

DESCRIPTION

Aix-Marseille Université a été créée par le décret n° 2011-1010 du 24 août 2011. Elle est entrée en vigueur le 1er janvier 2012, se substituant à l'université de Provence, de la Méditerranée et Paul-Cézanne. C'est aujourd'hui une des plus jeunes universités de France, c'est aussi la plus grande de par le nombre de ses étudiants, de ses personnels et par son budget. C'est également la plus grande université francophone. Autant d'atouts, au-delà des résultats d'ores et déjà notables dans le domaine de l'enseignement et de la recherche, qui font d'Aix-Marseille Université un établissement d'enseignement supérieur et de recherche d'excellence.



COLLECTIONS

1. Aix-Marseille Université - MARS herbarium – Cytogenetic data-base
Les variations chromosomiques concernent 90% des Ptéridophytes et 50 à 60% des Angiospermes. Il s'agit de processus fondamentaux de différenciation et de spéciation...

SITUATION GÉOGRAPHIQUE

Aix-Marseille Université Jardin du Pharo - 58, bd
Charles Livon
Marseille Cedex 07
13284
France

+33 (0)4 91 39 65 00

CONTACT

Bruno Vila
Téléphone : +33 (0)4 13 55 12 29
Contacter par mail

SITE WEB

Visitez le site internet de l'institution

Visiter la page de l'institution sur le GBIF.org

Illustration 27: Institution page (France)

Another example shows the configuration of the results page:

Occurrence records

Canis lupus Advanced search Quick search

21,857 results for SPECIES: Canis lupus

Customise filters

Narrow your results

- Taxon
- Identification
- Location
- Occurrence
- Record
- Assertions
- Attribution

Records Map Charts Record images

Downloads Alerts

per page: 20 sort: Date added order: Descending

Subspecies: **Canis lupus dingo** | Dingo Date: 1991-06-07 State: Queensland
Institution: Commonwealth Scientific And Industrial Research Organisation Collection: Australian National Wildlife Collection Basis Of Record: Machine
Observation Catalog Number: Sounds:X08430 [View record](#)

Subspecies: **Canis lupus dingo** | Dingo Date: 1991-05-30 State: Queensland
Institution: Commonwealth Scientific And Industrial Research Organisation Collection: Australian National Wildlife Collection Basis Of Record: Machine
Observation Catalog Number: Sounds:X15336 [View record](#)

Subspecies: **Canis lupus dingo** | Dingo Date: 1991-06-07 State: Queensland
Institution: Commonwealth Scientific And Industrial Research Organisation Collection: Australian National Wildlife Collection Basis Of Record: Machine
Observation Catalog Number: Sounds:X15422 [View record](#)

Subspecies: **Canis lupus dingo** | Dingo Date: 1991-06-07 State: Queensland
Institution: Commonwealth Scientific And Industrial Research Organisation Collection: Australian National Wildlife Collection Basis Of Record: Machine
Observation Catalog Number: Sounds:X15425 [View record](#)

Subspecies: **Canis lupus dingo** | Dingo Date: 1991-06-07 State: Queensland
Institution: Commonwealth Scientific And Industrial Research Organisation Collection: Australian National Wildlife Collection Basis Of Record: Machine
Observation Catalog Number: Sounds:X15437 [View record](#)

Illustration 28: Results page (Australia)

ENREGISTREMENTS D'OCCURRENCES

Canis lupus

Recherche avancée

Recherche rapide

Personnaliser les filtres

41,155 résultats pour text:Canis lupus

AFFINER VOS RÉSULTATS

TAXONOMIE

IDENTIFICATION

LOCALISATION

OCCURRENCE

ENREGISTREMENT

ATTRIBUTION

ENREGISTREMENTS

CARTE

GRAPHIQUES

Télécharger les données

par page : 20

trier : Ajouté le

ordre: Décroissant

Species: **Canis lupus** | **Wolf** Country: France

Data Resource: Données ONF Faune-Flore-Fonge Basis Of Record: Human Observation Catalog Number: Données ONF Faune-Flore-Fonge:248981 View record

Species: **Canis lupus** | **Wolf** Country: France

Data Resource: Données ONF Faune-Flore-Fonge Basis Of Record: Human Observation Catalog Number: Données ONF Faune-Flore-Fonge:249215 View record

Species: **Canis lupus** | **Wolf** Country: France

Data Resource: Données ONF Faune-Flore-Fonge Basis Of Record: Human Observation Catalog Number: Données ONF Faune-Flore-Fonge:260208 View record

Species: **Canis lupus** | **Wolf** Country: France

Data Resource: Données ONF Faune-Flore-Fonge Basis Of Record: Human Observation Catalog Number: Données ONF Faune-Flore-Fonge:256333 View record

Species: **Canis lupus** | **Wolf** Country: France

Data Resource: Données ONF Faune-Flore-Fonge Basis Of Record: Human Observation Catalog Number: Données ONF Faune-Flore-Fonge:248696 View record

Illustration 29: Results page (France)

4.6 Internationalization

We invite you to read the [Internationalization](#) chapter for more information about this. There are some modules ready to be translated as Generic-collectory and Generic-hub but it is possible that others modules are not ready because the code need to be *internationalizable*.

5 Loading data

Among the manipulated data, some are added by the administrator, other are objects created by the GBIF.org as those contained in the IPTs.

Upload data on ALA portal is one of the administrators rights, data are from different types:

- Metadata: collections, institutions, data providers, data hubs, contact, data resources
- Configuration data: providerCode, providerMaps
- Occurrence: Resource (including resources GBIF)

You will find below a schema showing how the data type works together.

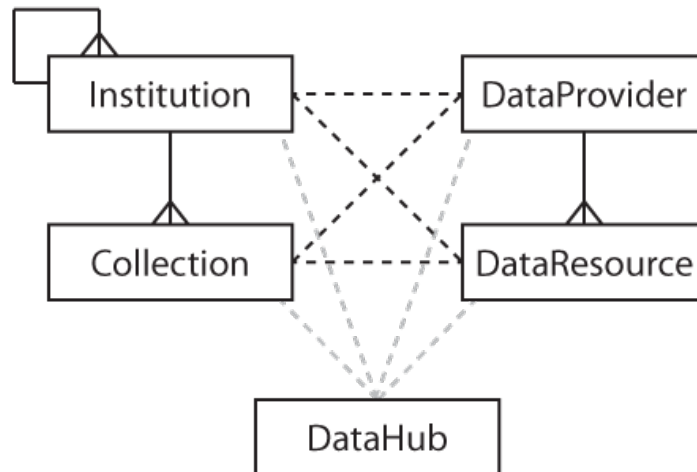


Illustration 30: Entity relationship diagram

1. One Institution can have many Collections.
2. One Collection can have just one Institution
3. One Data Provider can have many Data Resources, many Collections or many Institutions.
4. One Data Resource can have one or more Institutions and one Data Provider but can be link to many Collections and many Institutions.
5. One Data Hub can have many Collections, Institutions and Data Resources.

5.1 Type of data

The addition of objects in the portal is carried out via the administration pages so, the first step is to login in order to see it. The figure below is is the customization made by GBIF France but you will be able to see the same menu.

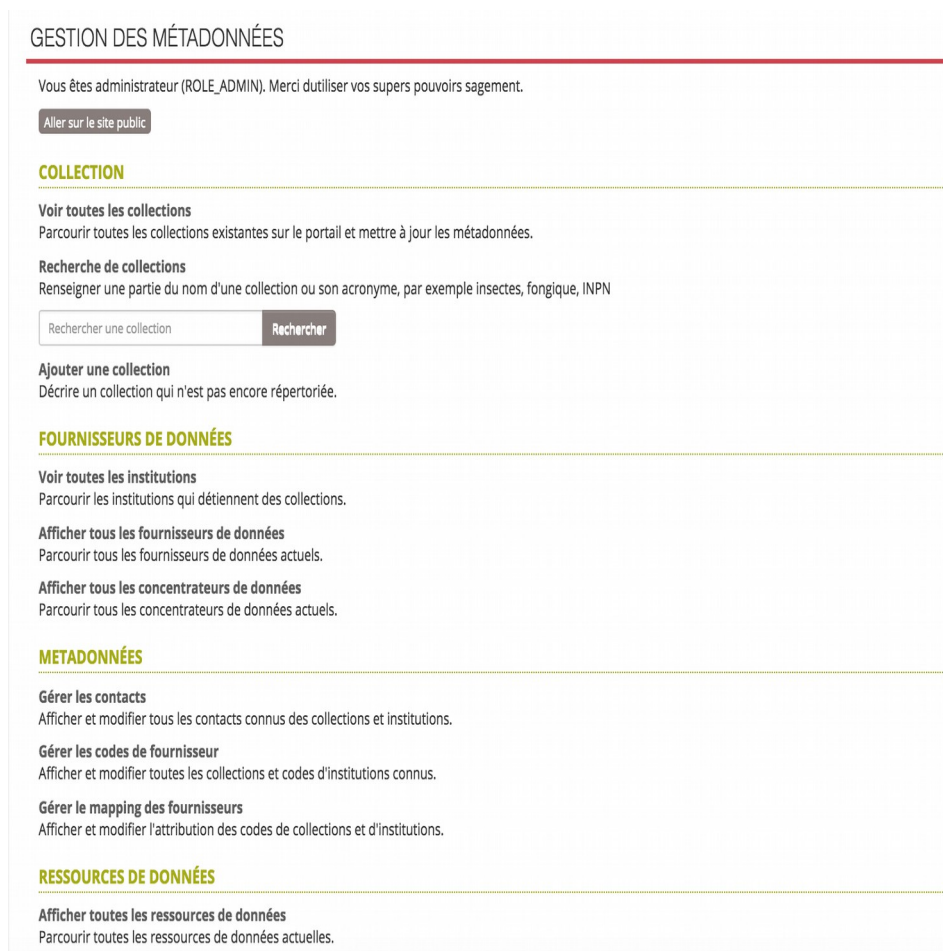


Illustration 31: GBIF France

You can add a collection directly from this page using the add collection link. For the other objects, you will need to go by the “See all [name_Object]” page. A link “Add [name_Object]” will be display and will lead you to the add form of the object.

5.1.1 Collection

To create a new collection, either you go with the “Add a collection” on the administration page or on the visualisation page of Collections; or you can create a collection when you create a new data provider.

5.1.2 Data provider

A data provider can be an institution or a data Hub; an institution is a Museum, a particular, a private or public collection; a data hub is a structure that have multiple data.

5.1.3 Institution

To create a new institution, information will be requested among which: Site collection, description, type, collections, logo, location, related data providers, contacts, etc.

5.1.4 Data Hub

To create a data hub, you need to add basic informations as name, url, etc. but you also need to specify collections, institutions and data resources link to this Hub. For more configuration and how it works, we invite you to read the [chapter about the architecture](#).

5.1.5 Contact

You can create contact directly on the menu dedicated to them but you can also create contact when you create an institution, a collection, a data provider or a data hub.

