FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Developing tools to use metagenomic databases as a global surveillance system

Tiago Cardoso

Mestrado em Engenharia Informática e Computação

Supervisor: Nuno Fonseca

Second Supervisor: Rui Camacho

January 3, 2023

Developing tools to use metagenomic databases as a global surveillance system

Tiago Cardoso

Mestrado em Engenharia Informática e Computação

January 3, 2023

Abstract

Due to the advancements in recent years, the number of DNA sequences from environmental sources has increased significantly since the analysis process has become both quicker and cheaper. Nowadays, the data sets generated comprise a very large number of DNA sequences. Despite that large number, only a smaller number are usually relevant for specific studies. These studies are usually focused on a specific subset of that data, and the rest is not used. There are nowadays a large number of public metagenomic databases [1][13][16] that the scientific community uses to store data generated during their studies.

This thesis aims to develop tools that help scientists to identify the presence of a given species around the globe. To accomplish that, the tools would make use of the most relevant data stored in each public database and retrieve it.

Hence, the first stage will revolve around querying the different databases for a specific species' data in an automated way. Then we will study the metadata associated with the data submissions that are of interest. From that data, we will extract information regarding the taxa's location to estimate its presence across the world. That is to be presented in an easy-to-read visual presentation, preferably, a website.

We hope that this thesis work will help the communities keep track of potentially dangerous taxa in their ecosystems and take preventive action to combat their negative impact.

As a proof of concept, illustrate the use and relevance of our work reporting four case studies were we have applied our tool in the detecting the presence of four taxa, *Batrachochytrium dendrobatidis*, *Sphaerothecum destruens*, *Vespa mandarinia*, and *Giardia lamblia*.

Keywords: DNA sequences, metadata, metagenomic databases

Resumo

Devido aos avanços nos últimos anos, o número de sequências de ADN obtidas de amostras ambientais tem aumentado significativamente, uma vez que o processo se tornou mais rápido e barato. A informção gerada a partir da análise destas amostras gera um vasto número de sequências de ADN que não são todas o foco de interesse do estudo que levou à sua geração. Estes estudos geralmente estão focados numa pequena parte dessas sequências e o resto não é usado. Existem muitas bases de dados metagenómicos públicas [1][13][16] que a comunidade científica utiliza para guardar os dados gerados pelos seus estudos.

O objetivo deste caso de estudo é desenvolver ferramentas que forneçam informação à cerca da distribuição de uma determinada espécie no planeta. Para esse efeito, as ferramentas têm de se servir dos dados relevantes armazenados nas bases de dados públicas disponíveis.

Deste modo, o primeiro objetivo é retirar os dados das diferentes bases de dados de forma automática para uma determinada espécie. Depois é analisar os dados retirados da informação recolhida. Dessa análise, é necessário retirar informação referente à localização das espécies para estimar a sua distribuição no mundo. Essa informação deve ser apresentada de forma visual e fácil de compreender, preferencialmente, num formato online.

Estas informações vão ajudar as comunidades a monotorizar espécies potencialmente perigosas, de modo a tomarem ações preventivas para combater o impacto destas espécies.

Neste caso de estudo vamos considerar quatro espécies *Batrachochytrium dendrobatidis*, *Sphaerothecum destruens*, *Vespa mandarinia* e *Giardia lamblia*.

Keywords: sequências de ADN, metagenómica, bases de dados metagenómicos

Acknowledgements

This adventure was very turbulent. I was not ready to face all the struggles than accompany the end of another chapter in my academic journey in the current times. I need to thank my friends that kept my sanity and helped me moving forward and avoid getting chained by the more complicated situations. Without my friends to support me I would have given up a long time ago, but they showed me that as long as someone is there everything can be rebuilt.

I must thank my family for their unconditional love and for being a safe haven of stability so that I can do my best. They have always done everything so that I can have all the possibilities and without them I would never dreamed of getting were I am now.

I need to acknowledge to my supervisors for being available to help me out when I was struggling the most.

A special thanks to Maria Teresa Ferreira for being the one who has the grueling task of pulling me out of deep dark waters so I can see the light and for believing in me when even I couldn't.

Author

"Life happens wherever you are,
wheter you make it or not."

Uncle Iroh

Contents

List of Figures

List of Tables

Abbreviations

API	Application Programming Interface
BLAST	Basic Local Alignment Search Tool
CSV	Comma Separated Values
DB	Database
DNA	DeoxyriboNucleic Acid
EBI	European Bioinformatics Institute
EMBL	European Molecular Biology Laboratory
ENA	European Nucleotide Archive
etc	*Et Cetera*
JSON	JavaScript Object Notation
GBIF	Global Biodiversity Information Facility
GIS	Geographic Information System
NCBI	National Center for Biotechnology Information
NGS	Next-Generation Sequencer
PCR	Polymerase Chain Reaction
REST	REpresentational State Transfer
RNA	Ribonucleic Acid
TSV	Tab Separated Values
URL	Uniform Resource Locator

Chapter 1

Introduction

1.1 Context

A taxon is a taxonomic group of any rank, such as a species, family, or class. There are some taxon that have more reasons to be monitored than others. For example, invasive species heavily impact the ecosystems on which they insert themselves. Either by lack of predators or by evolving in a much more competitive environment than the invaded one, their presence threatens local fauna and flora, which can then have negative impacts in economic or health activities of the region affected. Therefore, the existence of a surveillance system that tracks the observations of species will provide a means to determine if the species is increasing its numbers or in decline.

One of the methods employed to study the environment is the analysis of samples that represent that environment. Sampling is a complex endeavour in order to generate data of good quality that can be used by researchers to further their studies. It is essential that prior to sample collection the purpose of the study is well defined. The purpose will dictate the type and scope of the sampling program and therefore will establish the requirements to sample correctly and effectively. Every sampling program should contain details about sampling site and specific locations, sample source, number and frequency of samples (daily, weekly, monthly, etc.), type of sample, field measurements, field quality control, and the sample collector(s). There samples can be divided in four types: grab, composite, duplicates, and split. Grab sample is a sample that was collected at a specific location in a specific date. Composite samples are a mixture of grab samples that were retrieved from the same location but at different times. Duplicate samples are equivalent samples taken from the same place at the same time and are often used to evaluate the sampling process. Finally, the split samples result from carefully dividing a sample into two new samples that represent the same collection point and date of the original sample. The sampling process is vital to many studies and requires a lot of preparation to achieve the best results [6].

DNA sequencing is a technique used to determine the exact sequence of bases present in a DNA molecule or molecules. The Human Genome Project pushed DNA sequencing to its limits requiring testing the known techniques at the time and exposing their limitations when the techniques were used to sequence larger genomes. This project success demanded improvements in efficiency and scalability for the entire process of sequencing and was achieved in the 1990s. Since then the range and scope of DNA sequencing applications has also widen fuelled by breakthroughs in technologies that allowed for previous obstacles to be overcome [19]. Some of the advances made were a switch from dye-labelled primers to dye-labelled terminators, that would only allow one sequence reaction instead of four, a mutant T7 DNA polymerase that would incorporate the those dye-labelled terminators better, and methods enabling sequencing of double-stranded DNA, which eventually allowed paired-end sequencing. Paired-end sequencing generates twice the number of reads for equivalent amounts of time and effort in library preparations. This advancement provided the means to the high-throughput sequencing technology that is currently used in DNA sequencing. The sequencers that utilize massive parallel sequencing technology are vastly cheaper and faster than the previous methods. This factors contributed to the increased use of this method. The sequencers that operate with this technology are known as next-generation sequencers (NGS) [14].

The advances of high-throughput sequencing technologies have further increased the number of DNA sequences generated. These technologies allow for more studies to make use of DNA sequencing by providing very cheap, fast, and reliable results. Some of the sequences that are generated as results for studies come from environmental samples that can have many sources ranging from drinking water to salt water, soil, sediment, wastes, air or biological materials [6].

Many of the sequences from environmental samples are generated by untargeted techniques or sequencing from broad-range PCR assays that achieve a similar outcome. When utilising untargeted sequencing to analyse a sample from an environmental source, the outcome is the sequencing of all the different DNA molecules present in that sample. Therefore, some samples will contain sequences that are not interesting or relevant for the study behind their collection and analysis. The focus of such studies will fall upon only a particular subset of the DNA sequences generated. As such, there is a number of sequences that are submitted but were not the main target of the study and thus they are not tracked across different studies, and their presence in the sample is not thoroughly analysed.

These samples are submitted to public online repositories that present the sequences and associated sample metadata. This information includes location and date essential to attempt to find the distribution of a particular taxon around the globe. These platforms have built-in mechanisms that allow the researchers conducting analysis to classify their submitted samples. This feature allows the search of some results when trying to establish the occurrences of a determined species but in order to get more results it is necessary to gather more results based on the sequencing results of the available samples.

The public platforms available differ from each other in structure, main purpose, data stored, and how the data is made available [1][13][10]. Although some samples may be present in mul-

tiple databases, the information stored in each of them might differ and be more complete on one of the databases than the others. There is not a centralised method that gathers information across multiple platforms.Some databases also suffer from less and more inaccurate metadata about their samples and analysis but are working to change it. For example, the European Nucleotide Archive (ENA) may have some missing metadata when it comes to the samples in their database however they have started to enact better metadata sanitization in order to combat this problem [1]. Nevertheless, the samples can be used to track the species that are present in those samples using metadata such as the location where the sample was collected and when.

1.2 Motivation

Nature provides many challenges to human society. There is a fine balance between exploiting resources and depleting them. This balance also applies to the management of the fauna and flora in order to not damage the existent ecosystems and being able to utilises the resources that they provide.

Some species found in nature are detrimental to human socio-economic activities and even to public health. Most of these take time to operate and cause their damage but are not detected until its negative effects have manifested and it is too late to minimise their impact. Consequently, it is necessary to survey this species in order to anticipate the issues that those species may cause in the future.

The programs that track detrimental species such as invasive species, tend to be local and short-lived. Programs with a larger scope tend to record only scarcity or presence of species. Extensive surveillance normally occurs only after the negative effects start to manifest, such as, economic damages or dwindling numbers of threatened species [15]. In 2010, it was proposed that isolated datasets made it significantly harder to predict the evolution of invasive species [5].

1.3 Objectives

The work undergone during this thesis will revolve around two main objectives:

1. create a set of tools that retrieves and process data from different databases and process it;
2. display the information retrieve for each taxon in order to allow an overview of the spread of each taxon;

The first thing to do is to scout the largest public metagenomics databases to understand their strengths and weaknesses. One of the main aspects of this initial survey is to assess the availability of data of each database in order to understand the viability of utilising them. After the target databases have been decided upon, it is necessary to retrieve the information regarding samples containing the specified taxons. This process will have to be adapted to each different infrastructure in order to accommodate each. After collecting all the data, it is necessary to process and

analyse the metadata in order to be able to normalise it and to be able to compare it regardless of the collection method. The results will finally be presented in a visual manner represented by the location on an interactive map, tables containing all the matches obtained and some graphics containing details about the metadata that was gathered.

This thesis will focus on analysing the spread of four specific species:

1. Batrachochytrium dendrobatidis, a parasitic fungus that has been associated with population declines in endemic amphibian species;
2. Sphaerothecum destruens, a parasite of fish;
3. Giardia lamblia, a parasitic microorganism that colonizes the small intestine
4. Vespa mandarinia, the world's largest hornet commonly known as the Asian giant hornet;

Analysing the difficulties encountered in the developing phase and information available about these four species will be essential to achieve a base for understanding the possibility of further developing the tools necessary for this thesis.

The actualization of the objectives described is the implementation of *MetagenClues*, a set of tools that interact together to accomplish those objectives. The difficulties encountered will expose where there is room for further development in future work in order to monitor and possibly ascertain areas endangered by those species.

1.4 Document Structure

In Chapter 1 the basic concepts required to understand the goals of this thesis are laid out to contextualize the reader. Chapter 2 overviews the possible sources for data to be used by the tools developed during this thesis. For each source there will be an introduction, brief description of the data collection process and APIs if existent, and what difficulties were expected and encountered. The third chapter will describe the implementation of *MetagenClues*. The Chapter 4 shows some applications of tools in concrete scenarios. The Chapter 5 exposes the main conclusions achieve through this work, and future improvements that can be made to improve the current *MetagenClues*.

