Reference Sequence Browser: An R application with a User-Friendly GUI to rapidly query sequence databases

Samuel Rapp [1 ‡], Jorge Tapias Gomez[2 * ‡], Sriram Ramesh[1 ‡], Benjamin Levine[1], Daniel Tapias-Gomez[3], Dickson Chung[1], Zia Truong[1]

1 University of California Santa Cruz, Santa Cruz, CA, 95064, USA
2 Cornell University, Ithaca, NY, 14850, USA
3 University of Texas Southwestern, TX, 75390, USA

‡These authors contributed equally to this work.
* jt787@cornell.edu

Abstract

Land managers, researchers, and regulators increasingly utilize environmental DNA (eDNA) techniques to monitor species richness, presence, and absence. In order to properly develop a biological assay for eDNA metabarcoding or quantitative PCR, scientists must be able to find not only reference sequences (previously identified sequences in a genomics database) that match their target taxa but also reference sequences that match non-target taxa. Determining which taxa have publicly available sequences in a time-efficient and accurate manner currently requires computational skills to search, manipulate, and parse multiple unconnected DNA sequence databases.

Our team iteratively designed a Graphic User Interface (GUI) Shiny application called the Reference Sequence Browser (RSB) that provides users efficient and intuitive access to multiple genetic databases regardless of computer programming expertise. The application returns the number of publicly accessible barcode markers per organism in the NCBI Nucleotide, BOLD, or CALeDNA CRUX Metabarcoding Reference Databases. Depending on the database, we offer various search filters such as min and max sequence length or country of origin. Users can then download the FASTA/GenBank files from the RSB web tool, view statistics about the data, and explore results to determine details about the availability or absence of reference sequences.

Introduction

Environmental DNA (eDNA) is an emerging field that has helped answer questions in several disciplines, including molecular ecology, environmental sciences, conservation biology, and paleontology [1, 2]. eDNA techniques are advantageous to traditional monitoring due to their non-invasive nature, ability to monitor aquatic communities, and the relative ease of training field workers.

Environmental DNA workflows begin with acquiring samples from the environment under study so that genetic material can be extracted and then analyzed with various forms of Polymerase Chain Reaction (PCR) assays [2]. Depending on the specific goals of the study, eDNA primers are designed at varying levels of taxonomic inclusivity, often using either taxa-specific quantitative PCR (qPCR) or DNA metabarcoding PCR. Taxa-specific qPCR uses eDNA to detect a single targeted taxon, while eDNA metabarcoding aims to assess taxa richness.

Both qPCR assay design and taxonomic assignment in metabarcoding depend heavily on the "availability of DNA reference sequences in public data facilities (e.g., National Center for Biotechnical Information (NCBI), The Barcoding Of Life Data System (BOLD))" [3]. Reference sequences are pre-labeled sequences of DNA that allow researchers to either identify the unlabeled DNA they collect during their studies or design new species-specific primers for qPCR [4]. However, high-quality reference sequences are unavailable for many organisms, which has become a key factor limiting the broad application of eDNA techniques [5]. As such, screening genomic databases (e.g., NCBI Nucleotide and BOLD) or personalized reference sequence databases such as CRUX (Creating Reference Libraries Using eXisting tools) [6] for reference sequence availability and gaps is essential.

Additionally, detecting the absence of reference sequences is not the only challenge that eDNA scientists face. Through our own experience and interviews with over 70 scientists (see Acknowledgements), we found that downloading the potentially thousands of sequences needed to create personalized reference databases is tedious for both new and experienced labs. Using the official websites for sequence databases, like NCBI Nucleotide or BOLD, to manually search for and download quality sequences can take days. Some researchers avoid this time-consuming process by writing scripts, but not all researchers possess those skills.

Completing both tasks in a time-efficient and accurate manner requires computational skills to search, manipulate, and parse multiple DNA sequence databases. Ecologists, however, often need more training in this area. For example, in California, most (74 - 80%) University of California and California State University undergraduates in ecological and environmental sciences had no formal training in programming skills in any language [7]. Bioinformatic assessments of large genetic databases can be challenging and time-consuming for classically trained ecologists and may result in bioinformatic work becoming a study expense and delaying the project.

To address these issues, we built the Reference Sequence Browser (RSB), a Graphical User Interface (GUI) tool that allows researchers to perform customized batch searches and downloads for reference sequences on the following publicly available databases: NCBI nucleotide, BOLD, and CALeDNA CRUX databases for metabarcoding [6, 8, 9]. Those interested in a simple but powerful way of viewing and downloading reference sequences from multiple genomics databases for one or multiple organisms and barcoding loci simultaneously will find that this tool saves time and provides insight into reference sequence availability and gaps. With the RSB, researchers can efficiently develop qPCR assays, determine the efficacy of particular barcoding markers to match an eDNA project's objective, and engage in more deliberate and specific sequencing efforts to address gaps in reference sequence availability.