CRÉER CONTACT

Accueil

Afficher toutes les Contacts

Prénom

Prénom

Nom de famille

Téléphone

Portable

E-mail

Fax

Commentaires

Rendre public ☒

Créer

Illustration 32: Contact form

You can also add role to a contact, it is an open variable so you can add what you want and then you can say if you want him to manage the resource, if you want him to be inform by any annotation made on his occurrence and if this contact is the primary contact of the resource. If yes, you will see the name on the public page.

Rôle	<input type="text" value="Contact de la ressource"/>	
Administrateur	<input type="checkbox"/>	Détermine si le contact peut modifier les propriétés collection.
Informeur	<input type="checkbox"/>	Si activé, le contact sera notifié lorsque des annotations seront apportées aux enregistrements collection.
Contact principal	<input type="checkbox"/>	Le contact principal est celui qui est affiché sur la page pour ce collection.

Illustration 33: Resource Contact

5.2 Configuration data

5.2.1 Provider Code

You fill the Provider Code with Institution Code and Collection Code that you can find in the occurrences of your data resource.

5.2.2 Provider Map

In order to link your data resource to your collection and/or institution, you need to create provider map before running the indexation.

Id	7
Collection	Aix-Marseille Université - MARS herbarium – Cytogenetic data
Codes de l'institution	<ul style="list-style-type: none"> Aix-Marseille Université
Codes de la collection	<ul style="list-style-type: none"> MARS Herbarium Cytogenetic Database
Exact	Oui
Avertissement	
Correspond à n'importe quel Code de Collection	Non
Dernière mise à jour	2015-05-22 18:09:51.0
Date de création	2015-05-22 18:09:51.0

[Editer](#) [Supprimer](#)

Illustration 34: Provider map 1

Basically, you need to tell institution name, collection name (not mandatory), institution code and collection code and then you can create your provider map.

Illustration 35: Provider map 2

5.3 Data resource

This is two ways of adding data resource on the portal : a file with data in it or a dataset located online in the other GBIF nodal point.

5.3.1 External data resource

Data resource' occurrences is uploading as file using Darwin Core Archive; this file should contain a single resource.

In addition, this resource is assigned a UID (Unique Identification Number) to identify it among the other resources of the portal and a page displays the file information.

In order to this page to be used by the portal, we must ingest these data.

The admin will connect to the ubuntu instance on the virtual Machine, then open the biocache console to finally 'ingest' the resource with the line. To do this, you need to be in sudo mode.

```
$ sudo su
$ biocache
biocache > ingest -dr UID_of_resource
```

If the data resource is too big, you can divide the ingest tool into three different command:

```
biocache > load UID_of_resource
biocache > process -dr UID_of_resource
```

```
biocache > index -dr UID_of_resource
```

5.3.2 GBIF resources

The resources of the nodal point of the GBIF are also resources that can be assimilated by the portal.

The steps are different of the previous one : the admin only need to fill a form (using login of your gbif.org account), choose a country and the number of resource needed.

With the fill submit, the resources are download quickly after that.

There are institutions, organizations or individuals that possess data and who provide an access to them, they are the data providers.

5.3.3 Using IPT

There is a third option to get the IPT hosted data resources into your portal : you can directly harvest the data provider's IPT.

For this, you must first go through a necessary step: the creation of a new DataProvider.

In the administration area, go to "View all data providers" then on "Add a new DataProvider".

The name and URL of the IPT will be requested in the "enter name" section; then click on "Check endpoint" integration in the IPT section.

To complete the creation of a new DataProvider, upgrade the data resources ("Update data resources").

Once done, the IPT and ALA portal are connected. The next step is to 'ingest' the data.

To do this, open a terminal (available in a Unix OS) and type the following command line:

```
$ sudo biocache
```

to be a director on biocache Machine.

```
Biocache > ingest -a
```

to ask the biocache machine take all the data.

6 Backing-up and updating

6.1 Back-up databases

6.1.1 Backing up MySQL database

The backup and the recovery of mysql database are quite easy. You can use the mysqldump as show below:

```
$ mysqldump --lock-all-tables collectory > dump_mysql.sql --default-character-set=UTF-8

// collectory = name of the database
// dump_mysql.sql = output file
```

To restore a database using MySQL is easier. You just have to follow this step

```
$ mysql < dump_mysql.sql
```

Note: don't forget to specify with database you want to use!

6.1.2 Backing up Cassandra data store

Backing up Cassandra is essentially making a snapshot of occurrence store. Cassandra comes with some command-line tools that we use for this task: cassandra-cli and nodetool.

You will find more information about Cassandra on the chapter 2 about the architecture.

6.1.3 Making a snapshot

By default this directory is owned by root so you will need to sudo.

```
// Assuming it's the first time we make a snapshot,
// this directory should be empty before we do
$ cd /data/cassandra/data/occ/occ && ls

// Make a snapshot of occ, which we store occurrence data
$ nodetool snapshot occ
```

A folder 1406163740504 is created under /data/cassandra/data/occ/occ/snapshots. If you list the files under the 1406163740504 directory, you'll notice it has the same files in /data/cassandra/data/occ/occ. 1406163740504 is where you backup occurrence store and /data/cassandra/data/occ/occ is where backed up occurrence data would be restored.

6.1.4 Restoring a snapshot

To use this snapshot on the new machine follow these instructions:

1. Shut down the cassandra:

```
$ service cassandra stop
```

2. Clear all files in the commitlog directory:

```
$ rm /data/cassandra/commitlog/*
```

3. Move the snapshot data

```
$ mv /data/cassandra/data/occ/attr/occ-migration-20160113/*  
/data/cassandra/data/occ/attr/  
$ mv /data/cassandra/data/occ/dellog/occ-migration-20160113/*  
/data/cassandra/data/occ/dellog/  
$ mv /data/cassandra/data/occ/distribution_outliers/occ-migration-  
20160113/* /data/cassandra/data/occ/distribution_outliers/  
$ mv /data/cassandra/data/occ/duplicates/occ-migration-20160113/*  
/data/cassandra/data/occ/duplicates/  
$ mv /data/cassandra/data/occ/loc/occ-migration-20160113/*  
/data/cassandra/data/occ/loc/  
$ mv /data/cassandra/data/occ/occ/occ-migration-20160113/*  
/data/cassandra/data/occ/occ/  
$ mv /data/cassandra/data/occ/occ_duplicates/occ-migration-20160113/*  
/data/cassandra/data/occ/occ_duplicates/  
$ mv /data/cassandra/data/occ/occ_outliers/occ-migration-20160113/*  
/data/cassandra/data/occ/occ_outliers/  
$ mv /data/cassandra/data/occ/outliers/occ-migration-20160113/*  
/data/cassandra/data/occ/outliers/  
$ mv /data/cassandra/data/occ/qa/occ-migration-20160113/*  
/data/cassandra/data/occ/qa/  
$ mv /data/cassandra/data/occ/qid/occ-migration-20160113/*  
/data/cassandra/data/occ/qid/  
$ mv /data/cassandra/data/occ/queryassert/occ-migration-20160113/*  
/data/cassandra/data/occ/queryassert/  
$ mv /data/cassandra/data/occ/taxon/occ-migration-20160113/*  
/data/cassandra/data/occ/taxon/  
$ mv /data/cassandra/data/occ/upload/occ-migration-20160113/*  
/data/cassandra/data/occ/upload/
```

4. Restart cassandra

```
$ service cassandra restart
```

5. Run nodetool repair.

```
$ sudo nodetool repair occ attr dellog distribution_outliers duplicates  
loc occ occ_duplicates occ_outliers outliers qa qid queryassert taxon  
upload
```

6.1.5 Backing up SOLR data store

Backing up SOLR is essentially making a copy of data folder. You will find more information about SOLR on the chapter 2 about the architecture.

```
// SOLR index emplacement
$ ls /data/solr/biocache/data
index tlog
```

Looking inside the `data` directory you see `index` and `tlog` directories. `data` is the unit you want to back up.

```
// Making a copy of Solr index
$ cd /data/solr/biocache
$ sudo mkdir solr-index-backup
$ sudo chown tomcat7:tomcat7 solr-index-backup
```

At the point, for the `index` and `tlog` inside `solr-index-backup`, you can copy them from `/data/solr/biocache/data` from localhost or a remote host. Once those contents are in place, make sure they have owner and group set as `tomcat7`, which is the default user/group on Ubuntu that runs Tomcat.

Now, you are going to create a Solr core that uses this backup and can be swapped later. To do so, navigate your browser to the SOLR admin at

`~/solr/#/~cores/biocache` and click 'Add core' and enter values as the image shows:

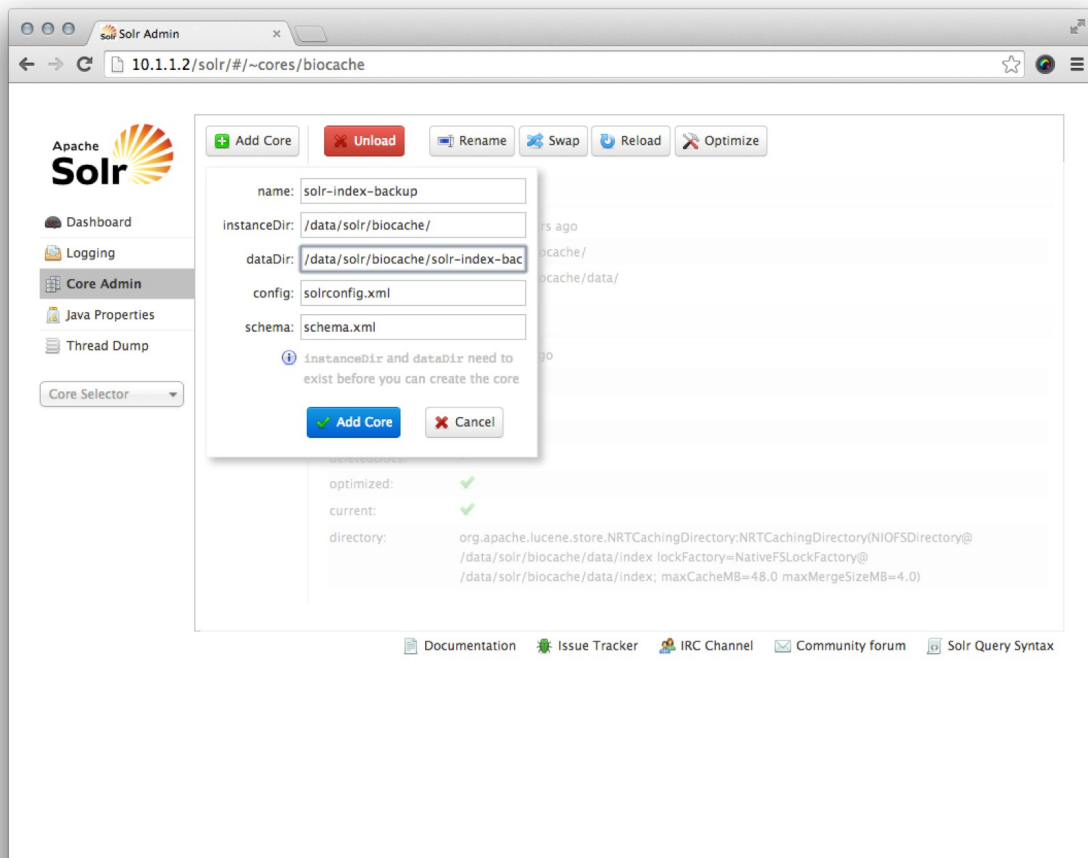


Illustration 36: Solr

Once the new core is successfully created, click the new core and see if all details of 'Core' and 'Index' section are all the same except file directories:

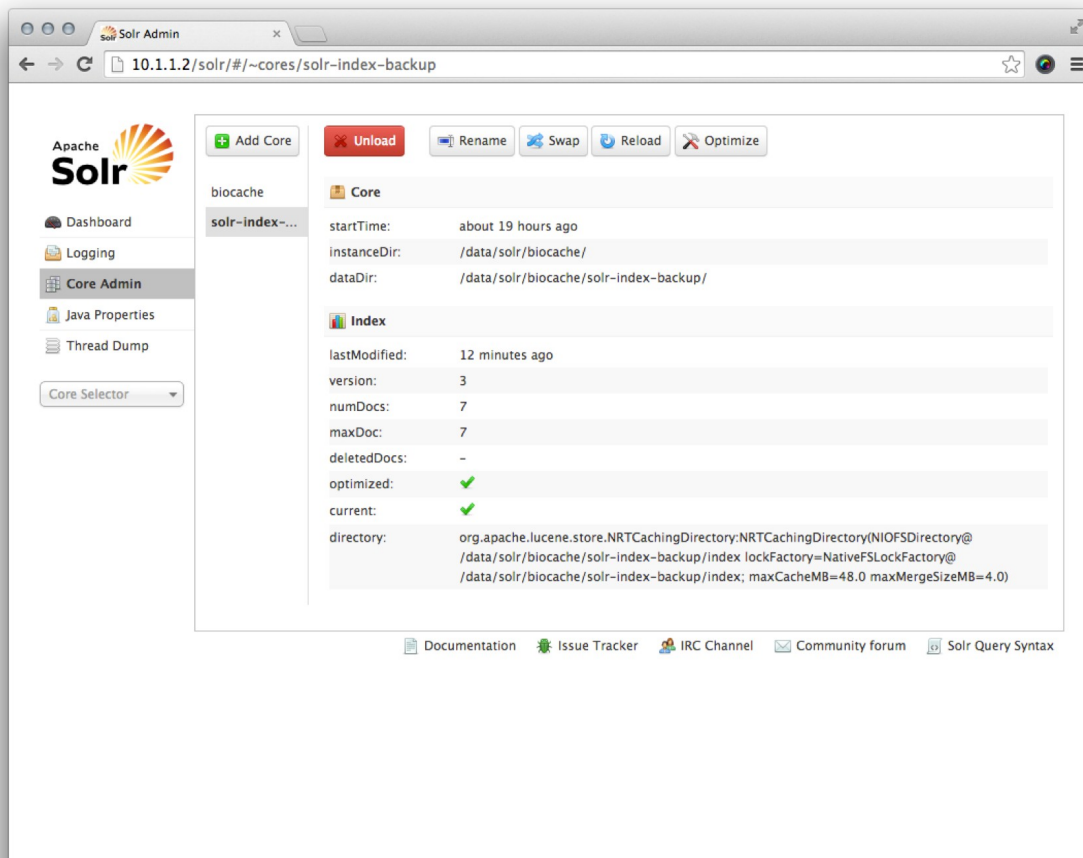


Illustration 37: Solr index backup

When ready, you can use the 'Swap' feature to use the backup.

6.2 Updating tools

Thanks to the Ansible scripts, if you want to update one or several of your components, you just have to re-run your configuration script. The main asset of the Ansible is the possibility to run the script as much as we want. The updating part of your architecture will not be difficult.

For example, if you want to update the configuration files of the backend of the biocache, you just have to run this following command on your bash:

```
ansible-playbook -i inventories/vagrant/gbiffance-biocache-backend
biocache-backend.yml -u ubuntu -s --tags biocache-properties
```

You can find several features link to ansible-playbook as:

```
--skip-tags
the list of tags will be skip during the running
```

```
--tags  
only this list of tags will be executed
```

If you want to upgrade tools as Tomcat or Java, we invite you to see on the Atlas of Living Australia Github or directly on the mailing list if the modules are compatible with the new version of those software, but don't forget to make your modification on a fork of the project and changing the name of it (see [configuration and styling](#) chapter). If it is the case, you can use the Ansible scripts because you can run as much as you want.

For more information on the updating, we invite you to read the [installation](#) chapter. You can use the tags features explained before.

7 Functionalities

The ALA portal key functionalities can be grouped as follows:

- [Data visualization](#)
- [Import features](#)
- [Export features](#)
- [Metadata management](#)
- [Errors detection and reporting](#)

Note: Please refer to the [Functional Glossary](#) for a list of ALA entities

7.1 Data Visualization

Visualization concerns data and the various objects present in the portal such as: collections, institutions, data providers, data resources, reports, audit events. See [Entity Relationship Diagram](#)

The entity of DataProvider represents the global organization that could be composed by several Institutions. e.g: In Spain the CSIC is composed by several centers or institutions, each Institution can publish several Collections and these collections contain the metainformation related with the global dataset that is shared through one or several DataResource.

Finally we have a feature in the data portal that allows us to create a DataHub: you can create your own portal with a subset of records coming from several DataResources. These ones could be related to different Institutions of different DataProviders.

To manage and use this information, you have two roles in the data portal: Administrators and Visitors, that obviously, do not have the same possibilities of permission and visualization:

- Administrator can create, modify and delete the information related with the Data Providers, Institutions, Collections, Data Resources and Contacts.
- Visitors can view data from:
 - The **collectory portal**: Users can see metadata from collections and institutions (name, contacts, location, etc.). From those dedicated pages, he will be able to access to occurrence data filtered by this institution or this collection.
 - The **occurrences search engine**: he can make a specific search and see the result list of occurrence data. Through the occurrences page, he will be able to go to the institution, data provider and collection pages.
 - The **datasets search engine**: he will be able to see metadata from dataset (name, contacts, location, etc.). From those pages, he will be able to access to the data of this specific dataset.



Illustration 38: Welcome page

7.1.1 Searching occurrences

The searching of occurrences can be done in two different ways: basic or advanced.

The basic search is to get into the search bar the scientific name or vernacular name of this occurrence search.

Search for records in

Simple search
Advanced search
Shapefile search

Note: the simple search attempts to match a known **species/taxon** - by its scientific name or common name. If there are no name matches, a **full text** search will be performed on your query

Illustration 39: Basic search

Search for records in

[Simple search](#) [Advanced search](#) [Shapefile search](#)

Find records that have
ALL of these words (full text)

Find records that specify the following scientific name (verbatim/unprocessed name)
Raw Scientific Name

Find records from the following species group
Species Group

Find records from the following institution or collection
Institution or Collection

Find records from the following regions
Country
State/Territory

Find records from the following type status
Type Status

Find records from the following basis of record (record type)
Basis of record

Illustration 40: Advanced search

Advanced Search allows to add criteria and filters to narrow the search.

We can see the result of a research in 3 different views:

the default view is a map centered on the area containing the results. Points represent the occurrences matching the search criteria.

Occurrence records

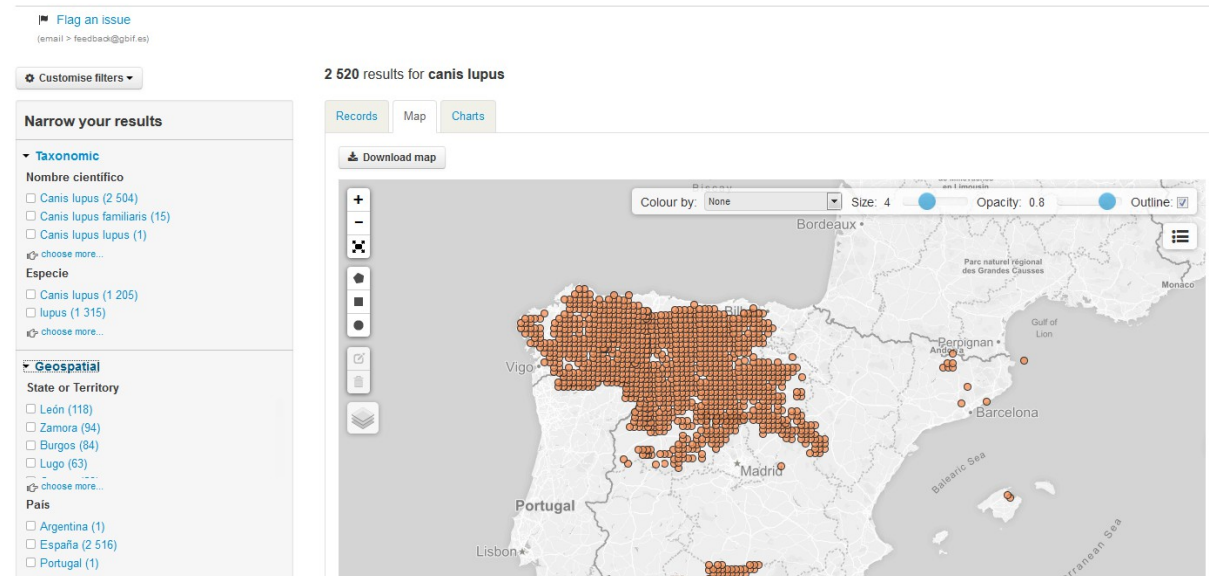


Illustration 41: Result page of a search (Map)

The second view is a list of the resources found in the search. Each element includes its name, record type, catalog number and a link to the original record. The administrator can configure this list of parameters to include more parameters, eg: scientific name, taxon rank, date, country, name of dataset...

The screenshot shows the 'Occurrence records' interface for a search of *Canis lupus*, displaying a list view of the results. The sidebar on the left is identical to the map view. The main area shows a list of records with columns for 'Records', 'Map', and 'Charts'. Below the list, there are controls for 'Downloads', 'per page' (20), 'sort' (Date added), and 'order' (Descending). The list contains several entries for *Canis familiaris* and *Canis lupus*, each with a date, country, and a link to view the record. The records are sorted by date added in descending order.

Records	Map	Charts
<p>Canis familiaris Linnaeus, 1758 Date: 1977-06-25 Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Preserved Specimen Catalogue Number: Mammals-Housed At MHNG, Geneva.MHNG-MAM-1492.040 View record</p>		
<p>Canis familiaris Linnaeus, 1758 Date: 2013-10-11 Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Human Observation Catalogue Number: Naturgucker-606093987 View record</p>		
<p>Canis familiaris Linnaeus, 1758 Date: 2009-05-16 Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Human Observation Catalogue Number: Naturgucker-1721980560 View record</p>		
<p>Canis lupus familiaris Linnaeus, 1758 Date: 2013-10-15 Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Human Observation Catalogue Number: User: 4537:15209 View record</p>		
<p>Canis lupus subsp. lupus Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Occurrence Catalogue Number: Mammals:174751.7260937 View record</p>		
<p>Especie: <i>Canis lupus</i> State: Sonia Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Fossil Specimen Catalogue Number: 73761:684223 View record</p>		
<p>Especie: <i>Canis lupus</i> Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Fossil Specimen Catalogue Number: 35339:419657 View record</p>		
<p>Especie: <i>Canis lupus</i> Country: España Data Resource: Registros Biológicos Españoles De Bases De Datos Extranjeras Basis Of Record: Fossil Specimen Catalogue Number: 34926:369203 View record</p>		
<p>Especie: <i>Canis lupus</i> Country: España</p>		

Illustration 42: Search result page (Record list view)

The last view displays diagrams illustrating analysis of the search result data using different criteria: species, time, etc.