Chapter 2

State of the art

In order to create a functional surveillance system, the most important information is time and spacial location. To achieve a better understanding of the distribution of a species the possession of these two attributes is required which are present in the metadata of collected samples. Information of this kind is submitted by researchers to the databases they are working with.

Multiple online public platforms provide storage services and act as metagenomic databases [1][10][13]. It is not simple to understand the raw results of DNA sequencing and each platform records those results in accordance to the software and pipelines they use. For example, ENA stores in the metadata the National Center for Biotechnology Information (NCBI) taxonomy identifier of the taxon present in the sample, while MGnify and GBIF utilise the name of the taxon to achieve the same result. This complicates the standardization of the information provided by different databases. Each platform will have different weaknesses and strengths. For example, while the European Nucleotide Archive has a larger database it lacks in metadata quality when compared to MGnify. In this chapter, public databases and what can be done with the information stored there will be reviewed. The use of NCBI BLAST+ [4] can be a source of more data that it is not covered by interacting directly with the other databases. Therefore, it is also necessary to address it in this chapter.

2.1 European Nucleotide Archive

2.1.1 Introduction

The European Nucleotide Archive (ENA) is an online platform that allows the management of sequenced data. It is one of the most extensive public collection of sequenced data. This platform facilitates the sharing and dissemination of the information and serves as an archive to keep the submissions for later review. The contents of this platform are the culmination of 39 years of breakthroughs in sequencing and archiving techniques.

ENA strives to provide their users with the means to access their data in easy and comprehensible ways. Accompanying the growth of their database, new endpoints and tools were developed in order to make interoperability simpler. Also in the recent years there were implemented stricter

guidelines regarding the submission of metadata in order to minimize the lack of information regarding some data in their database. In addition, ENA has introduced new data types and deployed a new browser using more updated technologies to facilitate its use [1].

2.1.2 ENA's Portal API

ENA, in accordance to their principle of interoperability, has an API that facilitates integration with external applications. The primary goal of this API is to allow access to all available data in ENA. The API allows to access public data as well as pre-publication data but in order to access non public data it is necessary to provide authentication which is not required otherwise.

There are six endpoints available to users: *doc*, *search*, *results*, *searchFields*, *returnFields*, and *controlledVocab*. The *search* endpoint is the one capable of retrieving the data from ENA and will be essential to any user trying to access ENA data programmatically. The other endpoints main function is to provide the full documentation, in the case of *doc*, or to provide shortcuts to specific parts of the documentation, in the case of the remaining ones. This API [7] can be a powerful tool to provide a quick and fairly comprehensible access to the ENA database. It has many functionalities (Figure 2.1).

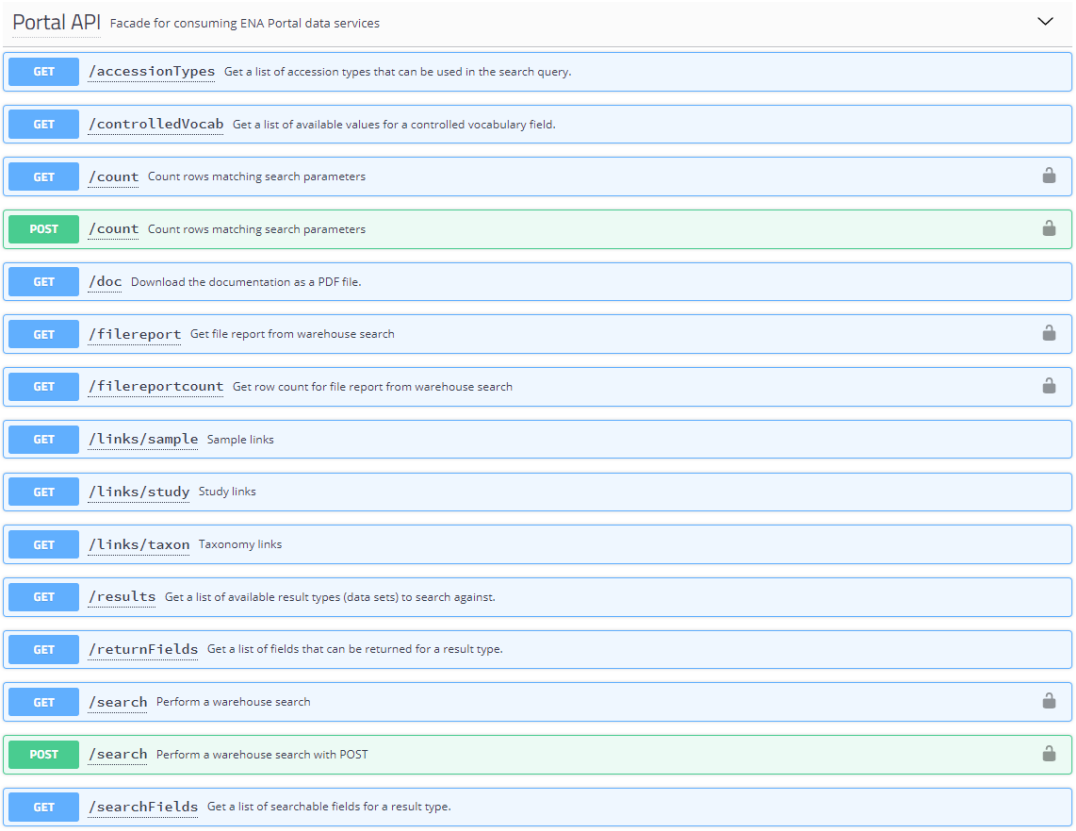

Figure 2.1: ENA's Portal API requests

The **/doc** request provides EMBL-EBI's documentation regarding the use of the ENA portal API. The documentation dated September 2018 details the data portals available (*ena* and *pathogen*), the seventeen (17) different available in *ena*, from which the *sample* result will be the one primarily used to interact with the ENA's Portal API.

As stated in the EMBL-EBI's documentation mentioned previously, the taxonomy identification is made based on the American National Center for Biotechnology Information (NCBI)[8] taxonomy identifiers. As a disclaimer, the NCBI firmly states that its identification values are not any form of an international convention.

For example, ***Giardia Lamblia*** *is under the name* ***Giardia intestinalis*** *and has the identification number* ***5741***.

With this information, it is possible to obtain the samples in the database that have the presence of a specific taxon by using the **/search** and filing the appropriate search fields as seen in Figure 2.2.

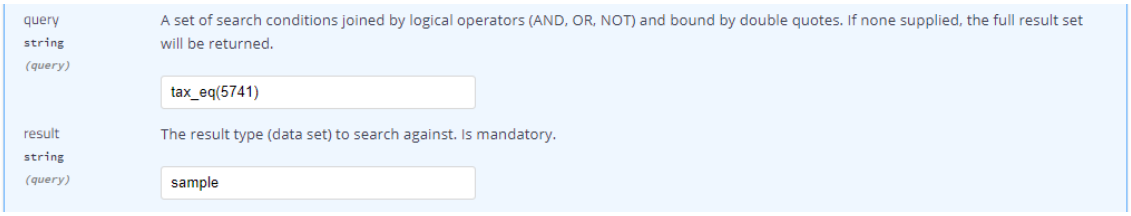

Figure 2.2: ENA search request parameters

Through the parameter "**fields**", the amount of information presented for each entry is configurable to return only a subset of the columns associated with the samples or can be set to "all", in order to retrieve the full set of sixty-four (64) fields regardless if those fields have no values. The fields that are more interesting for developing a surveillance are *collection_date*, *country*, and *location*.

All the results of any performed search can be return in two formats: JSON or TSV. By default, a TSV report is provided and this is the format that will be used in the implementation described in the next chapter (Section 3.1.2.1).

The main issue with this platform is that some of the samples submitted lack metadata, particularly coordinates from the point of collection (Table 3.1). This flaw is a setback since the location where the samples were taken is crucial information to achieve this case study goals and even if the country were the sample was taken is provided, the scope is still too large and affects the efficiency of a possible surveillance system.

2.2 National Center for Biotechnology Information

2.2.1 NCBI BLAST+

In the network of the European Molecular Biology Laboratory's European Bioinformatics Institute (EMBL-EBI) there are several tools provided [12] and the Basic Local Alignment Search

Tool (BLAST) is one of the most relevant. Based on the latest iteration of the NCBI BLAST+, the sequence similarity search finds regions of similarity between biological sequences [4]. The program compares nucleotide or protein sequences to the sequences stored in ENA databases and calculates the statistical significance. The main purpose of this tool is to provide an unknown nucleotide or protein sequence and obtain some insight about the structure and function of that sequence through possible matches. This feature can be used to search for similar sequences of a known sequence from a target species in order to attempt to find samples that were subject to DNA sequencing but were not marked as containing that taxon. There are some limitations to this tool since only one thousand (1000) results can be provided at a time and, because the job takes time to execute in the servers, the data retrieved can not be updated in real time but instead should be done only once at the user discretion. There is also another reason to not update in real time to avoid the server to be swamped with requests which can even trigger an automated response to block the user and deny the service.

2.3 MGnify

2.3.1 Introduction

MGnify [13] is a free to use platform that specialises on managing data generated from sequencing microbial populations. The platform has reported having more than doubled the number of their public data sets in 2019 compared with the previous two years[13]. Its main feature is providing standardised pipelines. MGnify is currently on version 5.0, which focused on bringing more interoperability to the tool.

This platform provides a free service for submission of raw metagenomics sequence data and metadata to ENA and then analyse that data in MGnify. By using this services the researchers provide consent to access the submitted data for analysis in spite of being conducted in a different platform.

MGnify excels in providing complete metadata about the samples stored in their database, going as far as to verify if the location coordinates are reversed and other clerical errors of the kind.

2.3.2 MGnify's API

This platform provides an API, but it is much more optimised to run the analysis through their pipelines and get the metadata associated rather than to filter the sequences based in their contents.

This API's documentation leans heavily on a client-side Python implementation and has some usefully requests as seen in Figures 2.3 and 2.4.

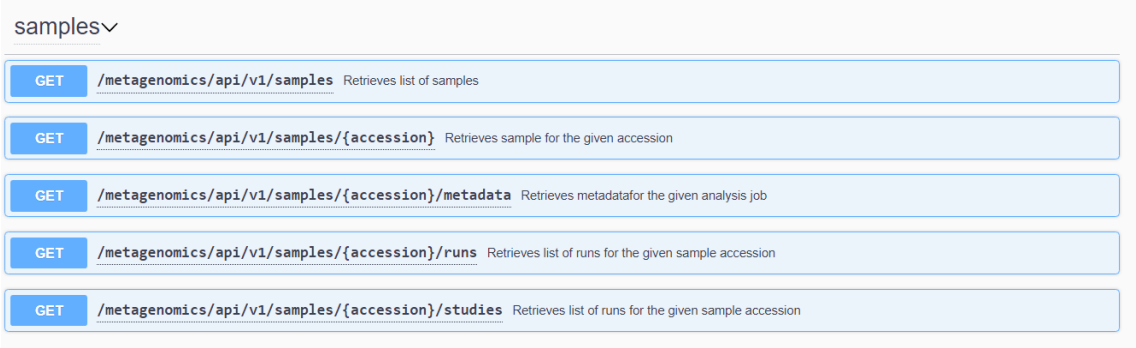

Figure 2.3: MGnify sample related requests

As it is possible to ascertain from Figure 2.3 the requests intended to retrieve the metadata about samples are structured in a way that only allows one sample to be considered at a time. This is a problem regarding scalability by not allowing to perform a bulk request. The first request visible in the Figure 2.3, */metagenomics/api/v1/samples*, allows to retrieve a list of samples that match the user supplied criteria. However, there is also a scalability issue with this request since the response is limited to one hundred (100) results a at time. Thus, multiple requests must be done in succession which will incidentally take more time the more results are available.