Materials and Methods

Installation

The RSB is a Shiny GUI app [10], which is available online at `https://sriramramesh.shinyapps.io/ReferenceSequenceBrowser/` or for download at `https://github.com/SamuelLRapp/BlueWaltzBio`. RSB uses the rentrez and bold CRAN packages to access the live NCBI and BOLD databases respectively and the Taxize package for some ease-of-use features. The app then visualizes the output using the shiny, ggplot, and plotly packages.

Alternatively, the application can be installed locally. This has been tested for Windows, Mac-OSX, and Ubuntu. The tool uses R version 4.1.2. To install the app, download the zip file found here

https://github.com/SamuelLRapp/BlueWaltzBio/releases/tag/v1.0.0-stable and then execute the script "rsbPackages.R" to install the correct versions of the required libraries. After that, the app can be run by opening an R console in the directory where the zip file was extracted and executing the command "shiny::runApp('Coverage')". A complete list of our app's package dependencies can be found in "rsbPackages.xlsx" in the zip file above and this paper's "References" section.

Input file structure

All search tabs use the same CSV file format for uploading large inputs. There are two columns: "OrganismNames" and "Barcodes". Every search tab uses the "OrganismNames" column, but "Barcodes" is only used by the NCBI Nucleotide search tab. Every pairing of entries in the "OrganismNames" and "Barcodes" columns will be searched for, so the two columns do not need to have the same number of elements. We provide a CSV template for uploading data in our github in the "paper_methods_species.csv" file that one can download and simply add their own species and barcodes to it.

Tool navigation and outputs

0.1 General features and workflow:

The GUI is split into different tabs for each reference database, which the user can select by clicking on the respective tab at the top of the application window. Regardless of which database the user will be querying, the general workflow remains the same. Users will first be presented with a screen allowing them to upload a formatted CSV file with their query specifications (see the "Operation" section to learn more about the CSV file template). While uploading a CSV file is optional, we highly recommend using this feature when doing large or expansive searches. Afterward, the RSB will present users with a screen where they may manually adjust their search parameters. After running a search, the app will display a table summarizing the results, and a list of download buttons will be present for the user to selectively download either the sequences or any metadata they will need for use with external bioinformatics tools.

For all databases, the user may choose to apply the taxize CRAN package's autocorrect feature to their list of organisms of interest. Instead of overriding the user's original input information, the tool will append any corrections to the user's list of organisms. For more information about how taxize's corrections work, see the taxize CRAN package documentation [11].

Additionally, a tabular summary of the search results can be downloaded from any of the three database tabs, allowing users to view statistics about the overall quantity of data in each given reference database that matches their queries. These tables help users to quickly notice any missing information they may need for their study, and how over-represented or under-represented certain organisms or barcodes are in the search results. The rows of the summary table represent the various barcoding loci. The four columns of the table are the number of sequences found for this barcode, the percentage of all sequences found from the user's search that this barcode accounts for, the number of organisms that have at least one sequence for this barcode, and the number of organisms that have no sequences for this barcode. Figure 1 shows an example of a summary table

While the summary tables are appropriate for seeing aggregated statistics about the search results of a user's query, the RSB also provides a more detailed view of a user's search results via the Coverage Matrix. The Coverage Matrix is a table that allows users to see how much of their needs can be "covered" by the data in the database they

CRUX summary data

Barcodes	Number of Sequences Found	Percent of Total Sequences Found	Number of Organisms with at Least one Sequence	Number of Organisms with no Sequences
Total	5671	100	25	1
18S	103	1.82	23	3
16S	1334	23.52	14	12
PITS	0	0	0	26
CO1	2946	51.95	23	3
FITS	1	0.02	1	25
trnL	0	0	0	26
Vert12S	1287	22.69	3	23

Fig 1. Example of the CRUX summary table as shown in the RSB Shiny app. All summary tables across databases have the same behavior and formatting. Their purpose is to summarize the data gathered and see the total number of species covered by each barcode.

are querying. Each row of the table corresponds to an organism name, each column corresponds to a barcoding loci, and each cell displays the number of sequences found for the given organism-barcode pair. This layout allows the user to quickly scan for zeros or other low numbers in the table and understand where there are currently gaps in the sequence information publicly available to the eDNA community. After any search, users can download the Coverage Matrix (in future sections abbreviated as "CM") and the summary table.

0.2 CRUX database specific features

The CRUX database search tab aims to identify the level of representation of a user-defined set of organisms in the public CALeDNA reference databases. This feature may be used by either scientists interested in using the public CALeDNA reference database or parties interested in working with CALeDNA scientists to conduct a study. The site displays the taxonomic resolution and reference sequence abundance available within the CALeDNA databases for different organisms, which informs the user of how well the databases may fulfill the needs of their study. The databases were created by running the CRUX Pipeline, part of the Anacapa Toolkit [6], on a snapshot of the NCBI Nucleotide database in 2019.