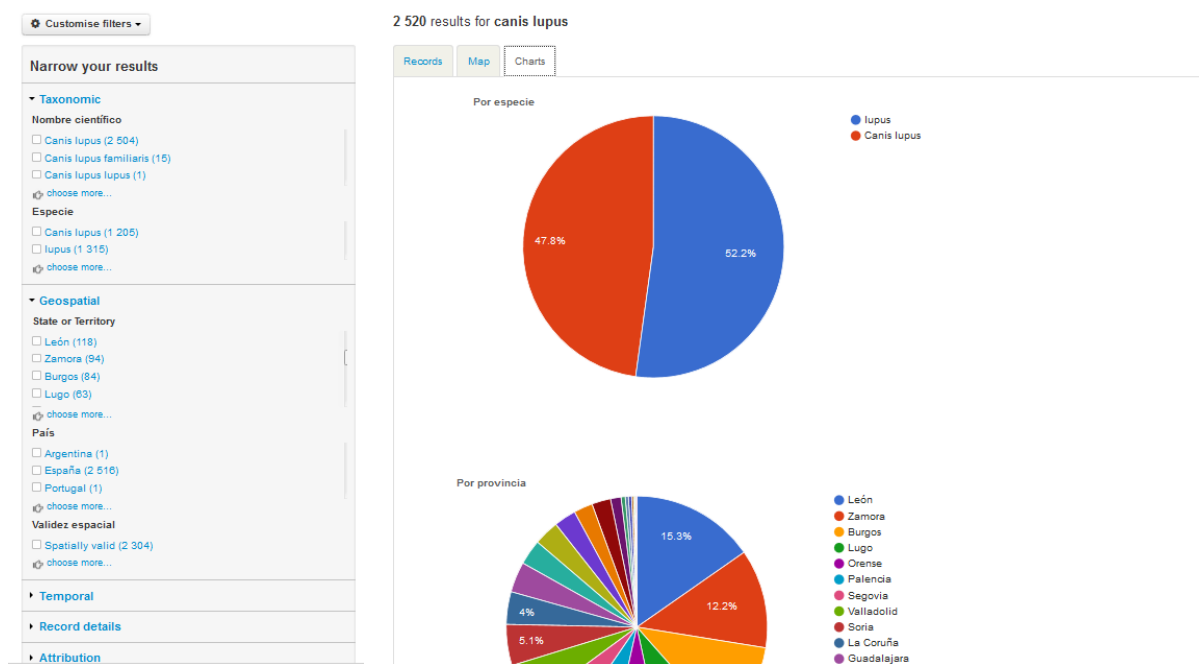


Illustration 43: Result page of a search (Charts)

Regardless of the view chosen, filters (taxonomy, space, time, etc.) are available on the left of the page to allow user to refine his search.

7.1.2 Searching metadata

Collections

Under this category, the visitor can view data on collections, institutions and projects that are connected.

Institution, collections and projects

Learn about the institution, the collections they hold and view records of specimens that have been databased. Currently only the collections of partners are shown. Over time this list will expand to include all natural history collections in

Click a button to only show those organisms.

All collections

Plants

Fauna

Insects

Microorganisms

162 colecciones en total.

159 colecciones visibles.



Illustration 44: Collections Page (Maps)

A world map is displayed with the location of connected collections. Note that a list view is also available.

This visualization can be refined by filtering using the type of organism: plants, animals, fungi (mushrooms), micro-organisms (viruses, bacteria, etc.) and Paleontology (fossil).

Plants

Fauna

Insects

Microorganisms

61 fauna colecciones.

59 colecciones visibles.

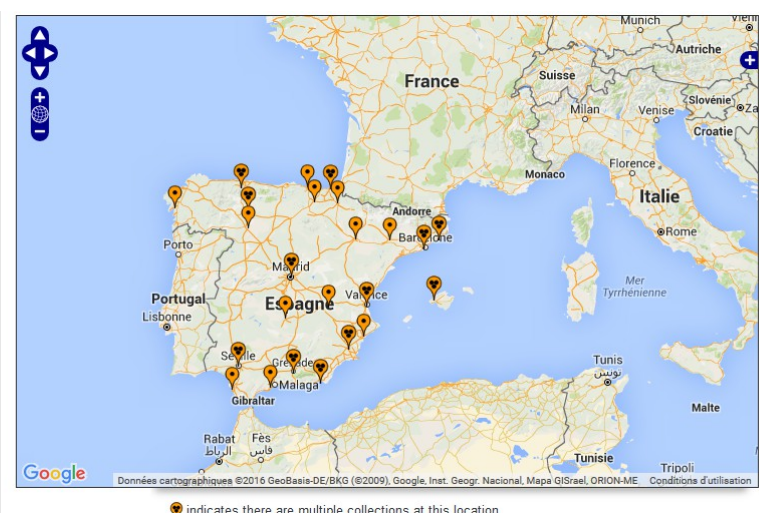


Illustration 45: Filter fauna on the collections page

These filters are exclusive to each other (not cumulative).

Datasets

Unlike the two previous options, search by datasets does not use any map but rather textboxes and dropdown lists to set the criteria for the search.

data sets

Much of the content in the Portal de Datos, such as occurrence records, environmental data, images and the conservation status of species, comes from data sets provided by collecting institutions, individual collectors and community groups. These data sets are listed on this page.

Refine the list of data sets by clicking on a filter in the left hand list. Click the [▶](#) toggle to see a description of the data set. To find out more, go to the data set's full metadata page by clicking on its name.

The screenshot displays the 'Datasets' search interface. On the left, a sidebar titled 'Refine Results' contains two sections: 'Tipo de licencia' (License type) with options 'CC BY-NC (119)', 'CC BY (68)', and 'Custom license (2)'; and 'Tipo de contenido' (Content type) with options 'Species list (87)', 'Point occurrence data (67)', 'Scientific names (36)', 'Gbif import (23)', and 'Taxonomy (16)'. Below these is an 'Institución' (Institution) section listing 'MNCN (12)', 'MCNB (10)', 'CSIC-Real Jardín Botánico (10)', 'MAGRAMA (9)', and 'BDDBC (6)'. The main area shows 'Mostrando 189 recursos.' (Showing 189 resources.) with search and reset buttons. Below this are dropdowns for 'Results per page' (set to 20), 'Sort by' (set to nombre), and 'Sort order' (set to ascendente). A 'Download' button is in the top right. The dataset list includes: 1. 'Antarctic Porifera database from the Spanish benthic expeditions: Bentart, Gebrap and Ciemar' (Type: records, License: CC BY-NC, Version: Ver registros). 2. 'Aranzadi Ringing Scheme (bird ring-recovery data)' (Type: records, License: CC BY, Version: Ver registros). 3. 'Aranzadi Zientzi Elkartea' (Type: records, License: CC BY, Version: Ver registros). 4. 'Atlas of Iberian water beetles (ESACIB database)' (Type: records, License: CC BY, Version: Ver registros).

Illustration 46: Datasets page

Results are shown as a list of different datasets corresponding with: the type of contained resources, the license and a link to see the records it contains.

Each dataset displayed includes a link that leads to a page listing the detailed information available about it: description, content type, list of recordings, etc.

Antarctic Porifera database from the Spanish benthic expeditions: Bentart, Gebrap and Ciemar

Centro Oceanográfico de Gijón, Instituto Español de Oceanografía

Acrónimo: esponjas_antartida LSID

Description


This database includes information on porifera collected during several Spanish Antarctic expeditions: Bentart 94; Bentart 95; Gebrap 96; Ciemar 99/00 and Bentart 2003. The expeditions focused from the Antarctic Peninsula to Bellingshausen Sea. Porifera were collected using rock dredge, Van Veen dredge, anchor dredge, suprabenthic sledge, fish traps, scuba diving, Box corer and Agassiz trawl. The dataset now reaches 771 occurrence data that have checked for systematic reliability. This database is not finished and the collection is growing. The data are available in Gbif.

Citation

Please cite this data as follows, and pay attention to the rights documented in the rights.txt: Please respect the rights declared for each dataset in the download: Oceanographic Center of Gijón, Spanish Institute of Oceanography (IEO): Antarctic Porifera database from the Spanish benthic expeditions: Bentart, Gebrap and Ciemar

Rights

Dataset: Antarctic Porifera database from the Spanish benthic expeditions: Bentart, Gebrap and Ciemar Rights as supplied: Not supplied

Creative Commons Attribution-NonCommercial Spain (CC BY-NC) 


Digitised records

766 registros can be accessed through the Atlas of Living Spain. La fecha en la cual este recurso fue consultado para su actualización fue el 23 Nov 2014. La última actualización se hizo el 23 Nov 2014.



Acceso a datos

766 registros

 Ver registros

Location

Avda Principe de Asturias 70bis
Gijón
33212
España

pilar.rios@gi.ieo.es
+34 985326277

Pilar Ríos López
Contacto administrativo

Web site

[Visit the data resource's website](#)

Illustration 47: Dataset details page

An analysis of the records and metadata associated to the dataset are also displayed through charts using different criteria (temporal, location, etc.) for better visibility of the dataset content.

It is also possible to access directly the records included in the dataset.

Georeferenced data

Occurrence records



Illustration 48: Georeferenced data

It is similar to the occurrence search, except that every occurrence record is displayed on a map.

Administration

Administrators have other ways to view and edit metadata from the administration page.