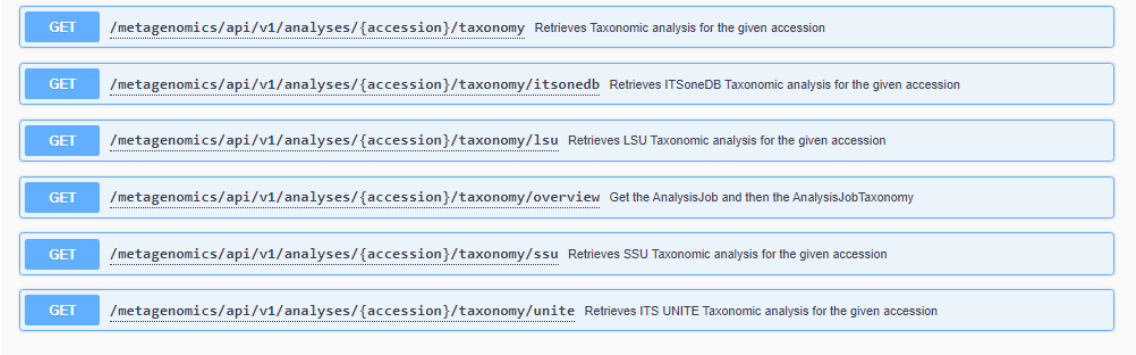

Figure 2.4: MGnify taxonomy related requests

2.4 Global Biodiversity Information Facility

2.4.1 Introduction

The Global Biodiversity Information Facility (GBIF) online platform [10] is a free to use international network to share and disseminate data through the many open-source tools[3] it offers.

This platform has maps that show the presence of a species in the world across a specified period of time. This platform has a lot of data entries because the presence of a species is registered through occurrences. This denomination can encompass a multitude of very distinct types of observation. The most straightforward is the human direct observation were a person reports the

sighting of an exemplar or more of a species in a specific location. Therefore, the occurrences are not purely metagenomic data. The higher amount of data also encompass more data entries with incomplete metadata. However, GBIF does everything possible to secure the validity of the data submitted in the platform.

2.4.2 GBIF's API

GBIF has an API that is divided into five sections:

- **Registry:** In this subsection of the documentation endpoinst are presented that allow the management of information about datasets, organizations, networks and the means to access their corresponding endpoints.
- **Species:** Provides services that rely upon taxonomic identification.
- **Occurrence:** Provides means to query the database with various filters in order to obtain relevant information. It has a limitation of a one hundred thousand (100,000) records per query but the platform offers a download service if a superior number is required.
- **Maps:** This services are meant to be a straight forward way of displaying GBIF contents in interactive maps on other websites. According to the documentation this services are intended to be used with GIS software.
- **News:** Is used to retrieve news or publications from or about GBIF

The **Occurrence** part of the API will be the one providing more accessible data. There are high volumes of data to be gathered but the limit of only accessing the first one hundred thousand (100,000) prevents access to the totally of the data using the available methods.

2.5 Summary

Table 2.1 summarizes the most relevant findings about the available public resources. The documentation quality was evaluated based on two factors: difficulty to learn and detail completeness. The quantity of data was evaluated by the results obtain from interacting with each API. The quality of the metadata was evaluated based on the completeness of the fields regarding temporal and spacial location in the data retrieved. Therefore, the quality of the metadata is relative only to the usefulness to the tools developed for this thesis. The quality will also not reflect on the quality of the databases of each platform since it is the responsibility of the researchers to submit the data. The metadata accessibility was based on the quantity of data available per request.

Table 2.1: Resources Expectation Summary

Resource	**API**	**Documentation**	**Quantity of data**	**Quality of metadata**	**Metadata accessibility**
ENA	yes	very good	high	average	high
GBIF	yes	very good	very high	average	high
MGnify	yes	average	conditioned	high	average

From the gathered information, GBIF provides an enormous amount of data, however the data gathered by this platform is not only metagenomical in nature as a sample that underwent DNA sequencing, including a large number of direct human observation.

ENA provides the most useful data complemented by the metadata cleansing done by MGnify when it is applicable. MGnify's API provided less data but more complete metadata for each match in that platform.

The documentations of ENA's and GBIF's API seemed more complete than the documentation provided for MGnify's API and were easier to understant. Furthermore, MGnify's metadata was stored in a less defined data structure than the other platform proving to be a more challenging information retrieval.

Chapter 3

MetagenClues

MetagenClues is a set of tools designed to extract information from different public databases to surveil species of interest. The set of databases include are the European Nucleotide Archive (ENA), MGnify, Global Biodiversity Information Facility (GBIF), and the National Center for Biotechnology Information (NCBI). The information contained in the latter overlaps with ENA and is used to take advantage of the NCBI BLAST+. The information retrieved is mostly metadata regarding the location and collection date of the samples that contain the taxon of interest. To gather the metadata is necessary to ascertain which samples are of interest and contain evidence of the presence of the desired species. The samples identifiers are called accession numbers and are different for each platform and are all unique. The different platforms provide methods for retrieving the identifiers of the samples that contain the targeted species. The only limitations lie in the submissions being a product of external users. Therefore the platforms can not guarantee that all the data was provided or that the data provided is completely correct.

3.1 Implementation

In order to work in any machine, the tools were developed in an Anaconda environment and allow customisation to a certain degree. To achieve more versatility, the operations of MetagenClues are compartmentalised, and each script communicates its results through TSV files. The tools interact in a cycle that has three phases: retrieve, process and store, and display. After all the scripts that collect information have stored their results in TSV files, the next set of scripts sets up a database, normalises the data, and stores all the results regarding a species. Different species have different databases. Lastly, the results that can be represented are compiled into an interactive map, a table containing all the data (even the samples that were missing some metadata), and graphics representing statistics of the different databases and data collected. The main technologies used to implement the tools were Python and R (Figure 3.1).

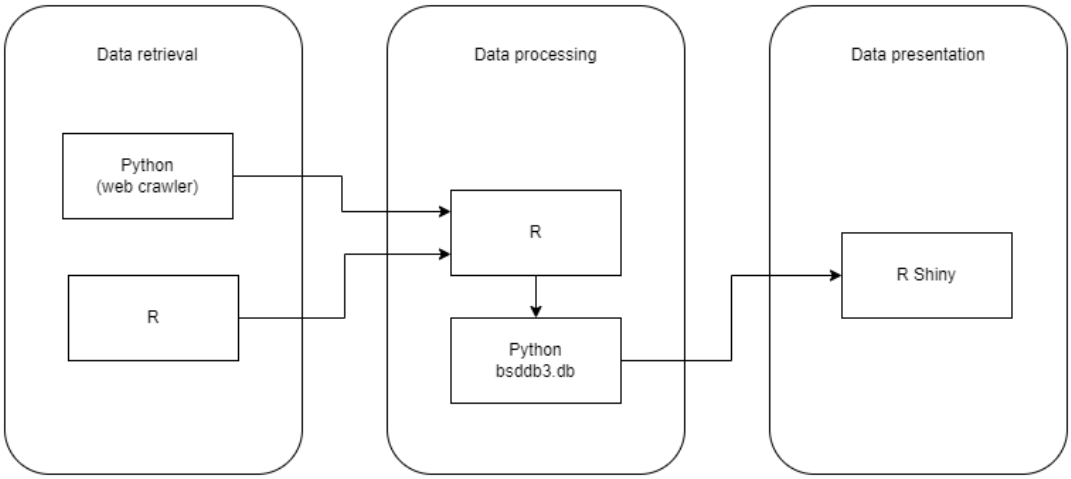

Figure 3.1: Technologies used in the solution

3.1.1 Architecture of the tools

MetagenClues is divided in many parts that construct a pipeline that gathers, processes and presents the information collected (Figure 3.2). The scripts that comprise *MetagenClues* can be grouped in three different sets according to their purpose.

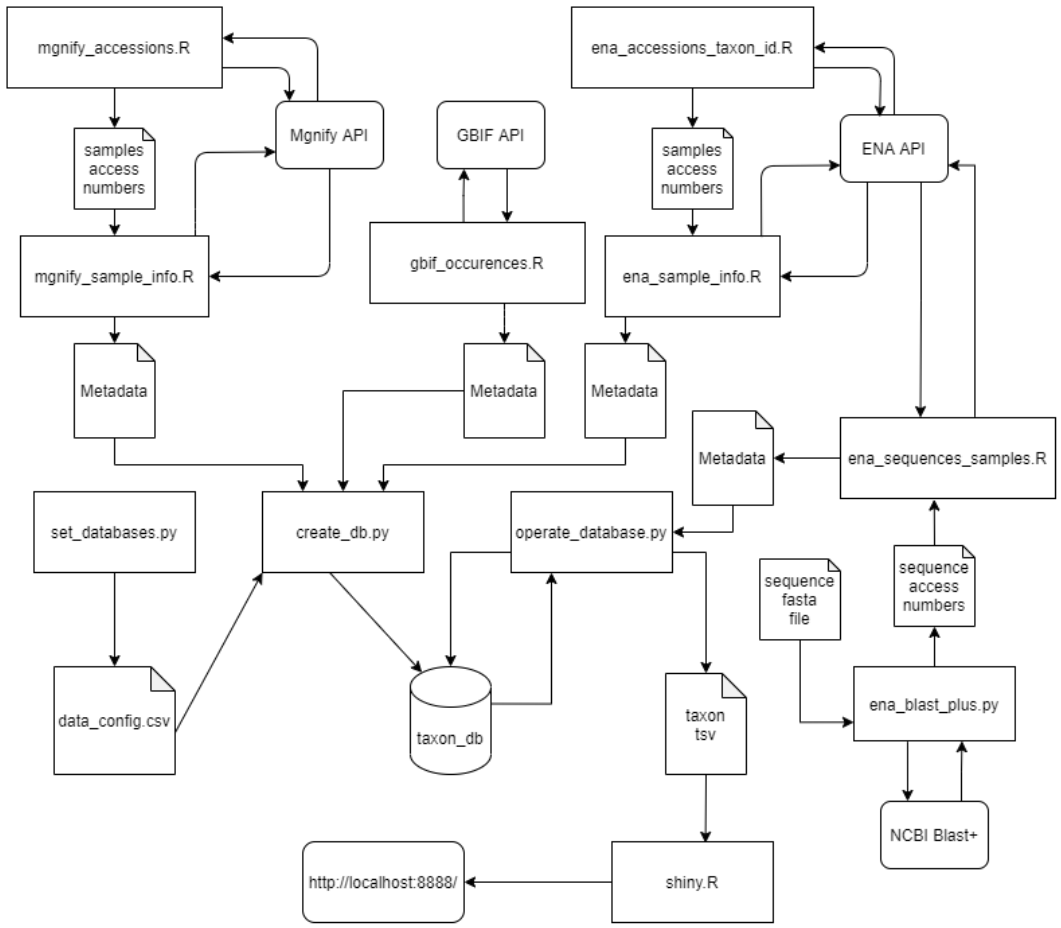

Figure 3.2: Architecture of the tools

The first set of scripts is responsible for requesting and retrieving information from the platforms. There are two scripts responsible for retrieving information from ENA's Portal API: *ena_accessions_taxon_id.R*, and *ena_sample_info.R* (Figure 3.3). The first script is responsible for requesting all the accession numbers of samples that are denoted as containing the provided taxon. The output is a TSV file containing all the accessions that fit the previous query. The second script requests metadata regarding the samples registered in the aforementioned file and outputs a table with the compilation of the metadata regarding each sample to another TSV file that will be picked up in the next phase of the tool's cycle.

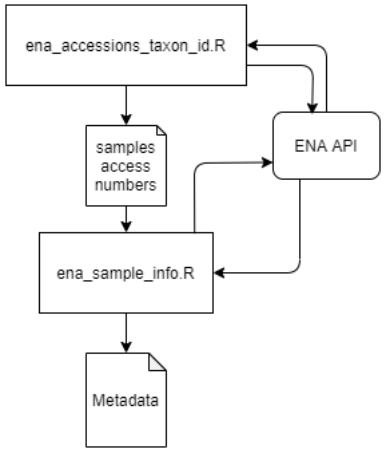

Figure 3.3: ENA Portal's API scripts for retrieving metadata related to samples

There are also two scripts that interact with MGnify's API (Fig.3.4) and have analogous functions to the first two scripts: *mgnify_accessions.R*, and *mgnify_sample_info.R*. The first one performs a full-text search in the MGnify's database to obtain the accession numbers of the samples tagged with the species being searched. The output is a TSV file containing those accession numbers that are picked up by the second script to retrieve the metadata regarding each entry in that TSV file. Similarly to the previous scripts, this one will also store the information in a table and then convert it to a TSV file.