The CM for the CRUX database search tab is slightly different from the other CMs in the tool. The rows still represent an organism, but the columns represent one of the public CRUX databases, and values in the table may not always be numbers. When the tool does not find direct matches in a database, it will instead search for higher taxonomic ranks of that organism until it finds a match. If the RSB performs lower-resolution searches, it will output the rank at which it found a sequence instead of displaying a numerical value. The tool works its way up the following taxonomic ranks: species, genus, family, order, class, phylum, and domain.

The cells of the outputted table will show one of the following:

1. The number of sequences in a database, if direct matches are found.
2. If no direct matches exist, the next most specific taxonomic rank for which sequences exist.

CRUX coverage matrix

	18S	16S	PITS	CO1	FITS	trnL	Vert12S
Canis lupus	2	1270	phylum	1297	order	0	1283
Codium fragile subsp. tomentosoides	order	genus	class	class	order	phylum	0
Heterosiphonia	4	family	phylum	23	order	0	0
Prorocentrum minimum	14	12	0	11	class	0	0
Sargassum muticum	2	1	0	20	1	class	0
Bonnemaisonia hamifera	2	genus	phylum	6	class	0	0
Ficopomatus enigmaticus	1	family	0	family	class	0	class
Teredo navalis	4	4	0	4	phylum	0	class
Marenzelleria viridis	1	1	0	25	class	0	order
Membranipora membranacea	3	genus	0	64	0	0	0

Fig 2. Example of the CM in the CRUX database as shown in the RSB Shiny app. The CM table displays the amount of entries found for each species & barcode combination. In CRUX specifically, if no direct matches are found in a database, it will instead search for higher taxonomic ranks of that organism until it finds a match and displays the taxonomic rank in the table.

3. "0" if nothing is found at any taxonomic rank.

For example, in the CRUX CM shown in Figure 2, there are no sequences for *Canis lupus* in the PITS database at the genus-species level. Therefore, the tool would search for any sequences belonging to the genus *Canis*. These sequences are also not present in the database, so it would search for any sequences from the family of *Canis lupus*, *Canidae*. It keeps going up in taxonomic rank until it finally finds a sequence belonging to the *Canis lupus* phylum, Chordata. Thus the word "phylum" is displayed in that cell of the CM.

0.3 NCBI nucleotide database specific features

The NCBI Nucleotide search tab allows the user to both examine the quantity of existing sequence data for any set of organism names and barcodes and also quickly download those sequences. Unlike the other tabs, this tab allows users to specify which barcode loci they want to search for. As with the organism names, this information can be input either through the CSV file that users can upload or by manually typing any barcode names into a textbox provided in the GUI (See "General Features and Workflow" section for more details). We also provided buttons on the GUI that add groupings of commonly used barcodes and any alternate names they may have to the search (see the "Discussion" section for more information).

By default, the tool will only search for sequences by matching against the "ORGN" and "GENE" Genbank metadata fields. Using this setting may lead to the user finding fewer sequences than what truly exists in NCBI Nucleotide. Therefore, we provide the option to toggle between metadata searches and full-text searches. Due to the inherent inaccuracy of full-text searches, we recommend that users only switch to full-text searches when metadata searches return too few sequences for the needs of the user.

However, the tool can still find entries in NCBI Nucleotide that include sequence data outside the user's requested barcodes. False positives can occur because full mitochondrial genomes or other large sequences often have their metadata labeled with all barcode loci contained within the sequence. To filter out these large sequences, we

provide an optional feature that lets users specify a minimum and maximum base pair length for each barcode sequence. Figure 3 shows the UI elements through which the user can specify minimum and maximum sequence lengths for their query.

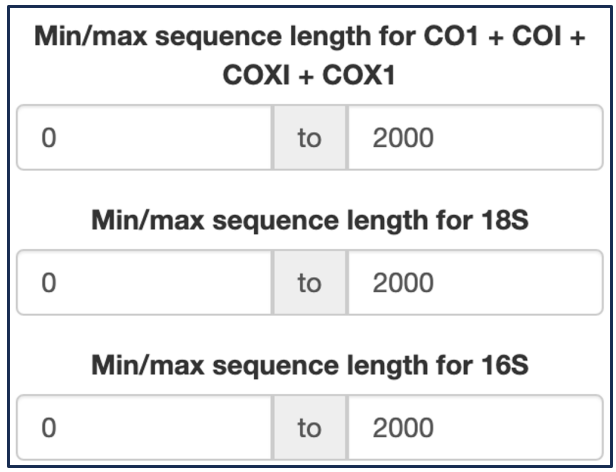

Fig 3. Example of NCBI number range inputs for the user to specify the sequence length (minimum and maximum) for each barcode as shown in the RSB Shiny app. The search results will display only entries in NCBI that are within these user specified ranges.