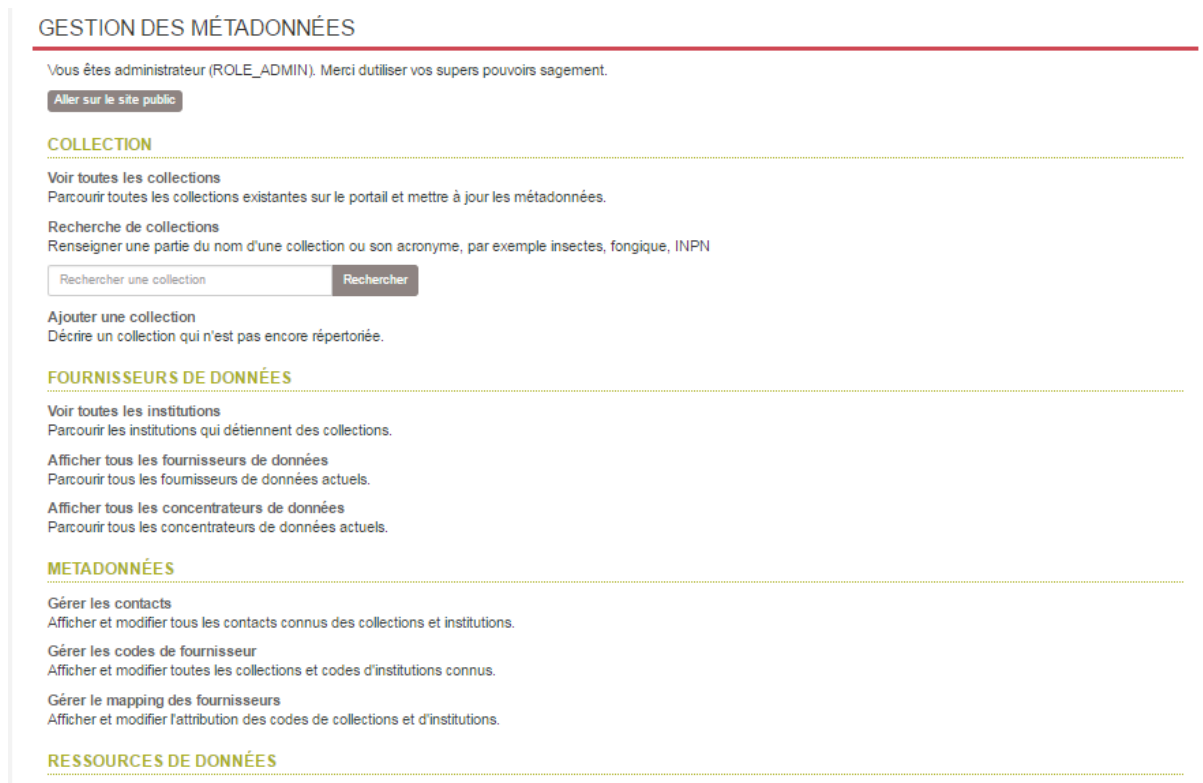


Illustration 49: Administration main page

The data available are: collections, data providers (institutions), data hubs and data resources. The metadata are: contacts, provider codes and provider maps.

Collections

The collections are presented in an ordered list including their name, acronym (not mandatory), the type of institution and collection...

The name of the collection and the name of the institution include links to their specific pages.

Accueil

Afficher toutes les collections

Ajouter un nouveau collection

Aller sur la page publique

Voir le résumé (JSON)

Voir les données (JSON)

Supprimer

Institution : Université d'Aix-Marseille

GUID : 4521e9af-e6c4-4f0c-9701-f2cd0c279b32

UID : co95

Site de la collection :

Dernière modification : not available on 2015-12-01 14:35:36.0

Editer

DESCRIPTION

Description publique :

Les variations chromosomiques concernent 90% des Ptéridophytes et 50 à 60% des Angiospermes. Il s'agit de processus fondamentaux de différenciation et de spéciation. Sur le plan taxonomique ces variations chromosomiques ont largement contribué à améliorer les classifications tandis que sur le plan évolutif, la compréhension de ces variations a permis de mieux cerner l'origine et l'histoire de certains groupes. Ainsi la base de données MARS-CYTO de l'Université d'Aix-Marseille a pour objectif de mettre à la disposition des chercheurs, via le portail du Gbif, les nombres chromosomiques et les informations relatives aux témoins herbiers, support de ces cytotypes. Cette base concerne essentiellement la flore du Bassin Méditerranéen : flore endémique insulaire, flore de Provence (SE. France), espèces rares et menacées et espèces invasives en France. Elle comporte également quelques monographies.

Illustration 50: Information page of a collection

A sequence of the letters of the alphabet on the top of the page acts as an index and facilitates access to the first collection whose name starts with the selected letter.

Institutions

AFFICHER TOUTES LES INSTITUTIONS

Accueil	Nom	Acronyme	Type
Ajouter un nouveau Institution	Agoralogie		
	Association des Naturalistes de la Vallée du Loing et du massif de Fontainebleau		
	Association Peau-Bleue		aquarium
	BRC Forage and turf, INRA Lusignan		
	CBGP (UMR INRA, Cirad, IRD, Montpellier SupAgro)		
	Centre de Ressources Biologiques de l'Institut Pasteur		
	Centre International de Ressources Microbiennes-Levures		

Illustration 51: Institutions page

Institutions are a particular kind of data providers. The visualisation is an array containing the name, acronym and type of each institutions. Name contains, each time, the link to the dedicated page.

Data providers

AFFICHER TOUTES LES FOURNISSEUR DE DONNÉES

Accueil	Nom	UID	N ° ressources
Ajouter un nouveau fournisseur de données	Association Peau-Bleue	dp4	1
	Institut National de Recherche Agronomique	dp2	1
	IRD - Institute of Research for Development	dp3	0
	Service du Patrimoine Naturel - Inventaire National du Patrimoine Naturel	dp0	5
	Tela Botanica	dp1	0

Illustration 52: Data Providers page

The visualization of data providers shows an array containing the name, the UID and the type of each institutions. Name contains, each time, the link to the dedicated page.

Data hubs

Same as the "Institutions" part; the notable difference is that data hubs have a UID instead of an acronym.

Data resources

AFFICHER TOUTES LES RESSOURCE DE DONNÉES

Accueil	Nom	UID	Type	Fournisseur de données
Ajouter un nouveau ressource de données	Actualisation de la cartographie des Ptéridophytes de France et d'Europe occidentale	dr154	records	
	Aix-Marseille Université - MARS herbarium – Cytogenetic data-base	dr61	records	
	Apterygota collection (EA) of the Muséum national d'Histoire naturelle (MNHN - Paris)	dr131	records	
	Araneae collection (AR) of the Muséum national d'Histoire naturelle (MNHN - Paris)	dr132	records	
	arim	dr14	records	
	Atlas des plantes vasculaires de Lorraine	dr33	records	
	Aves MZS	dr12	records	
	Baillon Collection	dr19	records	
	Barley French National Collection	dr22	records	

Illustration 53: Data resources page

Data resources are displayed in an array with different columns : name, UID, data resource type and the associated data provider. the name of a data resource led to the dedicated page.

7.2 Import features

Importing data are treated in the [Loading data](#) chapter.

7.3 Export features

A massive export of all data is possible in JSON or CSV format.

7.4 Metadata

7.4.1 Contacts

The contacts are arranged in an array according to six fields: ID, email, title, first name, name and phone.

AFFICHER TOUTES LES CONTACTS

Accueil	ID	E-mail	Titre	Prénom	Nom de famille	Téléphone
Ajouter un Contact	263	mberthel@snv.jussieu.fr		Martine Berthelin		
	264	regine.vignes_lebbe@upmc.fr		Regine Vignes-Lebbe		
	312	gbifprovider@mnhn.fr		Simon Chagnoux		
	313	gbifprovider@mnhn.fr				
	361	marie.meister@zoo-ulp.u-strasbg.fr		Marie Meister		
	374	philippe.lebaron@obs-banyuls.fr		Philippe LEBARON		
	375	curator.mola@obs-banyuls.fr		Laurent INTERTAGLIA		

Illustration 54: Contacts page

Their IDs are connected to their individual profile which has as data, in addition to the identity and coordinates, the list of collections where it is marked as contacts.

CONTACT -

Accueil	ID	158
Afficher toutes les Contacts	Titre	
Ajouter un Contact	Prénom	CRBIP technical contact
	Nom de famille	
	Téléphone	
	Portable	
	E-mail	crbip@pasteur.fr
	Fax	
	Commentaires	
	Rendre public	Oui
	Contact pour	<ul style="list-style-type: none">Centre de Ressources Biologiques de l'Institut PasteurCentre de Ressources Biologiques de l'Institut Pasteur

Editer

Supprimer

Illustration 55: Contact detail page

7.4.2 Providers codes

AFFICHER TOUTES LES PROVIDERCODES

Accueil	Code
Ajouter un ProviderCode	38
	agoralogie
	AIX
	Aix-Marseille Université
	Ananas
	ANDUZE
	Apterygota collection (EA) of the Muséum national d'Histoire naturelle (MNHN - Paris)
	AR
	Araneae collection (AR) of the Muséum national d'Histoire naturelle (MNHN - Paris)
	arim
	Atlantique Nord Est__ovipa
	AUT
	AUT

Illustration 56: Provider codes page

Viewing Provider codes is a list of codes of the different providers.

7.4.3 Providers maps

The maps include information of collection and the provider who provides it.

AFFICHER TOUTES LES PROVIDERMAPS

Accueil						
Ajouter un ProviderMap						
	Id	Collection	Exact	Égal à tous les codes collection	Code institution	Code collection
	7	Aix-Marseille Université - MARS herbarium – Cytogenetic data	Oui	Non	Aix-Marseille Université	MARS Herbarium Cytogenetic Database
	9	Apterygota collection (EA) of the Muséum national d'Histoire	Oui	Non	EA	MNHN
	10	Araneae collection (AR) of the Muséum national d'Histoire na	Oui	Non	MNHN	AR
	11	arim	Oui	Non	Institute of Research for Development	arim

Illustration 57: Provider maps page

An array display all their information: the id and the name of the collection, the institution's code and the code of the collection.

Two Additional information are also given to whether the collection is equal to the collection code in the mapping code.

7.5 Error detection and reporting

One of the best features of this data portal is that it helps to detect errors in occurrence records. In this way we can improve the data quality of our information stored in the data portal. When users detect an error, they can send us a feedback reporting that specific error, and we can put him in contact with the provider in order to try to solve it.

Below you can see the [process to report an error](#) in the Spanish data portal:

1. The first thing that you have to do it's find an error in the data portal. The most useful option to find errors it's the map view, but before that, to facilitate the task you can filter the records by one particular dataset.
2. In this example we have filtered the records of dr190
"NEOTROPICMYXO. A database of Myxomycetes from the Neotropics". This is the URL to filter the information:
http://datos.gbif.es/generic-hub/occurrences/search?q=data_resource_uid:dr190&lang=en#tab_mapView