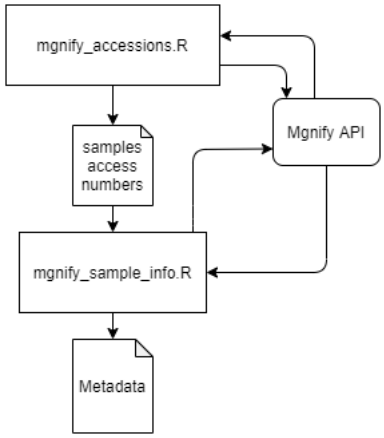

Figure 3.4: MGnify's API scripts for retrieving metadata related to samples

GBIF has a dedicated script (Fig.3.5) to gather information about occurrences of a species in their platform that are attributed to a species, *gbif_occurences.R*. It stores the information in the same way as the previous scripts, outputting a TSV file that will be picked up by the next set of scripts.

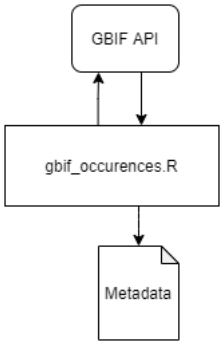

Figure 3.5: GBIF's API script for retrieving metadata related to occurrences

We have also implemented two scripts to find DNA sequences similar to the reference sequence of a species (Figure 3.6): *ena_blast_plus.py*, and *ena_sequences_samples.R*. The reference sequence must be supplied so that the scripts can work. The first script will submit a job using the EMBL-EBI NCBI Blast+ Nucleotide Similarity Search tool. The second script will return the metadata of each match returned by the first. The metadata is retrieved using ENA Portal's API.This tool has a comparatively low maximum number of one thousand (1000) results compared to the previous methods.

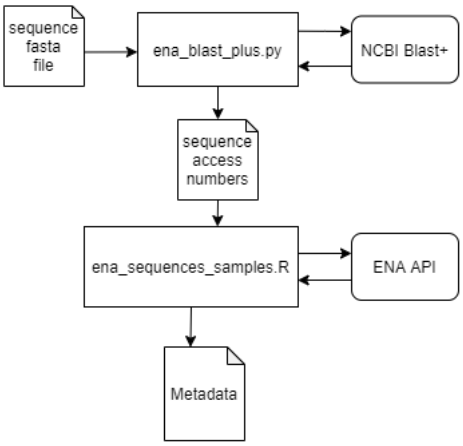

Figure 3.6: Scripts that retrieve metadata regarding the macthes from NCBI Blast+

The second set of scripts processes the TSV files generated by the first set and proceeds to create and operate a database for each species. Each public database requires a different approach to collect data and provide non-identical outputs. To make the data uniform and gather the results into a single database per taxon, the scripts responsible for this operation must be run with the assumption that the proper input files exist. These files contain information generated by the previous scripts. They are required to build the database for each species.

This set of scripts was coded in python in order to better interact with the database. The bsddb3.db python module was chosen to manage the database environment. This module attempts to mimic the Berkeley DB API and it is suggested to consult Oracle's documentation [2] to clarify any doubts during implementation. This set is divided into three parts: set up, creation, and management. Each part is implemented in a different script, *set_databases.py*, *create_db.py*, and *operate_database.py*, respectively. The first phase objective is to create the guidelines for the creation of one or more databases according to the user; determining which files will be used to create each taxon database. This step relies on the user to have used the previous tools to search for the same species and to not mix results from several taxa. Then the creation will be based on those instructions that will be left on a file in the databases directory by the previous script. Lastly, the databases can be managed by a third script that will also perform a translation operation so that the information is readable to the web app that will display it in a visual manner.

The web app is built using Shiny from R Studio [20], implemented in the script *shiny.R*. The web app has three main features. The first one is an interactive map containing the collection location and date of the search hits generated by the scripts mentioned above organised by taxon. There is also a table containing the totality of the hits because there are some that can not be represented in a map within a time frame because their metadata is missing.. Finally, there is a set of graphics that show statistics regarding the data collected.

3.1.2 Information Retrieval

3.1.2.1 European Nucleotide Archive

ENA provides an API that allows performing various searches against all the data existing on that platform [7]. R libraries *httr* and *jsonlite* facilitate the creation of requests that are in compliance with the requirements of the API. There are many types of requests provided. The ones that are more appealing are the POST and GET */search* methods that implement a warehouse search with different query options. The latter was the one integrated with MetagenClues and is used in two different ways for retrieving samples information. Both of them were integrated using the R language.

The first type of request is to obtain the accession numbers of the samples that contain or are identified with the taxon searched for. The GET */search* request provides a query option that allows to filter the results. There is a function explicitly implemented to obtain only the records that match the given desired NCBI taxonomy identifier. As described in the official documentation [7], that function is **tax_eq** and receives a number that corresponds to the NCBI taxonomy identifier desired. There is a default maximum of one hundred thousand (100,000) matches per search, and it is necessary to change the value of the parameter *limit* to zero to obtain all of the results instead. The GET */search* request can be used to find many different types of results through the various databases of ENA. The result type field must not be empty to make a valid request. As such, of the available results, *sample* is the straightforward option. The results can be ordered in an ascendant or descendant way. The final request URL will be formatted like the one in Figure 3.7

that retrieves the accession numbers of samples of *Giardia Lamblia* (NCBI taxonomy identifier 5471). The results of that request are then processed and stored in a TSV file that contains all the accession numbers present in the API response in a one-column table with a header.

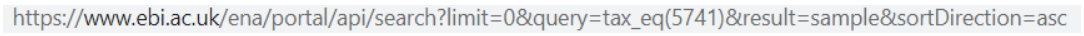

Figure 3.7: Example of a request to ENA's Portal API

The second type of request is responsible for retrieving the metadata of samples present in ENA databases through their accession numbers. These numbers are provided by means of a TSV file in the format of the one produced by the previous script but not necessarily a product of that script. Because there are two different scripts performing different functions, there is space for more versatility, as opposed to having one big script executing everything. For this operation, there are some parameters from the GET */search* request that are useful. Firstly it is necessary to provide the accession numbers in the request to the API. There is a constraint to how many numbers can be provided by filling the **includeAccessions** parameter. Only five hundred (500) can be required at a time. Therefore, it is necessary to make multiple requests and to join the metadata collected after that. By making use of the **fields** parameter, it is possible to retrieve relevant metadata such as the geographic coordinates where the sample was taken, the collection date, and the country. Unfortunately, these fields may be without value because that responsibility is entrusted to the submitters of data regarding the samples. There is not much the platform can do to enforce data submission guidelines without blocking the researchers from submitting data with incomplete metadata and that would result in a decrease of submissions. The output of this operation is a TSV file containing a table with columns for each field required to the API and the results of the search for each accession number provided.

3.1.2.2 MGnify

Similarly to the pipeline described in the previous section, there are two scripts that perform analogous operations to retrieve information using MGnify's RESTful API. However, to collect data from the API, it was necessary to have a different approach. To obtain the taxonomic constitution of a sample, possessing the accession number of it was required. Therefore, it was not possible to get samples accession numbers directly based on taxonomy identifiers or taxon names. The MGnify's API offers multiple search alternatives, but the only one that is suited for a search similar to the one done using ENA's Portal API was a full-text search. This kind of search compares a given expression with every field value of every sample in their database, which indirectly would cover the taxonomy related fields. Contrary to ENA's Portal API, the results can not be obtained in a single request because the search returns the matches paginated with a maximum of one hundred (100) hits per page. Fortunately, each response from the API contains in its body the URL to perform a GET request for the next page until there are no more results to present, thus facilitating the process of chaining multiple requests to get all the accession numbers. The GET request is

made using the */samples* endpoint of the API. In the response for each accession number in the MGnify database, it is also returned the corespondent ENA accession number that can be used to correct missing data and avoid redundancy. For example, to search for samples related to *Giardia Lamblia*, the resulting request would be the one seen in Figure 3.8. The final set of results is then stored in a two-column table transcribed into a TSV file containing for each entry the MGnify accession number and the respective ENA counterpart.

```
https://www.ebi.ac.uk/metagenomics/api/v1/samples?page=1&page_size=100&search=giardia+lamblia
```

Figure 3.8: Example of a request to MGnify's API

For the second script to operate, it is necessary to provide a TSV file containing a table with a row of MGnify accession numbers and a second row of the corresponding ENA accessions. The second row can be empty of values but must be present to avoid errors. To get the metadata of each of the samples in that file, it is necessary to make an individual request per accession number. The GET request URL is formatted as */samples/{accession number}/metadata*, and it is because of this format that the requests must be made individually. From the metadata,the latitude, longitude, country, and collection date are extracted. These values will be used to track the taxon on a map. The data collected is almost always complete because MGnify makes an effort to ensure that the metadata is accurate and to correct clerical errors such as mistakenly submit the latitude as longitude and vice-versa. In parallel to the previous scripts, the responses of the API are stored in a TSV in the format of a table.

3.1.2.3 Global Biodiversity Information Facility

GBIF also has a RESTful API that allows access to all the *occurrences* of a species in their database. Howbeit *occurrences* have similar metadata to the one collected from ENA and MGnify, and they are very different from samples. While samples are collected and analysed in at least a semi-controlled environment, the *occurrences* in GBIF take many forms. The most popular one is human observation, and many of them are not even specified. Therefore the number of *occurrences* far surpasses the number of samples retrieved from the previous methods. Although the samples are more reliable in nature than human observation, the GBIF sources are credible, and the information comes across as trustworthy. The end result is that the information retrieved from GBIF dwarfs the other methods. The API has some difficulty processing large amounts of results at the same time and as such, the results are required in increments of two hundred (200) matches at a time. In addition to that, the API will respond with an error message if any match with an index superior to one hundred thousand (100,000) is required. Therefore, by integrating the API, this is a limitation to retrieve all the *occurrences* in one swift operation. To retrieve the *occurrences* of a taxon is necessary to provide its name as seen in Figure 3.9. In that case, the target species is *Batrachochytrium Dendrobatidis*.

```
https://api.gbif.org/v1/occurrence/search?q=batrachochytrium%20dendrobatidis&limit=200&offset=0
```

Figure 3.9: Example of a request to GBIF's API

3.1.2.4 NCBI Blast+ Nucleotide Similarity Search

NCBI Blast+ Nucleotide Similarity Search tool is provided by the EMBL-EBI [12] and can perform a nucleotide similarity search provided a DNA or RNA sequence to compare with the sequences in ENA databases to find similar ones. Generally this process is done in order to gain more insight into an unknown sequence resulting in some analysis. Instead, it is provided with a well-known sequence, preferable with one that is marked as a reference sequence for that species, in order to find similar sequences in the ENA database that will be likely from the same species. This process is done via submission of a job that runs on the platform servers and can be tracked through a GET request that returns its status (Figure 3.10). To submit the job is necessary to perform a web scraping operation. Web scrapping simulates the interaction a human would have with the tool. Using this method, is possible to automate the process of submitting a job. The API of the tool will return the URL to a web page summarising the status of the job and presenting the results after completion. The results can be provided in various formats, including the JSON format (Figure 3.11). However, there is a maximum limit of only one thousand (1000) possible matches per job submitted. This limitation is crippling when trying to use the data to set up a possible surveillance system based on this tool. The second script will then take the accession numbers of the results to obtain the metadata about how the DNA sequences were produced.

```
https://www.ebi.ac.uk/Tools/services/rest/ncbiblast/status/<job_id>
```

Figure 3.10: NCBI Blast+ job status endpoint

```
https://www.ebi.ac.uk/Tools/services/rest/ncbiblast/result/<job_id>/json
```

Figure 3.11: NCBI Blast+ job JSON formatted results endpoint

3.1.3 Data Processing

3.1.4 Data normalisation

The information collected by these tools comes from various different places and takes different forms. Therefore, it is necessary to standardize all of the information before aggregating it. For example, the coordinates were marked with different notations and needed to be re-written to make it possible for the next component of the tool to be able to represent all of the results uniformly.