Once the search is complete, the CM and the summary table will be visible, as described in the "General Features and Workflow" section above. If the user wishes to view the exact search queries sent to NCBI to get these results, a few examples are displayed below the CM (as shown in Figure 4), these queries can be pasted into the NCBI Nucleotide web portal to retrieve the same results we present. The user can also download a CSV file containing the complete list of search queries by clicking a button below the table. The contents of the CSV file follow the same dimensions as the CM. There is also a button for downloading the CM table as a CSV file for usage in external tools or storage.

NCBI search queries

16S	(16S[GENE] AND Canis Lupus[ORGN])
18S	(18S[GENE] AND Canis Lupus[ORGN])
(CO1; COI; COXI; COX1)	(CO1[GENE] AND Canis Lupus[ORGN]) OR (COI[GENE] AND Canis Lupus[ORGN]) OR (COXI[GENE] AND Canis Lupus[ORGN]) OR (COX1[GENE] AND Canis Lupus[ORGN])

Fig 4. Example of the NCBI search queries table as shown in the RSB Shiny app. With these one can easily replicate the results in the table using the NCBI database if necessary.s

Two additional download buttons also exist for retrieving the .fasta and .genbank file contents associated with each sequence. Currently, the tool downloads the sequences into a zip file containing multiple files, one for each barcode.

0.4 BOLD database specific features

The BOLD (Barcode of Life) pipeline of the Reference Sequence Browser screens the BOLD database for genetic barcode coverage. At the beginning of the pipeline, users can upload a CSV file containing all the species they wish to search. They may instead

manually input this list, as described in the "General Features and Workflow" section above. Once inputted, the application will search through the most up-to-date version of the BOLD database using the BOLD Systems API Package for R [9]. Once the search is complete, the users will see a list of all species of interest not found in the database. This list will help quickly check whether the species of interest have reference sequences in BOLD.

Next in the pipeline is the filter tab, which allows users to manipulate the results gathered by filtering the results by country of origin and giving the option to remove any entries also present in NCBI Nucleotide. This allows users to avoid downloading duplicate FASTA files between BOLD and NCBI Nucleotide. All subsequent tables and visualizations depend on this filtering, which can be updated by returning to the filter tab at any time. This way, users can adjust their filters to gather more or fewer sequences as they see fit.

Once the search is complete, the app will first produce a CM and summary table as described in the "General Features and Workflow" section above. This data depends on both the results gathered from BOLD and also the filters the user selected. Additionally, the app provides users with buttons to download the CM, summary table, and the associated FASTA files. Afterward, the RSB presents various tables and graphs in the "Country Data", "Plot Total Sequences per Country", and "Plot Unique Sequences per Country" tabs to help users analyze the data and fine-tune their filters.

In the "Country Data" tab, there are two main tables. The first table has organism names as rows and countries as columns, and each intersection of a row and column shows how many records are in the BOLD database. The second table only displays information when some of the user's species of interest are absent in all selected countries. When this happens, the organism names are rows, and the three columns show the top three countries with the most sequences for that species, ordered from highest to lowest. The purpose of the first table is to provide the user with a breakdown of which countries their sequences are coming from, while the second table suggests additional countries to add to the filter when the current list of selected countries has no sequences for some of their target organisms. The user can download either table by using the buttons below them.

Following these two tables, we provide two plots in the "Plot Total Sequences per Country" and "Plot Unique Sequences per Country" tabs to visualize the geographic distribution of the data. The first plot is a treemap that provides a graphical representation of the distribution of the sequence record amongst the selected countries. The second plot is a bar graph that shows the number of unique species per country. Each of these plots also has a download button to download them as png files.

Finally, there is a table of BOLD entries with unlabeled barcodes. For each organism where this is the case, we display a list of BOLD UIDs so that users may explore these results further by searching in BOLD and looking at the sequences themselves. The rows are the organisms, and the only column displays the list of IDs. The tool also provides a button to download this table at the bottom of the page.

An example of each of these tables and graphs can be found at the end of paper.

Use cases

We envision scientists using this tool to save time and money by speeding up the preprocessing step of checking the reference sequence coverage and downloading reference sequences needed for their studies. Using the many visualizations the tool provides, researchers can quickly determine whether the publicly available reference sequence coverage meets their needs. Below, we outline two specific use cases where the tool can speed up and aid environmental DNA studies. The examples below follow a

hypothetical user who is interested in the invasive amphibians, fish, and invertebrates currently present in California from the California Department of Fish and Wildlife's website.