3. You can see the result in the map view:

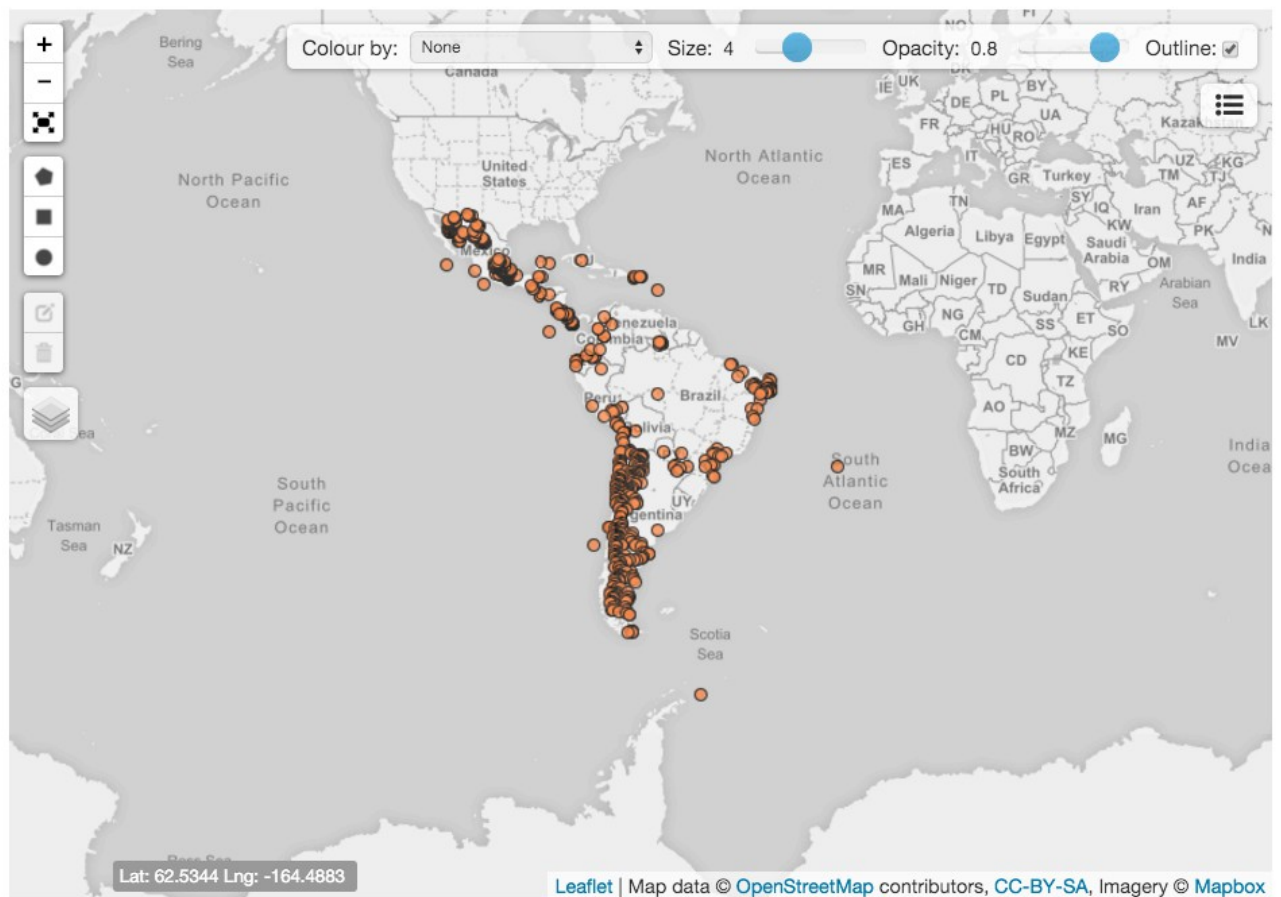


Illustration 58: Map view

4. The records should be located in the Neotropics regions, and in the image we can see at least two records that are located out of this location. Both records are identified for their related URL:
<http://datos.gbif.es/generic-hub/occurrences/934e4116-0de9-4f4c-bd3f-033785e940ab>

<http://datos.gbif.es/generic-hub/occurrences/9c60080f-69c0-442d-9478-959244954353>

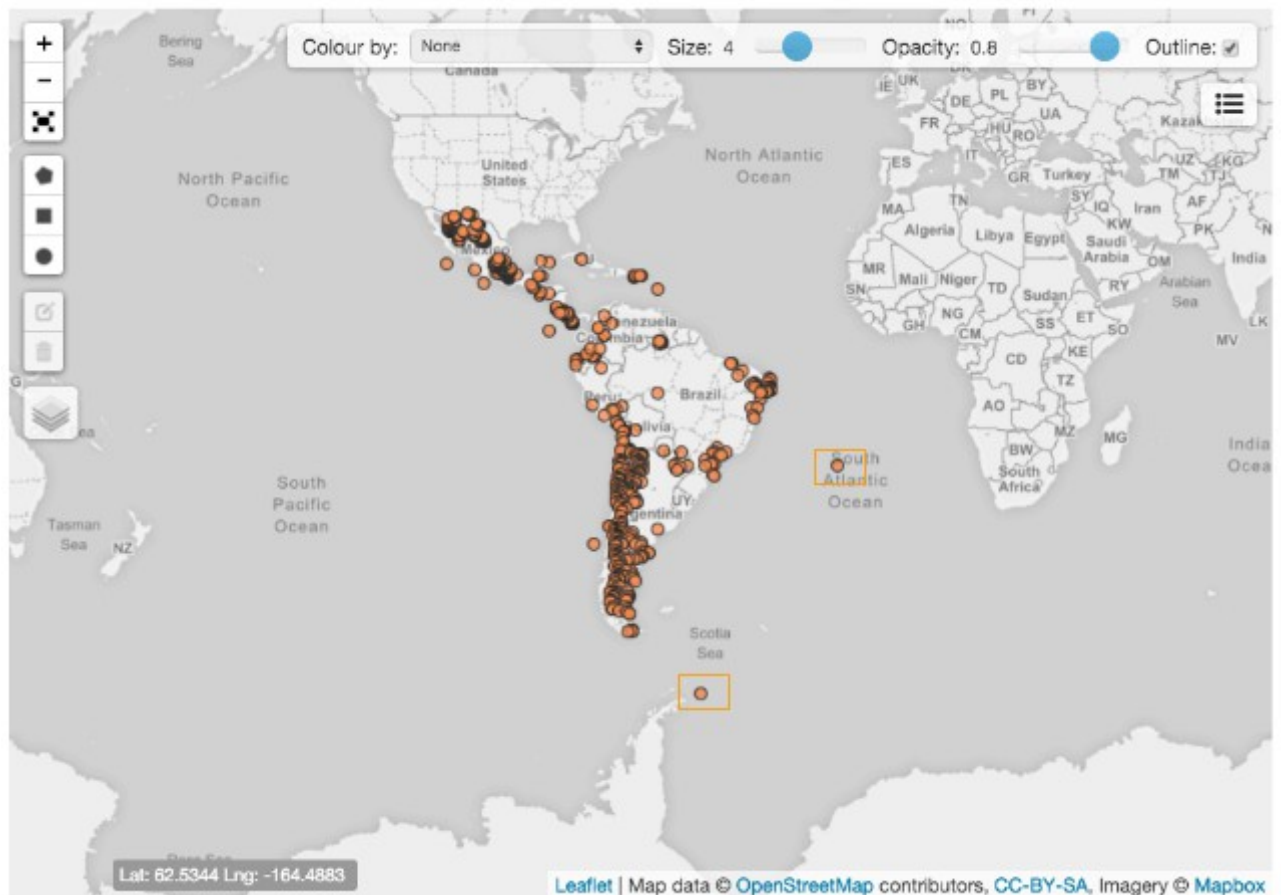


Illustration 59: Detecting errors in the Map view

5. Now you can send a mail to support team of gbif, using the tool of "feedback". It's very easy, the unique thing that you have to do it's copy the url where you detected the error and then you have to click on the button "Flag an issue", that you can find in the view of records or in the view of one specific record:

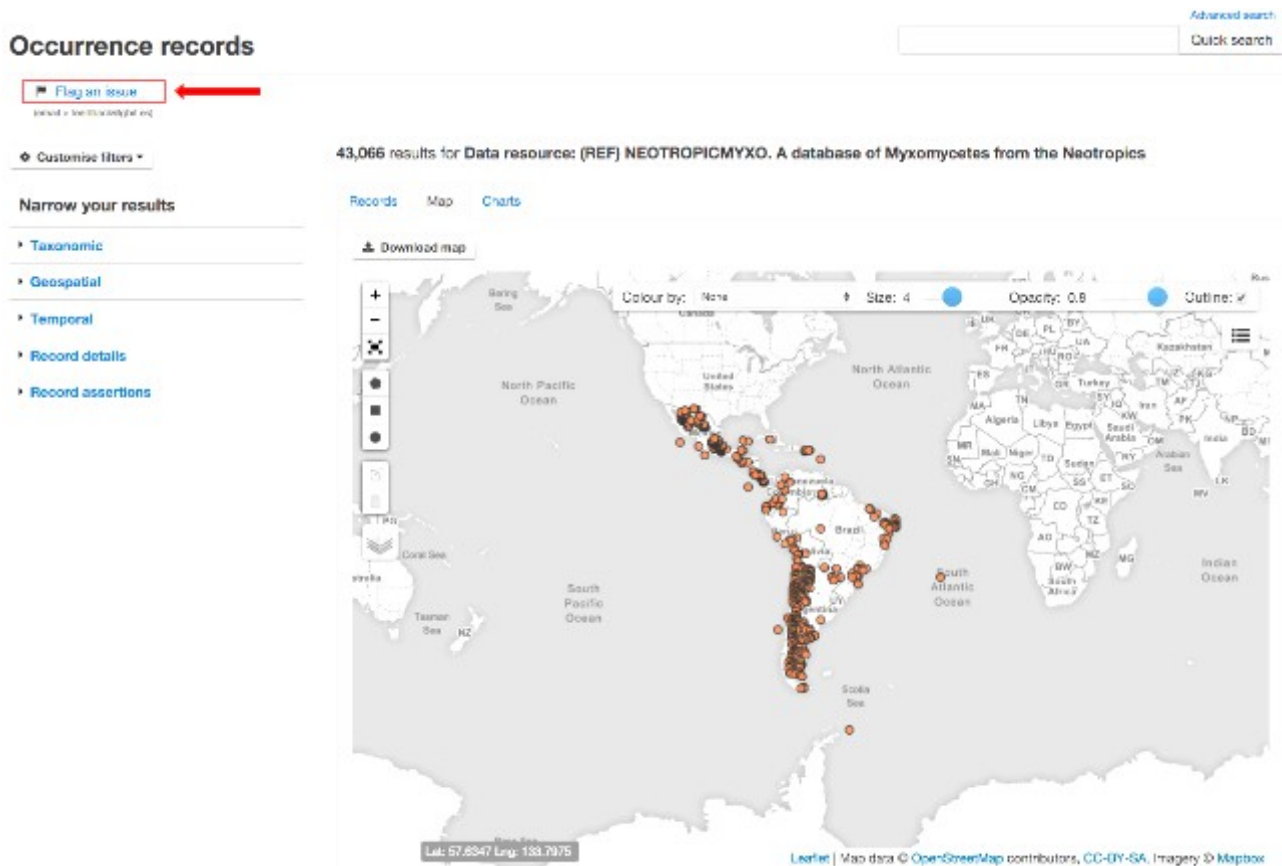


Illustration 60: Flag issue in the Map view



Illustration 61: Flag issue in the record view

- When you do click on this button, your mail client is opened and then you can report the errors that you have detected.
- In this moment the support team of GBIF studies this case and puts in contact with the owner of these records to try to solve these errors in case of this is necessary.