This step is performed as the final component of the scripts described in Section 3.1.2. The resulting output TSV files will be then used to build the databases for each taxon.

3.1.5 Retrieving different data

To decide which data to retrieve from each database, it is only necessary to check the documentation of that platform and, for each script, alter the fields that are being required. It is not difficult, and if there is a use for the tools that can be enhanced by obtaining different data, there should be no reason not to do this simple change.

In the script that requests data from ENA, one can add more options to the parameter *fields* by simply consulting the documentation for valid attributes downloadable by using the */doc* endpoint of the API [7]. The MGnify scripts require a little bit of fine-tuning, but by replicating the process implemented, one can add more fields by following a similar procedure of the current script Figure 3.12. The switch matches the names of the values returned by the API to truncated versions to facilitate reading the code and uniforming the data.

```
key_type = switch(attribute$attributes$key,
                  "geographic location (latitude)" = "latitude",
                  "geographic location (longitude)" = "longitude",
                  "geographic location (country and/or sea,region)" = "country",
                  "collection date" = "date")
```

Figure 3.12: Attributes extracted from MGnify

.

To alter the data received from GBIF, there is only the need to add the parameters to be processed. GBIF already sends all the fields available when using its API, so the only choice left is which information is useful or not.

3.1.6 Databases

The number of individual entries that these tools can recover from the different public platforms discussed can vary between a few thousand to hundreds of thousands. As such, it is imperative to have a database environment that can deal with the worst possible scenario. To accommodate that constraint, the Oracle Berkeley DB was chosen because it provides scalable, high-performance data management services. In order to make the tools run within the Anaconda environment following all the different software restrictions, the python library *bsddb3* was used instead of the newer *berkeleydb*. The *bsddb3* library provides a complete wrapping of the Oracle C API. The goal is to really emulate the real Berkeley DB API, and it is recommended to use the official Oracle Berkeley documentation for any difficulties during implementation. There are various possibilities when deciding which access methods: *btree*, *hash*, *recno*, and *queue*. The chosen method was *hash* using the accession number of an entry as a key. Because the primary key values are not logical

record numbers, the decision would be either *btree* or *hash* methods. The reason behind picking the latter was that it would outperform when the data set becomes very large.

There are three scripts that implement all the operations related to setting up the databases. All of them are implemented in Python because R did not have an up to date version of a library that could handle the use of Berkeley DB. The first script is the simpler of the three and will set up a CSV file that will contain the instructions to build the databases. Each line of this file, that will be named *database_config.csv*, represents one database and will contain the locations of the files necessary to build the database. As seen in Figure 3.13, the files must be provided in a specific order to make everything work. This script allows showing the current set of instructions, adding a new instruction for a new database, removing one instruction based on its index, and clearing all instructions in order to have a fresh start.

```
1,<database 1 name>, <ena_TSV_file>, <mgnify_TSV_file>, <gbif_TSV_file>
2,<database 2 name>, <ena_TSV_file>, <mgnify_TSV_file>, <gbif_TSV_file>
3,<database 3 name>, <ena_TSV_file>, <mgnify_TSV_file>, <gbif_TSV_file>
```

Figure 3.13: *database_config.csv* syntax

The second script will check the directory that it receives as an argument to look for the *database_config.csv* file. After finding the configuration file for each line, there will be a database created based on the files specified there. As stated before, the accession numbers will be used as the primary key, and all the attributes will be the data of that entry.

The third and final script implements the interactions with each database. It is responsible for an operation called *extraction* that will translate the data in the database to a TSV file so that it can be read by the last script of the entire pipeline of this toolset, the script responsible for the visual representation.

3.1.7 Visual solution

In order to present the results obtained from using the tools in a more user-friendly manner, the information is displayed using Shiny from R Studio [20]. This R package main feature is the creation of interactive web apps. The script *shiny.R* is the implementation of the web app that will display the information for the users. It is divided into two parts: "user interface", and "server". The "user interface" contains all the elements presented to the users to let them change the parameters that filter the data collected by the scripts described in Subsection 3.1.2.

3.1.7.1 Interactive map

The matches that possess both location and date attributes without empty or invalid values are marked on an interactive map (Figure 3.14). The interactive map displays the matches as circles. The circles provide information on the location and date of the occurrences as well as the platform that provided the information and the accession number (Figure 3.15). The accession number in

conjunction with the platform parameter allows the search of the full submission on the original databases. This map is built using R language Leaflet libraries [11]. The data being used of such data can be consulted by interacting with the marker, and the map can be zoomed in to understand better the disposition of the occurrences (Figure 3.16).

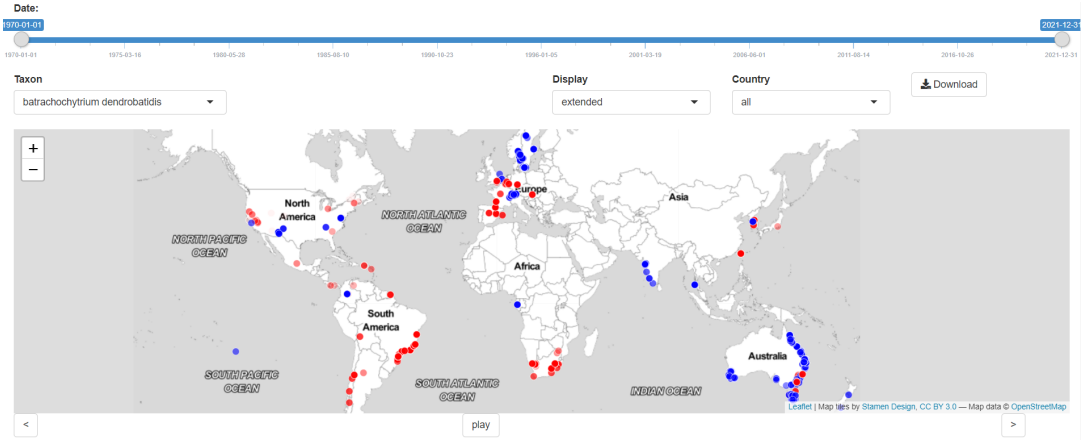

Figure 3.14: Interactive map

Figure 3.15: The information is displayed in each circle.

Figure 3.16: The interactive map zoomed in on Switzerland

.

The circles have different levels of opacity to make the more recent matches stand out over older ones since they are more likely to be more relevant. In Figure 3.18, the circle showcasing a match retrieved from ENA from 2007 is more transparent than the match from GBIF from 2012 of the same data set (relative to *batrachochytrium dendrobatidis*) in Figure 3.17. The colour of a circle indicates from which platform was the data retrieved.

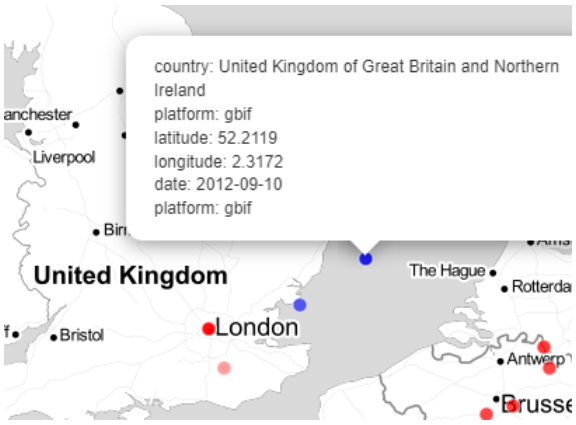

Figure 3.17: *batrachochytrium dendrobatidis* United Kingdom match in GBIF from the year 2012.

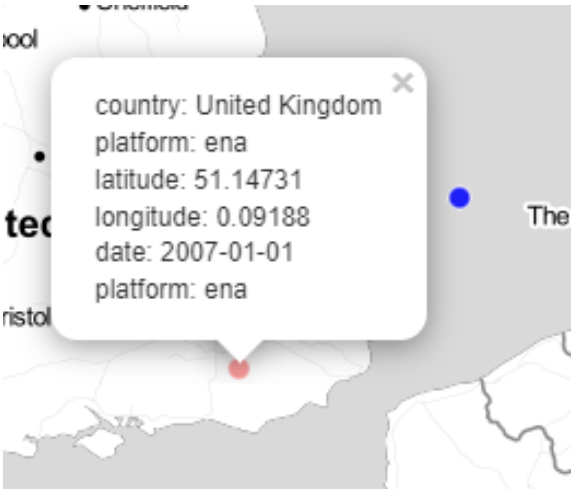

Figure 3.18: *batrachochytrium dendrobatidis* United Kingdom match in ENA from the year 2007.

There is a drop-down list right above the interactive map that allows to choose the taxon displayed in the elements of the web page (Fig. 3.19). Not only the interactive map updates to show the taxon selected in the drop-down but also the table and the graphic as well. The names of taxa that appear in the drop-down list are retrieved to the configuration list. The time frame displayed can also be customized in the interactive map. There is an adjustable slider that controls the window of time of the results shown in map as seen in the top of Figure 3.14. This feature allows for the analysis of the distribution of a taxon during specific years. Choosing the appropriate time frame is important to study species dispersion and to choose the more appropriate models to apply to make the analysis [17].

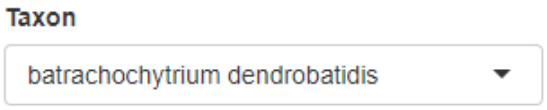

Figure 3.19: The drop-down in this figure allows to change the taxon to be displayed.

The interactive map can also be filtered to allow different visualizations (Fig. 3.20). Some species have a very high number of occurrences, in the order of tens of thousands as it was the case with the *Vespa Mandarinia*, commonly known as the Asian giant hornet (Table 3.4). This amount of data can be challenging for the shiny server to handle due to high amount of objects added to the interactive map. In order to tackle this issue there is a *display* parameter with two options: extended and aggregated. In extended mode each circle in the map shows the information collected regarding one unique occurrence (Fig. 3.18). This is the default mode and it is the one that provides more information at once with the possible disadvantage of being harder to interpret the data.

Figure 3.20: The filters *display* and *country* allow for a more narrower visualization of the data and the *Download* button allows to download a TSV file with the data shown in the map.

If the aggregated mode is the chosen option of display, each circle in the map will display different values than the extended mode. Instead of the metadata of one occurrence,the circles will present the number of results that were grouped into that circle (Fig. 3.21). The results are grouped by platform, country, and year. The coordinates of the circle in the map correspond to the median of the aggregated results. This method was chosen over performing an average operation in order to minimise the impact of faulty records, however there is still margin for error and some of the circles will be attributed to a country and be drawn outside of that country boarders. Occurrences that were retrieved from the ocean have a particularly negative effect on the accuracy of the circles if there are few results for a specific country (Fig. 3.21).

Figure 3.21: In the aggregated mode the map becomes easier to read because there are less objects draw in the map which allows a better global perspective of the distribution of the species.

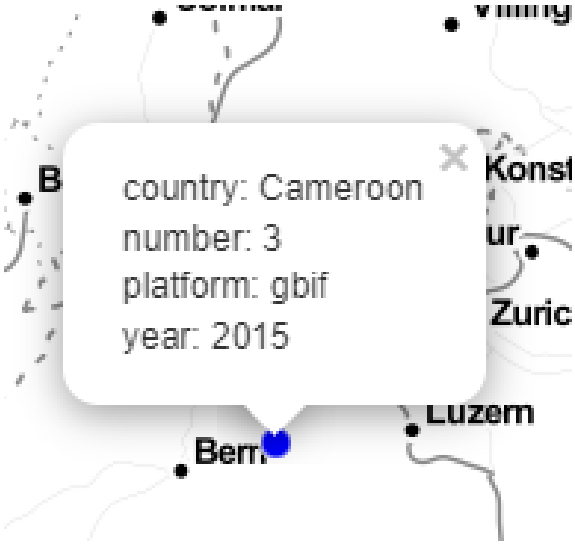

Figure 3.22: *batrachochytrium dendrobatidis* match from Cameroon with coordinates near the Switzerland capital.