Category	Species
Amphibians	*Eleutherodactylus coqui, Lithobates catesbeianus, Xenopus laevis*
Fish	*Ctenopharyngodon idella*
Invertebrates	*Dreissena polymorpha, Dreissena bugensis, Eriocheir senensis, Euwallacea kuroshio, Euwallacea whitfordiodendrus, Pomacea canaliculata, Potamopyrgus antipodarum*

Table 1. List of California Invasive Amphibians, Fish, and Invertebrates Species by Category. We have also included this list as a CSV you can upload to the tool in your github in the file "invasive_CA_Species.csv"

Use case 1: Assessing online reference sequence availability

Knowledge of which organisms have publicly available reference sequences at known DNA barcoding loci is crucial for both metabarcoding studies and the design of new primers. In a metabarcoding study, taxa of interest can not be detected if there are no labeled reference sequences to compare to. Similarly, designing species-specific primers is only possible if there are enough reference sequences for both the target taxa and their phylogenetic relatives to represent the genetic diversity across individuals. Unfortunately, countless organisms are yet to be sequenced (Nagarajan et al., 2023), so it is best practice to investigate the current state of sequence availability before beginning any study. The tool can easily accomplish this because it allows researchers to a) find how many reference sequences there are for each species per barcode and b) broaden or narrow their search depending on the results using the various filters and parameters.

The RSB summary data table allows scientists to quickly compare barcodes against each other regarding how well they cover the organisms of interest. Directed by the summary data, scientists can then look at coverage for organisms at the most well-represented barcodes. Looking at the CM, the user can determine which organisms have enough sequences and which organisms currently need sequencing.

Case 1.1: For metabarcoding studies

It is important to note that while any of the three databases are valid for this use case, the CRUX databases deserve special attention. The publicly available CRUX databases were made to be shared for use in metabarcoding studies and contain a curated list of high-quality barcodes. In contrast, NCBI Nucleotide and BOLD also store lower-quality sequences and entries with incomplete metadata. Whichever database is used, the primary source of value comes from the CM and Results Summary tables.

If the user is interested in determining the utility of the public CALeDNA CRUX databases for their study, they can start by following the CRUX pipeline until they reach the Summary Data Figure 5.

As shown in the summary table, of the 11 invasive species searched, six organisms had at least one sequence. The vast majority of the sequences were found in the CO1 barcode and includes five organisms but the 16S barcode captures all six detected organisms. To get more information, the user would then inspect the CM like the Figure 6.

Looking at the CO1 column, one can see examples of the three different ways in which the RSB highlights missing information in the CRUX databases. Firstly,

CRUX summary data

	Barcodes	Number of Sequences Found	Percent of Total Sequences Found	Number of Organisms with at Least one Sequence	Number of Organisms with no Sequences
1	Total	1059	100	6	5
2	18S	6	0.57	3	8
3	16S	62	5.85	6	5
4	PITS	0	0	0	11
5	CO1	970	91.6	5	6
6	FITS	0	0	0	11
7	trnL	0	0	0	11
8	Vert12S	21	1.98	3	8

Fig 5. Example of CRUX Summary Data table for the list of California invasive Amphibians, Fish, and Invertebrates Species

CRUX coverage matrix

	18S	16S	PITS	CO1	FITS	trnL	Vert12S
Eleutherodactylus coqui	order	1	phylum	6	phylum	0	3
Lithobates catesbeianus	family	family	phylum	family	phylum	0	family
Xenopus laevis	3	14	phylum	18	phylum	0	11
Ctenopharyngodon idella	order	9	order	69	order	0	7
Dreissena polymorpha	2	4	0	genus	phylum	0	class
Dreissena bugensis	genus	genus	0	genus	phylum	0	class
Eriocheir senensis	0	0	0	0	0	0	0
Euwallacea kuroshio	family	family	order	genus	order	0	family
Euwallacea whitfordiodendrus	0	0	0	0	0	0	0
Pomacea canaliculata	1	4	0	520	class	0	class
Potamopyrgus antipodarum	family	30	0	357	order	0	order

Fig 6. Example of CRUX CM table for the list of California invasive Amphibians, Fish, and Invertebrates Species

Lithobates catesbeianus, Dreissena polymorpha, and *Euwallacea kuroshio* are not in the database, but sequences exist for other species in the genuses *Lithobates, Dreissena*, and *Euwallacea*. Secondly, it is important to note that the species *Lithobates catesbeianus* was recently updated in the NCBI taxonomic backbone and has been moved to the genus *Aquarana*. Because we use the NCBI taxonomic backbone when searching up the phylogenetic tree, our tool does not look for entries of *Lithobates* which are actually present in the CO1, 16S and Vert12S databases. Additionally, the species name *Euwallacea whitfordiodendrus* is not recognized by the NCBI taxonomic backbone and therefore when it was not found at species level, it was not searched at broader scales like the tool did for *Euwallacea kuroshio*.

Considering these results, the user may either decide that using the 16S and CO1 primers alone will be sufficient for their study, or the user can broaden their search to the NCBI or BOLD search tabs and change any filters as needed. For example, Figure 7 displays the CM output from searching NCBI Nucleotide.

Here, there are CO1 sequences for species that were not present in CRUX. Additional searches could be made in BOLD.