8 Further readings

For more in-deep information about the ALA framework, development technologies and frameworks, data standards, check the following online resources:

8.1 About Atlas of Living Australia software

Atlas of Living Australia documentation wiki

<https://github.com/AtlasOfLivingAustralia/documentation/wiki>

ALA Install (README file)

<https://github.com/AtlasOfLivingAustralia/ala-install>

Presentations about ALA portal in international congresses or ALA technical workshops

<https://github.com/AtlasOfLivingAustralia/documentation/wiki/Presentation-ala-project>

8.2 Development technologies and frameworks

Ansible

Platform for configuring and managing computers

<http://docs.ansible.com/>

Creation of Ansible playbooks for ALA software (README file)

<https://github.com/AtlasOfLivingAustralia/ala-install/blob/master/ansible/README.md>

Bootstrap

Collection of tools for creating websites and web applications

<http://getbootstrap.com/2.3.2/getting-started.html>

GitHub

Distributed source code management and revision control

<https://github.com/>

Groovy

Object-oriented programming language for the Java platform

<http://groovy-lang.org/documentation.html>

Grails

Web application framework, Groovy version of Ruby on Rails, to use in Java development

<https://grails.org/wiki/Documentation>

<http://grails.org/doc/2.3.x/guide/>

Vagrant

Create and configure development environments

<https://www.vagrantup.com/docs/>

Tomcat

Open source web server with implementation of the Java Servlet, JavaServer Pages, Java Expression Language and Java WebSocket technologies

<http://tomcat.apache.org/tomcat-7.0-doc/index.html>

8.3 Data management

Apache Cassandra

Open source distributed database management system

<http://cassandra.apache.org/>

MySQL

Open source relational database management system

<https://dev.mysql.com/doc/>

<https://www.mysql.com/>

Apache Solr

Open source Enterprise search platform

<http://lucene.apache.org/solr/>

<http://wiki.apache.org/solr/>

8.4 Biodiversity data formats

Darwin Core

Stable standard reference for sharing information on biological diversity

<http://rs.tdwg.org/dwc/terms/>

GBIF (2010). Darwin Core Archives – How-to Guide, version 1, released on 1 March 2011, (contributed by Remsen D, Braak, K, Döring M, Robertson, T), Copenhagen: Global Biodiversity Information Facility, 21 pp, accessible online at:

http://links.gbif.org/gbif_dwca_how_to_guide_v1

GBIF - Darwin Core Archive Assistant

Tool helping to prepare a Darwin Core file

<http://tools.gbif.org/dwca-assistant/>

GBIF Community Site

<http://community.gbif.org/>

9 Internationalization

Grails uses the standard Java way of providing [internationalisation](#) - using `messages.properties` and the `g:message` [taglib](#) for use in GSP pages. Language detection can be triggered by the browser providing a locale value or by providing an additional request parameter called `lang`. E.g. add `?lang=es` to a request URL.

The client webapp (e.g. generic-hub) inherits some internationalisation (i18n) properties from the biocache-hubs plugin, which in return, inherits further properties from biocache-service (via a [webservice call](#)). Custom i18n properties can be added/overridden by including the appropriate messages file. A complete list of the i18n codes (and English) translations can be seen at:

```
~/generic-hub/messages/i18n/messages_en-US.properties
```

9.1 About the ALA portal User Interface

There are many ways to include the i18n in your data portal. The way most used is included the selector of the different languages support by the data portal in the header. Here you can see some examples:





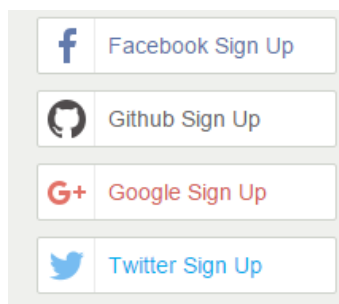
Illustration 62: Examples

9.2 Step-by-Step guide to UI translation

The first thing that we have to do to translate our interface, it is translating our `messages.properties` using the Crowdin portal. Crowdin is a localization project management platform and translation tool for developers and website owners. We make it easy to work with content being translated using our collaboration platform.

Then now we are going to suppose that we want that your interface supports Brazilian Portuguese.

Firstly you have to register yourself <https://crowdin.com>. You can use one of these accounts:



*Illustration 63:
crowdin.com*

or create a new crowdin account in <https://crowdin.com/join>

The next step that you have to do is accessing into the page of project: <https://crowdin.com/project/ala-i18n> and request the access to the project to one of managers of the project that you can see in the right side of the page:



*Illustration 64:
Managers*

When you can access such as editor/translator, you can begin to do the translation of one specific language. To do that, you have to access to the page of the project and you have to select the language that you want to translate:

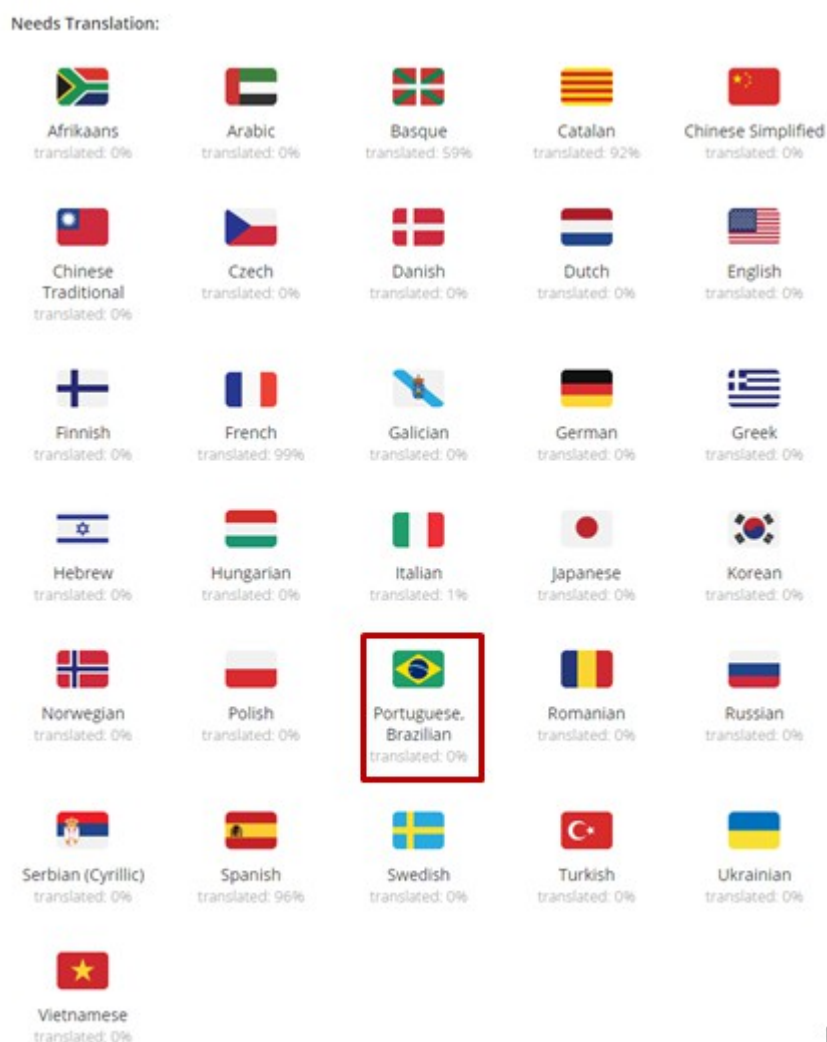


Illustration 65: Languages

In the below image you can see that we have 4 projects and each one has their own `messages.properties` that contains the properties which will be used in the related module and in this way it carries out the i18n:



Illustration 66: Projects

At this moment all `messages.properties` of this language are in the 0 % of progress, when we do the translations these bars increase.

How can we translate each variable with this platform? Very easily!!

We have to select one of the previous `messages.properties`, e.g: biocache-hubs. The view that we have below is divided in three sections:

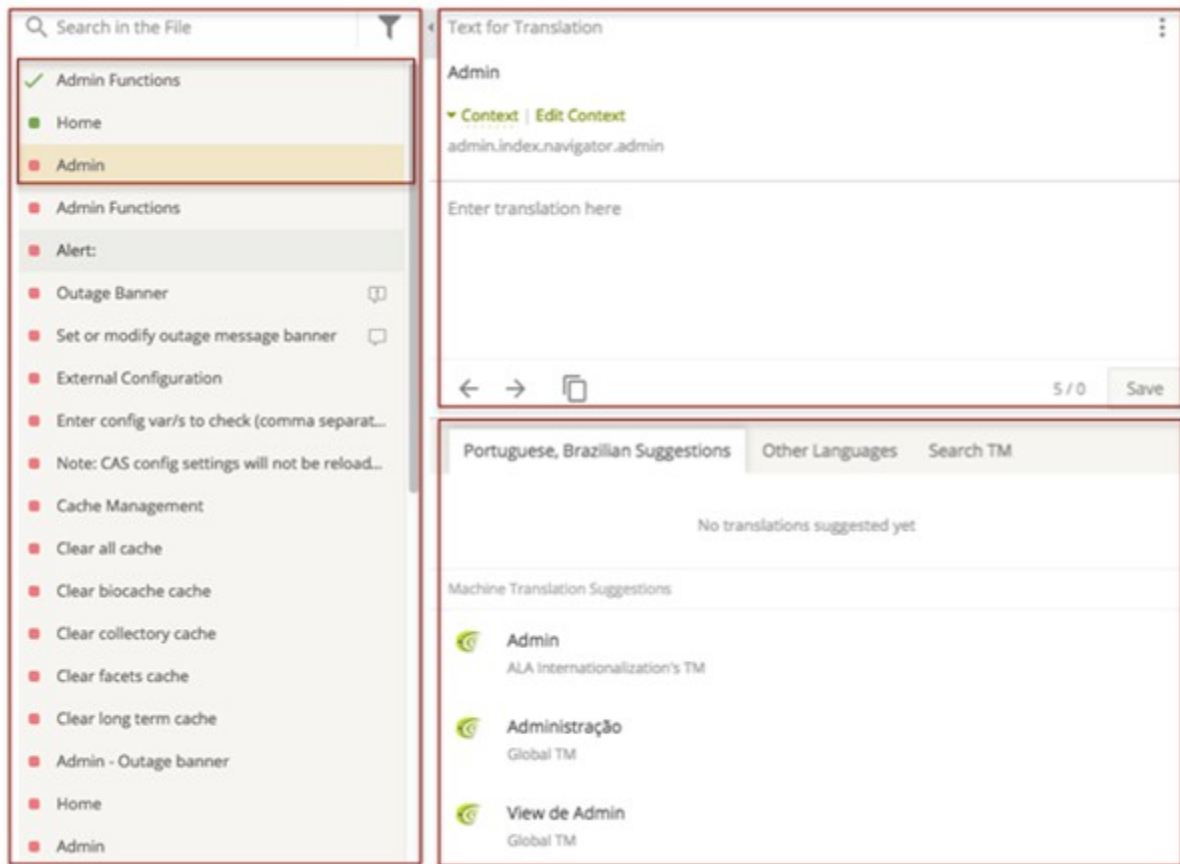


Illustration 67: Translation

The left column contains all the texts for translation, each one is related with one variable defined in the `messages.properties` file. Each variable has three status:

- Red: no translated
- Green: translated
- Green light: validated

The right column is divided in two sections:

- The down section suggests us several possibilities of translation.
- and in the up section we must write the final translation.

Once we write the translation, we have to click on "Save" button, then the variable changes its status, no translated by translation done, red by green.

The mechanism to do the rest of translation it's the same to the rest of variables.

Other relevant aspect very important to do the translation it's that the user must know the context where each variable goes, if you have some doubt, please don't hesitate and put in contact with some manager to request more information.

Each translation need to be validated by one manager. In this way, when you are sure that all the translations done, have been translated correctly, you must request the validation to one manager.