The data extracted from GBIF is, in the case of some taxa, several times larger than the other platforms (Section 3.2). To further mitigate the impact of having too many circles in the map, there is also the *country* filter option. Additionally, the aggregated view will not provide the detail necessary if the accession numbers of the occurrences are necessary for the user (Fig. 3.21). In such cases, it is possible to show the records extracted from GBIF by country. There is a drop-down that contains an *all* option to show the results unfiltered by country and a list with all the possible countries registered in GBIF. That list can be extracted from the GBIF's API using a utility script (*gbif_get_countries.R*) that makes a request to the API and prints the returned list of countries.

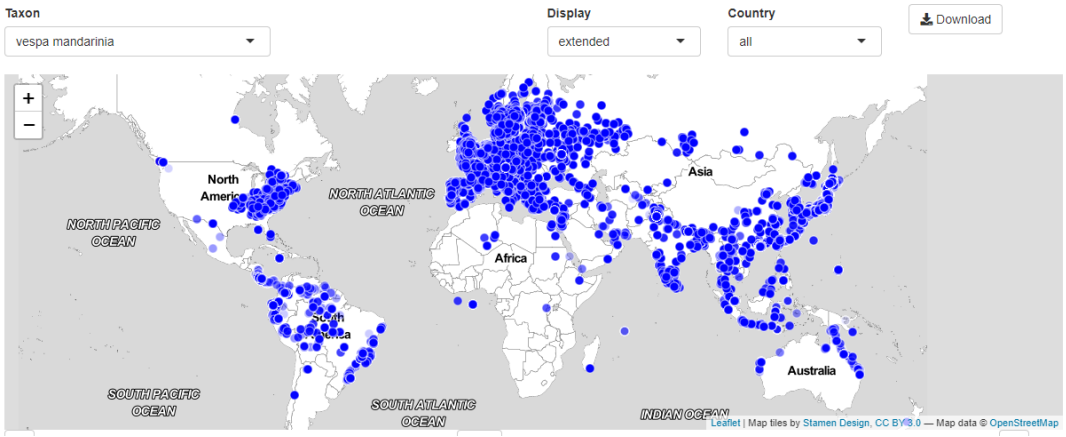

Figure 3.23: The high number of occurrences of *Vespa Mandarinia* complicates the analysis of the interactive map without zooming in and sacrificing global perspective in favor of visually understanding the data better.

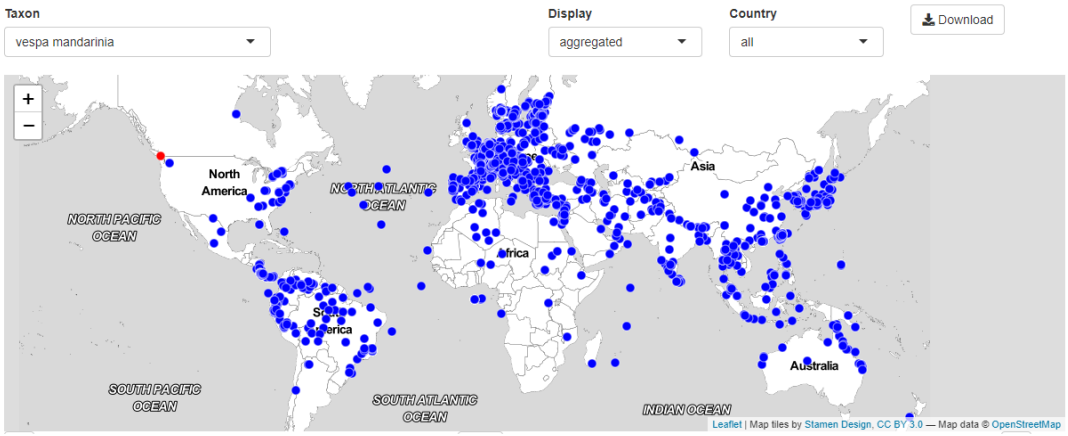

Figure 3.24: The aggregated mode provides a clearer view of the world dispersion of the *Vespa Mandarinia*

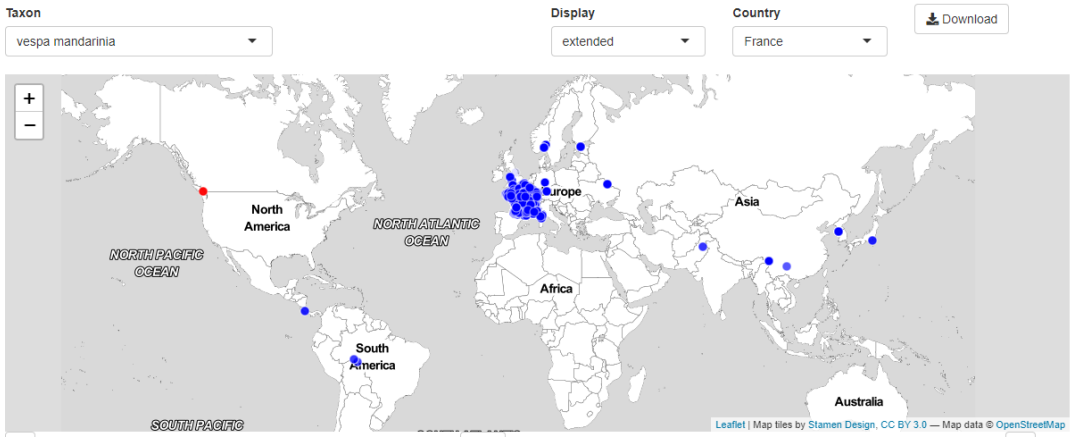

Figure 3.25: *Vespa Mandarinia* matches filtered to only show results marked as recovered from France.

The information displayed in the map can be easily downloadable via the *Download* button seen in Figure 3.20. Interacting with this button will provide a TSV file with the data displayed in the map with the current filters except for *display*. This file will always contain the extended version of the occurrences in order to provide more detail and provide the accession number of the occurrences for further follow-up in the respective platform if deemed necessary.

Another feature of the interactive map is the option of drawing the data in succession by year. This feature is controlled by the buttons bellow the map (Figures 3.26 and 3.27).The first time a taxon is loaded to the interactive map all the results will be drawn into circles and a list of what years have at least one result is created. The central button alternates between being a *play* button, to initiate the process of going throw the years and altering the map accordingly, and a *pause* button, to stop that process and return the map to a static state. When paused the map will not reset and will not change automatically until the *play* button is pressed again. In that case, the map

will resume the circle update cycle from the the latest year that is currently being drawn. This way it is not necessary to restart the whole cycle and start by the first occurrence every time the process is paused.

Figure 3.26: Before interacting with the *play* button for the the interactive map is frozen and will not change automatically.

Figure 3.27: After interacting with the *play* button the text is replaced with *pause* and the map will update automatically every five seconds.

The interactive map will be updated every 5 seconds using an ordered list of those years to determine which occurrences will be drawn as circles in the map. When a circle of a specific year is drawn the circles representing previous years will already be draw. When using the ">" and "<" buttons the interactive map will revert to its static state and will move forward, to the next year in the list, or the previous year, respectively.

Figure 3.28: *batrachochytrium dendrobatidis* matches in USA east coast from 1999 and previous years

Figure 3.29: *batrachochytrium dendrobatidis* matches in USA east coast from 2000 and previous years. There is an additional entry in the state of Maine than those from Fig. 3.28

3.1.7.2 Data table

The next component is a data table that contains every result obtained from the tools, and its main purpose is to show even the results that could not be represented on the map. There are options to customise how many entries are displayed at a time and to perform full-text searches on the data (Figure 3.30). By changing the species on the drop-down above the map, the data in the table will also change accordingly to data from that taxon.

Show 10 entries Search:

	accession.key	platform	latitude	longitude	date
1	1647212115	gbif	-41.685769	146.519369	2004-12-04
2	1647212160	gbif	-42.852959	146.202217	2006-12-03
3	1655893638	gbif	-27.45	152.98333	
4	1655894019	gbif	-28.448	153.168	
5	1655895456	gbif	-21.06222	148.63917	2006-03-20
6	1655896000	gbif	-41.44905	146.56302	2011-03-31
7	1655896383	gbif	-28.183	153.268	2011-03-31
8	1655896527	gbif	-28.225	153.223	
9	1655896725	gbif	-26.83583	152.87861	
10	1655898983	gbif	-15.70806	145.26361	

Showing 1 to 10 of 2,690 entries Previous 1 2 3 4 5 … 269 Next

Figure 3.30: Data table with all the data collected from the tools from ENA, MGnify and GBIF

The final component is a graphical representation of the distribution of the data collection that will change to represent data from the species selected on the drop-down. The graphic shows how much data was retrieved from each platform and how many entries were missing metadata.

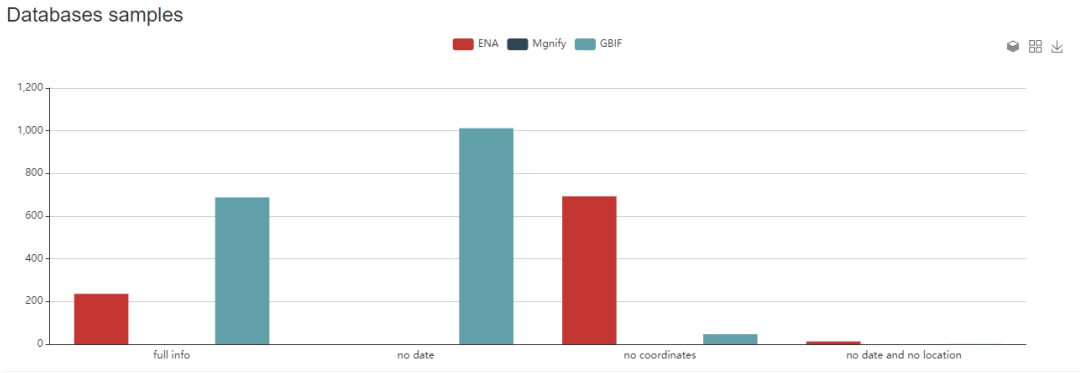

Figure 3.31: Data quantity and coordinates and date parameters quality check

3.2 Results

There were four species that were proposed to serve as a proof of concept for the utilisation of these tools: *Batrachochytrium dendrobatidis*, *Sphaerothecum destruens*, *Vespa mandarinia*, and *Giardia lamblia*. The data retrieved shows that some species have a much bigger presence in some databases than others. The results from using NCBI Blast+ were parallel to what is seen in this table. Using a reference DNA sequence for *Sphaerothecum destruens* only one match was returned while in contrast, using a DNA sequence for *Vespa Mandarinia* the maximum amount of one thousand results were returned.

Species	**ENA**			
	No Date	**No Coordinates**	**No Date & Coordinates**	**Full info**
Batrachochytrium dendrobatidis	0	693	13	236
Sphaerothecum destruens	0	1	0	0
Giardia Lamblia	0	12	187	91
Vespa Mandarinia	0	4	1	1

Table 3.1: Data retrieved from ENA

.

The ENA platform returned fewer results than expected for *Vespa Mandarinia* and most of the results did not have coordinates of the local where the sample was taken. This results show that there is still some growth needed in the metagenomics databases in order to solely use the data extracted from those to surveil species. However, *Batrachochytrium dendrobatidis* data retrieved from ENA has a comparable number of matches to the data retrieved from GBIF (Table 3.4). This was the only taxon that had this proportion of metagenomic entries and occurrences. The fact that this specific pathogenic fungus is considered responsible for the decline and extinction of some amphibian species [9] can contribute to being more closely monitored, while the fact that is pathogenic fungus drastically diminishes occurrences based on human direct observation.

Taxon	MGnify using NCBI Taxonomy current names			
	No Date	No Coordinates	No Date & Coordinates	Full info
Batrachochytrium dendrobatidis	0	0	0	0
Sphaerothecum destruens	0	0	0	0
Giardia Lamblia	0	0	0	0
Vespa Mandarinia	0	0	0	0

Table 3.2: Data retrieved from MGnify

.

Taxon	MGnify using common names			
	No Date	No Coordinates	No Date & Coordinates	Full info
Batrachochytrium dendrobatidis	0	0	0	0
Sphaerothecum destruens	0	0	0	0
Giardia Lamblia	61	0	0	0
Vespa Mandarinia	0	0	0	0

Table 3.3: Data retrieved from MGnify

.