NCBI coverage matrix

	12S	trnL	ITS2	CO1 + COI + COXI + COX1	ITS1	16S	18S
Eleutherodactylus coqui	3	0	0	6	0	5	0
Lithobates catesbeianus	26	0	45	54	41	54	15
Xenopus laevis	70	0	38	342	34	714	577
Ctenopharyngodon idella	24	0	5	191	6	573	23
Dreissena polymorpha	5	0	16	125	5	904	25
Dreissena bugensis	0	0	0	39	0	22	2
Eriocheir senensis	0	0	0	0	0	0	0
Euwallacea kuroshio	0	0	1	8	4	0	1
Euwallacea whitfordiodendrus	0	0	0	0	0	0	0
Pomacea canaliculata	19	0	8	1796	1	215	52
Potamopyrgus antipodarum	36	0	99	403	95	78	104

Fig 7. Example of NCBI CM table for the list of California invasive Amphibians, Fish, and Invertebrates Species

Case 1.2: For species-specific primer design

In the case of species-specific primer design, the user would be interested in populating a local reference database with sequences for the target species, closely related species, and sympatric species (Klymus 2020). In this case, we suggest looking at both the NCBI and BOLD tabs to get the full representation of the available barcodes for the organism(s) listed above.

Here, we describe the example of designing a primer for *Xenopus laevis*, one of the invasive amphibian species mentioned in case 1.1. Because the RSB BOLD tab allows users to deduplicate results also found in NCBI, users can start searching in either NCBI or BOLD first. To get a list of closely related species to search NCBI, users are advised to visit the NCBI taxonomy database to find the complete list of species names under a given taxonomic group. The user would first gather the highest quality sequences they can find from NCBI by using the most restrictive search parameters. For example, users should initially search NCBI by searching via the species and gene metadata fields, which is the default. The metadata for NCBI entries is not always completed, so searching in all fields is sometimes needed. By scanning the CM, the user can look for organisms with poor or no sequence coverage and loosen up the parameters if needed. Users can download the search statements used to search NCBI, which can be copy-pasted into NCBI's web interface to explore and validate results. Searching with [GENE] and [ORGN] but not using the taxize or sequence length parameters, NCBI produced the results shown in Figure 8

Researchers can then quickly determine which barcodes have the most coverage in NCBI. In this case, the CO1 barcode group is the only one with any coverage. If more sequences are needed, the user can search in BOLD. One of the advantages of searching in BOLD is that researchers can simply search using a taxonomic group such as the genus *Xenopus*, and it will return results for all species under that group. Then, by excluding results found in NCBI shown in Figure 9, the user can observe the full unique set of available reference sequences across the two databases.

Following this, a user would search for results with a list of species that co-occur with the target (*Xenopus laevis*) in both BOLD and NCBI. The degree of sequence coverage of these co-species at any of 16S, 12S, or COI may help the user decide on a desirable loci for primer design.

NCBI coverage matrix

Barcodes	Number of Sequences Found	Percent of Total Sequences Found	Number of Organisms with at Least one Sequence	Number of Organisms with no Sequences
Total	394	100	34	44
12S	0	0	0	78
trnL	1	0.25	1	77
ITS2	0	0	0	78
CO1 + COI + COXI + COX1	393	99.75	34	44
ITS1	0	0	0	78
16S	0	0	0	78
18S	0	0	0	78

Fig 8. Example of NCBI Summary Data table for the genus Xenopus

BOLD coverage matrix

	COI-5P	CYTB	COXIII	COII	atp6
Xenopus (Silurana) tropicalis	1	0	0	0	0
Xenopus laevis	3	1	1	1	1
Xenopus muelleri	4	0	0	0	0

Fig 9. Example of BOLD CM table for the list of California invasive Amphibians, Fish, and Invertebrates Species

Use case 2: Creating a local reference sequence database

In some procedures, once a researcher has assessed the online reference sequence availability, they create a local reference sequence database. As outlined in "Development and Testing of Species-specific Quantitative PCR Assays for Environmental DNA Applications" (Klymus 2020), the initial step of designing a specific quantitative PCR assay is creating a local reference sequence database. Following the use case 1.2 described above, RSB facilitates this process by making downloading sequences in bulk from NCBI and BOLD simple.

Only the NCBI Nucleotide and BOLD search tabs of the RSB currently support downloading sequences. In either case, fasta files will be downloaded as a zip file containing one file for each barcode loci included in the search results. Within each file, sequences are separated by newlines. This formatting is compatible with several alignment tools (e.g., Geneious, MegAlign Pro, DNADynamo), so scientists can move on to the next steps in the pipeline outlined in Klymus et al.'s publication.