When we have finished to translate the `messages.properties` file of one of modules, we can install that file in the related module. If you want to download it you have to send a request to one of the managers, he will compile the project, and after he will send the final `messages.properties`.

Now we can add the new `message.properties` in the `i18n` directory of your own project. We will see that in the next chapter.

9.3 Language configuration

Grails `i18n` files are found in:

```
/grails-app/i18n
```

and follow the Java naming convention of `messages_xx_XX.properties`. The ALA is keen to get contributions for non-English translations, either via pull requests to biocache-hubs project or by sending us a patch file or individual properties files.

Note: If you want to contribute in the `i18n` project, please visit the page of the project: <https://crowdin.com/project/ala-i18n/> and request your participation to one of managers.

9.4 Multi-lingual support

How to include language selector in your project?

Source: <http://grails.org/plugin/lang-selector>.

In this page you have the instructions to do the correctly configuration of lang-selector plugin in your module. Now, we are going to suppose that you want to install and configurate this plugin in generic-hub.

- step 1: You have to include in the section of plugins of `BuildConfig.groovy` the next code:

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

- step 2: After this, you can add the tag to your gsp. For example, you can include it in the header sections of /grails-app/views/layouts/generic.gsp to see the lang selector in all views of your generic-hub module.

```
<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>
      <li><langs:selector langs="es"/></li>
    </ul>
  </li>
</ul>
```

- step 3: 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"]
```

Also, you can do this in the collectory module.

Now we are going to suppose that you don't want to have a flags, maybe for more accessibility or for other reasons. In this case, you have to modify the code of this plugin. To do that you have to download the plugin, calling it locally in the /grails-app/BuildConfig.groovy, and you have to modify /lang-selector-0.3/grails-app/views/langSelector/_selector.gsp:

BuildConfig.groovy:

```
grails.plugin.location.'lang-selector' = "plugins/lang-selector-0.3"
```

_selector.gsp:

```
<div id="lang_selector">
<span style="color:white"><g:message code="1" default="|"/></span>
<g:each in="${flags.keySet().sort()}" var="lang">
<%--
```

uri + lang = with this expression, we are building the url that redirects the application to the related view in that language "lang".

```
--%>
    <a href="{ uri + lang }" title="{message(code:'title.lang_link')}">
        <span class="lang_flag ${ lang==selected.toString()? selected_class :
not_selected_class }" style="margin-left: 14px;">
```

```
<%--
```

In this section we are checking what language we are going to apply. We have detected it, we can specify the message that we want to show.

```
--%>
```

```
    <g:if test="{flags[lang] == 'catalonia'}">
        <g:message code="1" default="Català"/>
    </g:if>
```

```
    <g:if test="{flags[lang] == 'gb'}">
        <g:message code="1" default="English"/>
    </g:if>
```

```
    <g:if test="{flags[lang] == 'es'}">
        <g:message code="1" default="Español"/>
    </g:if>
```

```
<%--
```

We can also include the flag of language instead of a message

```
--%>
```

```
    <%----%>
        </span>
    </a>
    <span style="color:white"><g:message code="1" default="|"/></span>
</g:each>
```

10 Collaboration Channels

You can help developing ALA software , even if you are not a developer. There are several tasks, and certainly one role is right for you.

Join ALA mailing list

Join the ALA mailing list managed by GBIF (<http://lists.gbif.org/mailman/listinfo/ala-portal>) to get to know what is being developed, request a feature, report issues and enter in contact with the rest of the ALA software community.

You can also join to Atlas of Living Australia HipChat, by asking to join to the Atlas of Living Australia project manager (<https://atlasoflivingaustralia.hipchat.com/home>) to chat with other members of the ALA community

Contact your GBIF node

Do your country/organization has an implementation of ALA portal? If that is the case, your GBIF node most probably is already involved with the ALA community, and can help you identify the best contacts so you can join the ALA community, and which role you can play in it. See [list of GBIF participants](#).

How to contribute - community roles

Your contribution to ALA software can be on different ways, depending on your skills or talent. Join the ALA mailing list and tell what you would like to contribute. There are several roles that you can play.

Software developer

If you are a developer, you can contribute by:

Helping to turn the support of ALA global

The Atlas of Living Australia was developed for the Australian context. In early stages, it was not planned to support a global community, so many parts of the code need to be generalised from an Australian context (e.g. administrative regions of Australia) to a global context. Also, some of the modules did not have the i18n support inserted from the beginning. You can help in making ALA platform global in support, which is simultaneously a great deal to get to know better the code.

Developing a new module of Atlas

The platform of Atlas of Living Australia is easily expandable by the addition of new plugins or modules that adds new features to the software. Also the architecture of the system includes a layer of web services that can easily allow the integration with other tools or software. If you have the need and the capacity to expand ALA software, developing new modules or plugins, propose that to be included in the future development roadmap of ALA. Visit <https://github.com/AtlasOfLivingAustralia/documentation/wiki/RoadMap> to check what are the future development plans of ALA platform.

Adding more features to an existing module

Likewise, adding features or making changes to a module in order to respond to specific implementations of the portal could be a small size project where your help could be valuable.

Reporting and issue

Report an issue, either by sending a message to the GBIF ALA-portal mailing list or filling an issue report in the GitHub repository of the corresponding ALA module.

Helping to implement ALA at a national level for a national GBIF node

The implementation of ALA portal at the national level is an intensive task where any help is welcome. In this document we have found many tasks that need to be performed:

- customization of interfaces of each module to the national layout
- data management of national species list
- data management of national data resources

Writing documentation

This manual can be expanded, either by reporting other implementation environments, or adding experiences on the other modules of ALA. If you had that knowledge, share it to other writing a chapter about it.

Atlas User

If you are an Atlas user, your contribution is essential to:

Give Feedback

Have you found any error on data, or how the data is handled. Your feedback is essential to track issues and promote corrections. Please use the existing

contacts on the portals implementation to report the issue, which will help to benefit the quality of the portal.

Propose new functionalities / topics

Do you have ideas on how to improve the portal usability, or new features that would benefit its use? Propose that, helping to expand the use value of the portal.

Write documentation

From small how-tos to manuals, you can contribute to create or expand the documentation for users for many of the existing ALA modules. That will be even better if you build that documentation based on examples based on real uses where the ALA was useful for you.

Translate documentation

Likewise GBIF, we should help to make ALA platform global by supporting different languages. You can contribute on this by helping to translate documentation or manuals about ALA portal use, or the resource files that populate the web interfaces of ALA portal. The translation of ALA resource files is done using the user-friendly web platform Crowdin. Check which languages need you collaboration [here](#).

Advertise ALA portal

Are you aware of colleagues, organisations or institutions which could find value in using or developing ALA portal? For example, do you work with schools, which students' projects can be helped by biodiversity information? Or do you have contacts with citizen science projects that produce primary biodiversity information, but need a web platform or mobile application to manage observations? Are you a member of a project or network that needs a web platform a national or international project on biodiversity information of a special topic? The platform of ALA can provide support for these and more, so it is important that all the potential users know that ALA is available to visualize, access and develop solutions about biodiversity data. And remember, the best advertising is by showing cases with examples.

Annex I: Functional Glossary

DataResource	This is the dataset link to the Darwin Core Archive or CSV uploaded. On the dataresource administration page, you will find a link to the DwC Archive. All data are link to one dataresource.
DataProvider	DataProvider represents the global organization that could be composed by several Institutions. It could be use for a data aggregator for example.
Contact	User link to a dataResources, collections, institutions or dataProviders. You can give editor role to any contact.
Institution	This is the owner of the data.
Collection	This is a related to collection dataset (e.g. : herbarium specimen). It can be linked to one or more dataResources.
ProviderCode	You create provider code using collection code and institution code that you can find on your dataResource.
Provider map	This is the mechanism that links dataresource to institution and collection or just institution.
Administrator	It is one of the roles using in ALA system. With this one, you can manage the entire software. Administrator can create, modify and delete the information related with the Data Providers, Institutions, Collections, Data Resources and Contacts
Editor	It is one of the roles using in ALA system. With editor privilege, you can manage, modify metadata from your own dataresources/collection/institution.
Metadata	Information about dataresource, collection, institution or contact.

Annex II: Illustrations

Illustration Index

Illustration 1: ALA Architecture.....	8
Illustration 2: Modules overview.....	9
Illustration 3: Occurrences search modules.....	10
Illustration 4: Collectory modules.....	12
Illustration 5: Species modules.....	14
Illustration 6: Datahub modules.....	15
Illustration 7: Myxotropic Datahub.....	16
Illustration 8: Myxotropic Occurrences.....	17
Illustration 9: Data Hubs.....	18
Illustration 10: Regions modules.....	19
Illustration 11: Spatial portal modules.....	20
Illustration 12: Spatial Portal.....	21
Illustration 13: Spanish data portal infrastructure.....	24
Illustration 14: Australian Infrastructre.....	26
Illustration 15: French Infrastructure.....	27
Illustration 16: Ubuntu.....	28
Illustration 17: Ubuntu login.....	29
Illustration 18: Ansible.....	30
Illustration 19: ALA Demo.....	31
Illustration 20: Flow of information.....	32
Illustration 21: Plug tree view.....	36
Illustration 22: GBIFrance-hub tree view.....	36
Illustration 23: Create your specific layout.....	38
Illustration 24: Internal file.....	39
Illustration 25: Add a new module.....	40
Illustration 26: Institution page (Australia).....	43
Illustration 27: Institution page (France).....	44
Illustration 28: Results page (Australia).....	44
Illustration 29: Results page (France).....	45
Illustration 30: Entity relationship diagram.....	46
Illustration 31: GBIF France.....	47
Illustration 32: Contact form.....	48
Illustration 33: Resource Contact.....	49
Illustration 34: Provider map 1.....	49
Illustration 35: Provider map 2.....	50
Illustration 36: Solr.....	55
Illustration 37: Solr index backup.....	56
Illustration 38: Welcome page.....	59
Illustration 39: Basic search.....	59
Illustration 40: Advanced search.....	60
Illustration 41: Result page of a search (Map).....	61
Illustration 42: Search result page (Record list view).....	61
Illustration 43: Result page of a search (Charts).....	62

Illustration 44: Collections Page (Maps).....	63
Illustration 45: Filter fauna on the collections page.....	63
Illustration 46: Datasets page.....	64
Illustration 47: Dataset details page.....	65
Illustration 48: Georeferenced data.....	66
Illustration 49: Administration main page.....	67
Illustration 50: Information page of a collection.....	68
Illustration 51: Institutions page.....	68
Illustration 52: Data Providers page.....	69
Illustration 53: Data resources page.....	69
Illustration 54: Contacts page.....	70
Illustration 55: Contact detail page.....	70
Illustration 56: Provider codes page.....	71
Illustration 57: Provider maps page.....	71
Illustration 58: Map view.....	73
Illustration 59: Detecting errors in the Map view.....	74
Illustration 60: Flag issue in the Map view.....	75
Illustration 61: Flag issue in the record view.....	75
Illustration 62: Examples.....	79
Illustration 63: crowdin.com.....	79
Illustration 64: Managers.....	80
Illustration 65: Languages.....	80
Illustration 66: Projects.....	81
Illustration 67: Translation.....	82