The results from MGnify were not according to what was expected. The chosen species had no matches using the NCBI taxonomy current names (Table 3.2). Although NCBI states that their nomenclature is not the general norm, both GBIF and ENA used it. Using the common name of the species, "giardia" for *giardia lamblia* and "asian giant hornet" for *vespa mandarinia*, only a few matches were found for *giardia lamblia* (Table 3.3). Those matches had both coordinates and date of sample collection. The preference of the MGnify platform for data derived from sequencing microbial populations plays a role on some species having no matches while others like *Homo Sapiens* have about one hundred and thirty thousand (130,000) samples present on the platform. The scripts were able to retrieve those entries and display the metadata correctly. In the future, possibly by studding a different approach to the MGnify's API or by the enlargement of their database through new submissions, it will be possible to retrieve more data from this platform.

Taxon	GBIF			
	No Date	No Coordinates	No Date & Coordinates	Full info
Batrachochytrium dendrobatidis	1012	47	1	688
Sphaerothecum destruens	1	144	27	1225
Giardia Lamblia	38	2187	306	31486
Vespa Mandarinia	55	2480	664	94754+

Table 3.4: Data retrieved from GBIF

.

GBIF was by an extensive amount the biggest contributor of data (Table 3.4). The number of matches returned by the API of this platform is explainable by the nature of the occurrences.

While in ENA and MGnify the occurrences represent samples submitted after analysis in GBIF, the occurrences can be as simple as a human observation. Although DNA sequencing techniques have become much faster and cheaper, the price is still much higher when compared to direct human observation. While this disparity remains so large, it seems that using the metagenomic databases as a surveillance system without complementing with data retrieved by researchers actively seeking out potentially dangerous species will be inefficient. However, the tools will follow the growth of DNA sequencing technologies and breakthroughs.

Chapter 4

Applications

4.1 Tracking the invasive Asian Giant Hornet

Vespa Mandarinia, commonly known as "asian giant hornet" is the world largest hornet. It is a very predatory and invasive species that impacts humans and is notably destructive to other species that are beneficial for the humans like the bee [21]. The affected individuals can undergo severe allergic reactions which can even cause death in cases of extreme sensitivity and in case of receiving multiple stings, an attack by asian giant hornets can leave to long-term health effects [18]. The East coast of the United States of America, West Europe, and the south of Brazil are considered the regions that have very high climate suitability for this species combined with increasing severity of invasions due to human activity [21].

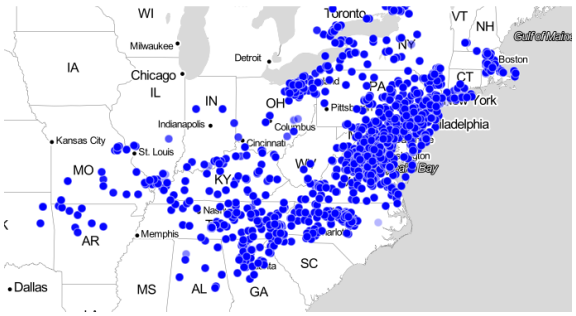

Figure 4.1: In the East coast of the United States of America there are plenty of results with high opacity, indicating that the results are recent.

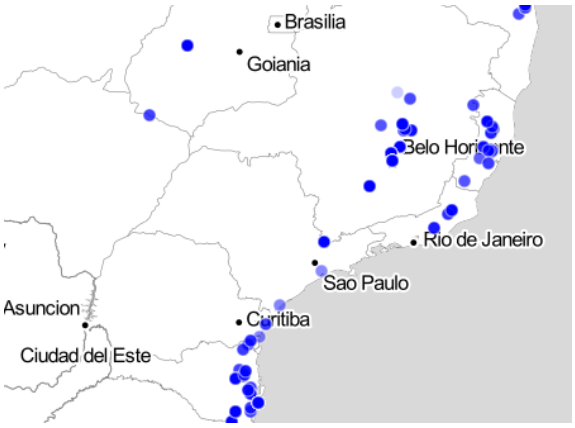

Figure 4.2: Although the south of Brazil has fewer matches than the East coast of the United States of America, the matches there are also recent.

Figure 4.3: Europe has many recent occurrences of *Vespa Mandarinia* which serves as a testimonial of the proliferation of this invasive species.

The matches retrieved are in accordance with the zones that are more climate compatible with the *Vespa Mandarinia*, with the exception of the south of Brazil that theoretically should have have more hits.

4.2 Possibility of prevention

Invasive species like *Vespa Mandarinia* require close monitoring to protect not only the ecosystems that are being invaded but also the medical and economic interests of the local population. The feature that automatically updates the map establishes a pattern of growth first in Japan and Asia, then in Europe, and after that in the Americas (Figures 4.4 to 4.9).

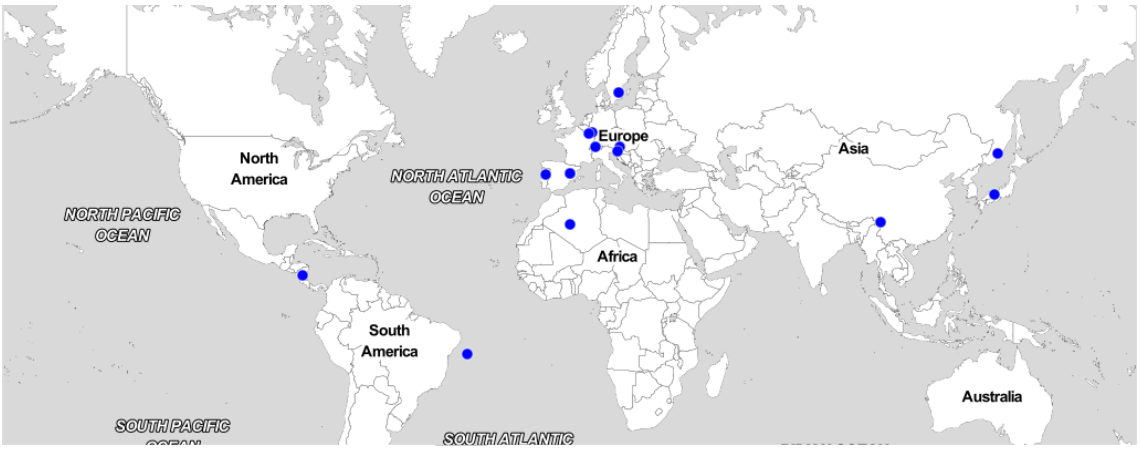

Figure 4.4: The first matches of *Vespa Mandarinia* are dated to 1970 and are seen in this figure.

In the following years there is an increase of occurrences reported in Japan in particularly, with eighty-three occurrences in the year of 1973. After that, unique occurrences start being recorded in Europe, Asia, Indonesia, Africa, and South America. However, at this point in time there is at most three records per year.

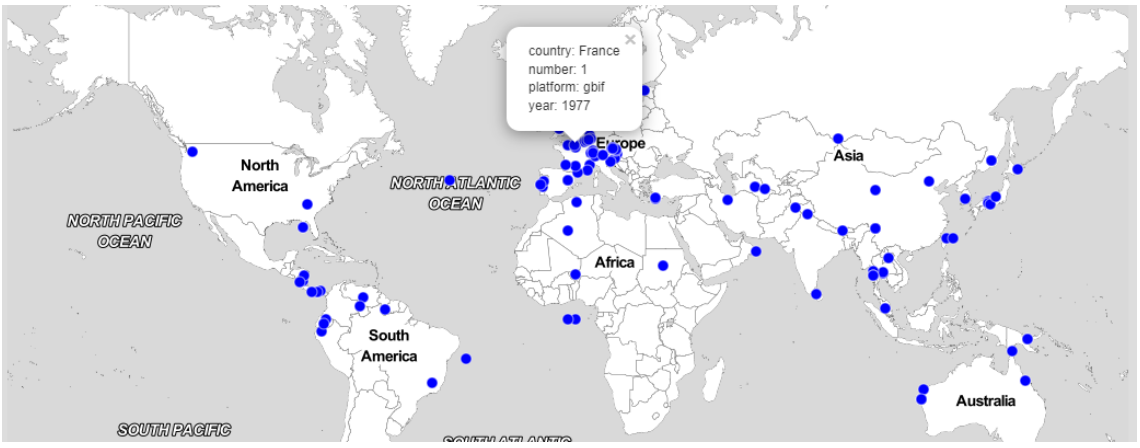

Figure 4.5: *Vespa Mandarinia* aggregated matches from 1977 and previous years.

In Europe *Vespa Mandarinia* the number of occurrences slightly increases but still maintaining a low average of occurrences per year.

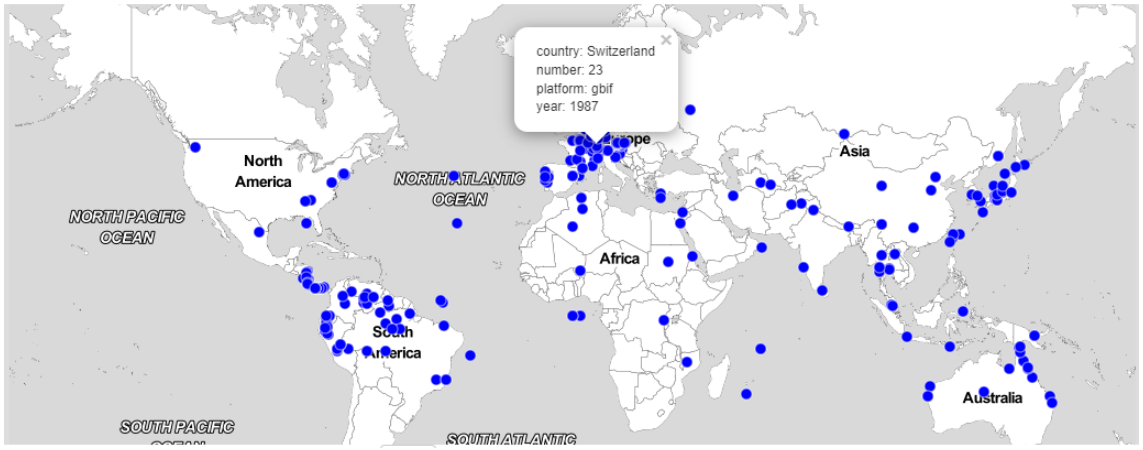

Figure 4.6: *Vespa Mandarinia* aggregated matches from 1987 and previous years.

The number of registered occurrences in Europe starts to increase more significantly to the order of a couple dozens instead of single digits.

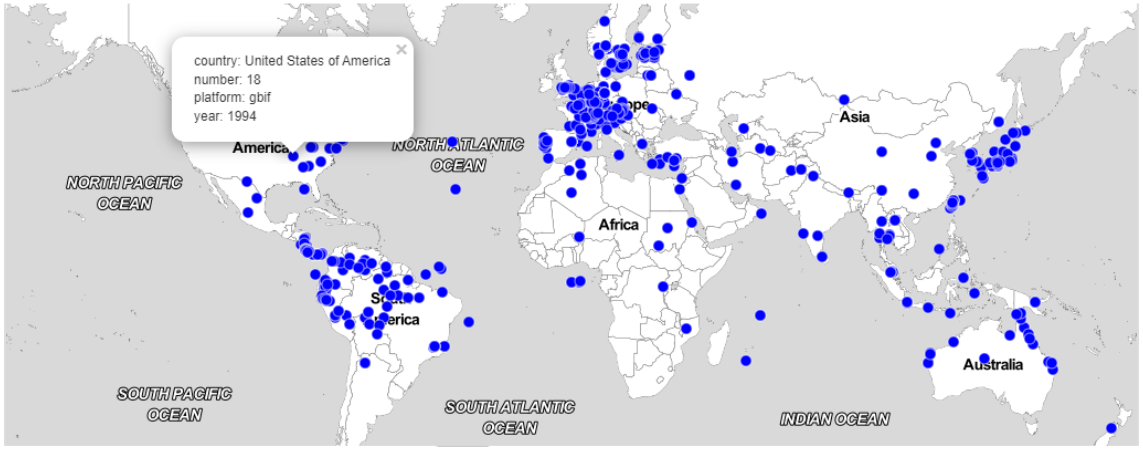

Figure 4.7: In 1994 the *Vespa Mandarinia* seems to have established a foothold in Europe.