Discussion

Screening of and access to available reference sequences is necessary for the development of new quantitative PCR (qPCR) primers, to understand the efficacy of existing primers, and to understand gaps in what taxa can be identified in eDNA samples for given DNA barcodes. However, access to reference sequence information for the numerous organisms eDNA scientists hope to detect or exclude via primer design requires learning bioinformatic skills or investing large amounts of time into manual searches. With the RSB, users can easily screen NCBI, BOLD, and the public CALeDNA CRUX databases for reference sequence availability and download sequences for numerous organisms at multiple DNA barcodes. Therefore, users can generate deduplicated, up-to-date local

sequence databases from the combined pools of BOLD and NCBI for primer design by using the NCBI filter in the BOLD tab. Additionally, summary statistics and other data visualization features allow users to glean other insights about reference sequence availability. Therefore, this tool can be used to explore gaps in reference sequence availability and help direct future sequencing efforts and funds. The RSB GUI bridges the gap between ecologists and computer scientists by providing efficient and intuitive access to NCBI, BOLD, and CRUX databases without requiring any programming.

While the RSB will be very helpful to eDNA scientists of all backgrounds and levels of experience, the tool does have limitations and space for some improvements. Text searches in NCBI Nucleotide create the possibility of false positives and negatives, which we attempted to address via the [GENE] and [ORGN] metadata search options. Updating the UI to better highlight oddities in the data, such as homonyms in organism names, would give users more control over the quality of their datasets. Additionally, new features, such as the ability to filter by geographic regions or to search for all species under a specific genus (or other higher taxonomic rank), could open up new use cases. We are also aware that the scientific name searches performed by this tool are vulnerable to inaccurate results when searching for organism names for which homonyms exist. Adding some way of handling homonyms would make the tool more robust and easier to use. Finally, a great quality-of-life feature would be highlighting all the changes made by taxize so that users can more easily notice when taxize adds unwanted organisms to the search query and a way to download sequences from the CRUX database similar to features NCBI and BOLD already posses.

Our hope is that by opening up this tool to the eDNA community people can contribute to the github and add more quality of life features and/or incorporate more databases. One could also simply fork our repository and make their own personalized features that they would like to have for their own use. Additionally, while we made it so the server can hold as many users as we could we have some scaling concerns, however, the tool can be easily used locally instead if any problems arise.

Lastly, our github contains all the necessary input files to replicate the results of this paper however it is important to keep in mind that these databases are constantly being updated and therefore the exact numbers in our tables may not exactly match searches done in the future.

Software availability

- A Shiny app, is available at
  `https://sriramramesh.shinyapps.io/ReferenceSequenceBrowser/`
- Source code is available from GitHub:
  `https://github.com/SamuelLRapp/BlueWaltzBio`
- Software license (GPL-3)

Acknowledgements

We thank Hailey Nava for helping during the interview and research phase, Nitya Jain for helping in the BOLD pipeline particularly with the creation of the graphs, Jay Ferreira for helping us refactor and test our code, Dr. Rachel Meyer for guidance, review of the written process and helping us host our tool in the UCLA EEB servers.

Lastly, we would like to thank all the scientists we have interviewed in the last 3 years who helped us identify workflow bottlenecks in the field of eDNA and improve this tool.

References

1. Seymour M. Rapid progression and future of environmental DNA research. Communications Biology. 2019;2. doi:10.1038/s42003-019-0330-9.

2. Thomsen PF, Willerslev E. Environmental DNA – An emerging tool in conservation for monitoring past and present biodiversity. Biological Conservation. 2015;183:4–18. doi:https://doi.org/10.1016/j.biocon.2014.11.019.

3. Andersen J, Møller P, Kallenbach E, Hesselsøe M, Knudsen S, Bekkevold D, et al. Steps toward nation-wide monitoring of non-indigenous species in Danish marine waters under the Marine Strategy Framework Directive. NIVA; 2016.

4. Klymus KE, Hrabik RA, Thompson NL, Cornman RS. Genome resequencing clarifies phylogeny and reveals patterns of selection in the toxicogenomics model Pimephales promelas. PeerJ. 2022;10:e13954.

5. Nagarajan R, Bedwell M, Holmes A, Sanches T, Acuña S, Baerwald M, et al. Environmental DNA Methods for Ecological Monitoring and Biodiversity Assessment in Estuaries. Estuaries and Coasts. 2022;45. doi:10.1007/s12237-022-01080-y.

6. Curd EE, Gold Z, Kandlikar GS, Gomer J, Ogden M, O'Connell T, et al. Anacapa Toolkit: An environmental DNA toolkit for processing multilocus metabarcode datasets. Methods in Ecology and Evolution. 2019;10(9):1469–1475. doi:https://doi.org/10.1111/2041-210X.13214.

7. Hernandez RR, Mayernik MS, Murphy-Mariscal ML, Allen MF. Advanced Technologies and Data Management Practices in Environmental Science: Lessons from Academia. BioScience. 2012;62(12):1067–1076. doi:10.1525/bio.2012.62.12.8.

8. Winter DJ. rentrez: an R package for the NCBI eUtils API. The R Journal. 2017;9:520–526.

9. Chamberlain S. bold: Interface to Bold Systems API; 2021. Available from: `https://CRAN.R-project.org/package=bold`.

10. Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, et al.. shiny: Web Application Framework for R; 2023. Available from: `https://CRAN.R-project.org/package=shiny`.

11. Chamberlain S, Szoecs E, Foster Z, Arendsee Z, Boettiger C, Ram K, et al.. taxize: Taxonomic information from around the web; 2020. Available from: `https://github.com/ropensci/taxize`.

12. R Core Team. R: A Language and Environment for Statistical Computing; 2021. Available from: `https://www.R-project.org/`.

13. Wilkins D. treemapify: Draw Treemaps in 'ggplot2'; 2021. Available from: `https://CRAN.R-project.org/package=treemapify`.

14. Wickham H, Girlich M, Ruiz E. dbplyr: A 'dplyr' Back End for Databases; 2021. Available from: `https://CRAN.R-project.org/package=dbplyr`.

15. Wickham H, François R, Henry L, Müller K. dplyr: A Grammar of Data Manipulation; 2022. Available from: `https://CRAN.R-project.org/package=dplyr`.

16. Wickham H. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York; 2016. Available from: https://ggplot2.tidyverse.org.

17. Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, et al. Welcome to the tidyverse. Journal of Open Source Software. 2019;4(43):1686. doi:10.21105/joss.01686.

18. Weigand H, Beermann AJ, Čiampor F, Costa FO, Csabai Z, Duarte S, et al. DNA barcode reference libraries for the monitoring of aquatic biota in Europe: Gap-analysis and recommendations for future work. Science of The Total Environment. 2019;678:499–524. doi:https://doi.org/10.1016/j.scitotenv.2019.04.247.

19. Warnholz S. modules: Self Contained Units of Source Code; 2021. Available from: https://CRAN.R-project.org/package=modules.

20. Van Rossum G, Drake FL. Python 3 Reference Manual. Scotts Valley, CA: CreateSpace; 2009.

21. Sievert C. Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC; 2020. Available from: https://plotly-r.com.

22. pandas development team T. pandas-dev/pandas: Pandas; 2020. Available from: https://doi.org/10.5281/zenodo.3509134.

23. Wes McKinney. Data Structures for Statistical Computing in Python. In: Stéfan van der Walt, Jarrod Millman, editors. Proceedings of the 9th Python in Science Conference; 2010. p. 56 – 61.

24. Perrier V, Meyer F, Granjon D. shinyWidgets: Custom Inputs Widgets for Shiny; 2022. Available from: https://CRAN.R-project.org/package=shinyWidgets.

25. R Special Interest Group on Databases (R-SIG-DB), Wickham H, Müller K. DBI: R Database Interface; 2021. Available from: https://CRAN.R-project.org/package=DBI.

26. Müller K, Wickham H, James DA, Falcon S. RSQLite: SQLite Interface for R; 2021. Available from: https://CRAN.R-project.org/package=RSQLite.

27. Marques V, Milhau T, Albouy C, Dejean T, Manel S, Mouillot D, et al. GAPeDNA: Assessing and mapping global species gaps in genetic databases for eDNA metabarcoding. Diversity and Distributions. 2021;27(10):1880–1892. doi:https://doi.org/10.1111/ddi.13142.

28. Lyttle I. vembedr: Embed Video in HTML; 2021. Available from: https://CRAN.R-project.org/package=vembedr.

29. Kahle D. mpoly: Multivariate Polynomials in R. The R Journal. 2013;5(1):162–170.

30. Fellows IE. ipc: Tools for Message Passing Between Processes; 2019. Available from: https://CRAN.R-project.org/package=ipc.

31. Cheng J. promises: Abstractions for Promise-Based Asynchronous Programming; 2021. Available from: https://CRAN.R-project.org/package=promises.

32. Chang W, Borges Ribeiro B. shinydashboard: Create Dashboards with 'Shiny'; 2021. Available from: https://CRAN.R-project.org/package=shinydashboard.

33. Bengtsson H. A Unifying Framework for Parallel and Distributed Processing in R using Futures; 2021. Available from: https://journal.r-project.org/archive/2021/RJ-2021-048/index.html.

34. Bailey E. shinyBS: Twitter Bootstrap Components for Shiny; 2022. Available from: https://CRAN.R-project.org/package=shinyBS.

35. Attali D, Edwards T. shinyalert: Easily Create Pretty Popup Messages (Modals) in 'Shiny'; 2021. Available from: https://CRAN.R-project.org/package=shinyalert.

36. Sali A, Attali D. shinycssloaders: Add Loading Animations to a 'shiny' Output While It's Recalculating; 2020. Available from: https://CRAN.R-project.org/package=shinycssloaders.

37. Attali D. shinyjs: Easily Improve the User Experience of Your Shiny Apps in Seconds; 2021. Available from: https://CRAN.R-project.org/package=shinyjs.