Around 1994, besides the ever increasing number of occurrences in Europe, in the United States of America the *Vespa Mandarinia* occurrences start to increase similarly to the increase in Europe in the previous years. Instead of having one to three occurrences per year, in 1994 eighteen sightings were reported.

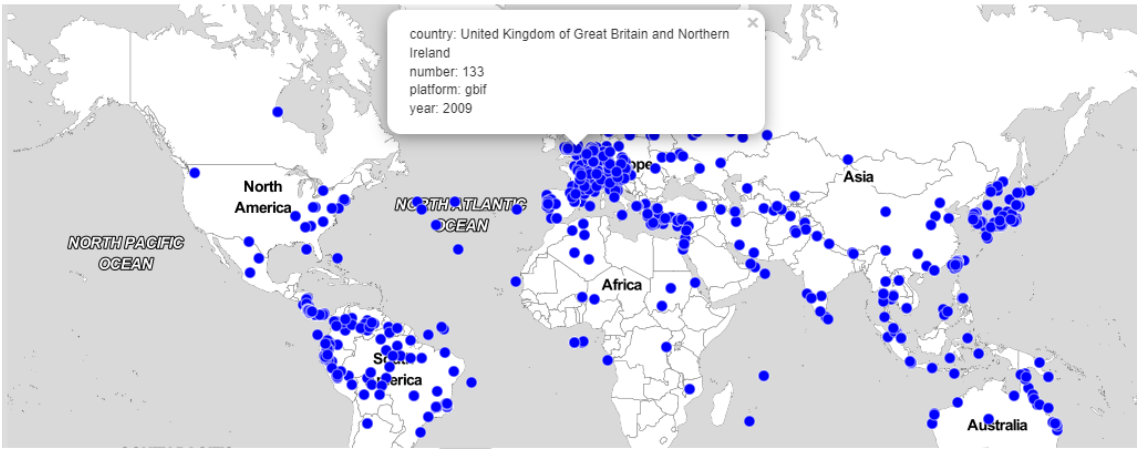

Figure 4.8: In 2009 there are a lot more occurrences in Europe.

The numbers in Europe start to increase a lot and North America seems to follow in the same behaviour.

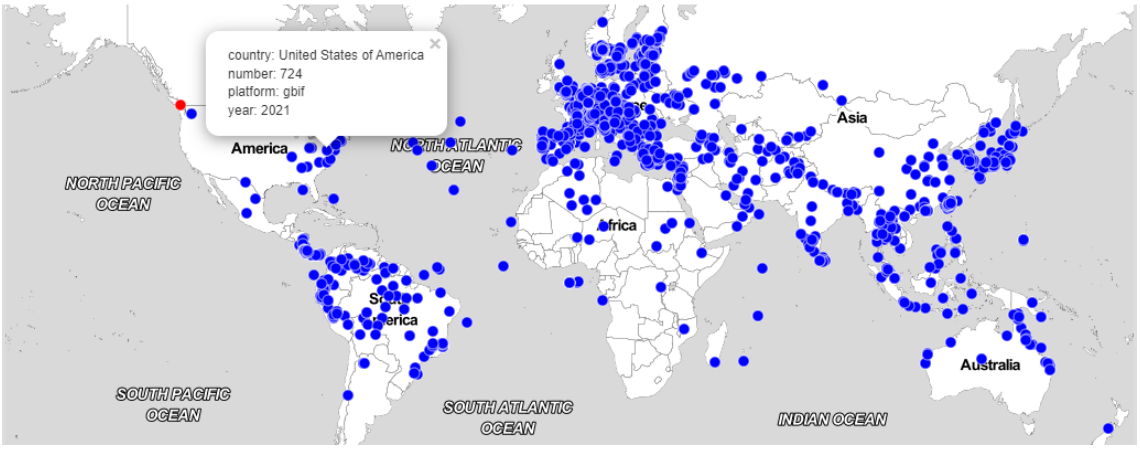

Figure 4.9: In the couple of years leading to 2021 the occurrences of *Vespa Mandarinia* in Europa have escalated to hundreds of sightings per country and North America has a surge of sightings as well.

Despite the global situation of health crisis between 2019 and 2021, there was still great concern over the spread of this invasive species and a lot of monitoring programs were created [21]. The visualization of the data extracted regarding the *Vespa Mandarinia* suggests that the East coast of North America, being a similar region to West Europe in terms of climate suitability for the species, followed a similar path than the other side of the Atlantic, mimicking the variations in Europe during previous years. However it is natural for the human brain to find relations of causality where they do not exist. Therefore it is necessary to be prepared for the fact that coincidences may happen, such has an increase in researchers interested in *Vespa Mandarinia* that actively studied and searched specimens with more proficiency and thus, increasing the number of

sightings disproportionately. The more data collected and the wider the scope of analysis the less impact will random events affect the consistency of the conclusions taken.

Chapter 5

Conclusions

The natural world and the imbalances caused by humanity have created many situations where a particular species becomes harmful to our society or to the environment where it is inserted and the other species that co-habit that environment. It is necessary to survey these situations to prevent severe damages to the health and economic status of the populations affected and prevent lasting damages in the biosphere. DNA sequencing has been developing fast and has become a lot more affordable, accurate, and fast in the past two decades. This significantly increased the quantity of data available. Specifically, more sequences were studied, and more samples were sequenced. This increase puts public databases as a source of knowledge that can be used to track the presence of species in samples and the place where those samples were taken. However it is clear that it will still take a little while to have enough data to compete with human direct observation of species. This method is obviously simpler and produces many more accounts than analysing samples and attempting to cross-reference different studies. The tools developed in this thesis address a necessary but bothersome part of any study. Using the tools is possible to gather the data from platforms like GBIF, and automatically remove database entries with empty or incomplete metadata. This process is part of cleaning the data so it becomes usable and by using the tools, the researchers can save their time and speed up the data preparation process. With this thesis the ground work for the viable utilization of these tools was laid out. It is necessary to continue to act upon it to be able to achieve it in the future, but with more data it will definitely be possible to spot patterns in the occurrences of species and passively surveil potentially dangerous species with the help of metagenomic databases.

5.1 Future work

- Study the MGnify DNA sequence analysis in more detail to ascertain the missing piece that is blockading the effective use of this platform. If possible, the work should be continued by developers with or partnering with people with background and experience submitting and performing DNA sequencing.

- The shiny app presents the results satisfactorily. However, it becomes clear that if the the data volume increases too much, R is not capable of keeping up with visual representation on a map as it stands and the web app is slowed down. Therefore, it is suggested to migrate to another language or alternatively explore new packages that might emerge in the future for R that address this issue.
- The shiny app look can be improved to make it more aesthetically pleasing. The functionalities of the app are important, but the visual aspect must be continuously improved in order to better the user experience.
- The platforms studied present a lot of possibilities for the creation of more tools.

References

[1] C. Amid, B. T. F. Alako, V. Balavenkataraman Kadhirvelu, T. Burdett, J. Burgin, J. Fan, P. W. Harrison, S. Holt, A. Hussein, E. Ivanov, S. Jayathilaka, S. Kay, T. Keane, R. Leinonen, X. Liu, J. Martinez-Villacorta, A. Milano, A. Pakseresht, N. Rahman, J. Rajan, K. Reddy, E. Richards, D. Smirnov, A. Sokolov, S. Vijayaraja, and G. Cochrane. The European Nucleotide Archive in 2019. *Nucleic Acids Research*, 48(D1):D70–D76, January 2020. Publisher: Oxford Academic.

[2] Berkeley DB Programmer's Reference Guide. https://docs.oracle.com/cd/E17276_01/html/programmer_reference/index.html, 2021.

[3] Florian P Breitwieser, Jennifer Lu, and Steven L Salzberg. A review of methods and databases for metagenomic classification and assembly. *Briefings in Bioinformatics*, 20(4):1125–1136, September 2017.

[4] Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L. Madden. BLAST+: architecture and applications. *BMC Bioinformatics*, 10(1):421, December 2009.

[5] Alycia W. Crall, Gregory J. Newman, Catherine S. Jarnevich, Thomas J. Stohlgren, Donald M. Waller, and Jim Graham. Improving and integrating data on invasive species collected by citizen scientists. *Biological Invasions*, 12(10):3419–3428, October 2010.

[6] Maria Csuros. *Environmental Sampling and Analysis for Technicians*. CRC Press, Boca Raton, October 2017.

[7] Ena portal api. https://www.ebi.ac.uk/ena/portal/api/, 2021.

[8] Scott Federhen. The NCBI Taxonomy database. *Nucleic Acids Research*, 40(Database issue):D136–D143, January 2012.

[9] Trenton W.J Garner, Matthew W Perkins, Purnima Govindarajulu, Daniele Seglie, Susan Walker, Andrew A Cunningham, and Matthew C Fisher. The emerging amphibian pathogen Batrachochytrium dendrobatidis globally infects introduced populations of the North American bullfrog, Rana catesbeiana. *Biology Letters*, 2(3):455–459, September 2006.

[10] GBIF.org. Gbif home page. https://www.gbif.org, 2021.

[11] Christian Graul. leafletr: Interactive web-maps based on the leaflet javascript library, 2016. R package version 0.4-0.

[12] Fábio Madeira, Young Mi Park, Joon Lee, Nicola Buso, Tamer Gur, Nandana Madhusoodanan, Prasad Basutkar, Adrian R N Tivey, Simon C Potter, Robert D Finn, and Rodrigo Lopez. The EMBL-EBI search and sequence analysis tools APIs in 2019. *Nucleic acids research*, 47(W1):W636–W641, July 2019.

[13] A. L. Mitchell, A. Almeida, M. Beracochea, M. Boland, J. Burgin, G. Cochrane, M. R. Crusoe, V. Kale, S. C. Potter, L. J. Richardson, E. Sakharova, M. Scheremetjew, A. Korobeynikov, A. Shlemov, O. Kunyavskaya, A. Lapidus, and R. D. Finn. MGnify: the microbiome analysis resource in 2020. *Nucleic Acids Research*, 48(D1):D570–D578, January 2020. Publisher: Oxford Academic.

[14] Takeru Nakazato, Tazro Ohta, and Hidemasa Bono. Experimental Design-Based Functional Mining and Characterization of High-Throughput Sequencing Data in the Sequence Read Archive. *PLOS ONE*, 8(10):e77910, October 2013. Publisher: Public Library of Science.

[15] Jan Pergl, Petr Pyšek, Franz Essl, Jonathan M. Jeschke, Franck Courchamp, Juergen Geist, Martin Hejda, Ingo Kowarik, Aileen Mill, Camille Musseau, Pavel Pipek, Wolf-Christian Saul, Menja von Schmalensee, and David Strayer. Need for routine tracking of biological invasions. *Conservation Biology*, 34(5):1311–1314, 2020. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cobi.13445.

[16] S. Ratnasingham and P. D. N. Hebert. bold: The Barcode of Life Data System (http://www.barcodinglife.org). *Molecular Ecology Notes*, 7(3):355–364, 2007. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1471-8286.2007.01678.x.

[17] Samuel M. Scheiner, Alessandro Chiarucci, Gordon A. Fox, Matthew R. Helmus, Daniel J. McGlinn, and Michael R. Willig. The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81(2):195–213, 2011. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1890/10-1426.1.

[18] Justin O. Schmidt, Soichi Yamane, Makoto Matsuura, and Christopher K. Starr. Hornet venoms: Lethalities and lethal capacities. *Toxicon*, 24(9):950–954, January 1986.

[19] Jay Shendure, Shankar Balasubramanian, George M. Church, Walter Gilbert, Jane Rogers, Jeffery A. Schloss, and Robert H. Waterston. DNA sequencing at 40: past, present and future. *Nature*, 550(7676):345–353, October 2017. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 7676 Primary_atype: Reviews Publisher: Nature Publishing Group Subject_term: Genetics;Sequencing Subject_term_id: genetics;sequencing.

[20] Shiny. https://shiny.rstudio.com/, 2022.

[21] Gengping Zhu, Javier Gutierrez Illan, Chris Looney, and David W. Crowder. Assessing the ecological niche and invasion potential of the Asian giant hornet. *Proceedings of the National Academy of Sciences*, 117(40):24646–24648, October 2020. Publisher: National Academy of Sciences Section: Biological Sciences.