EcoHealth Alliance

EIDITH R

Local conservation.
Global health.

EcoHealth Alliance

Getting Started: Installation

```
install.packages(c("devtools", "tidyverse"))

devtools::install_github("ecohealthalliance/eidith@dev")
```

When the package is done installing, it will prompt you download the database and ask you for your EIDITH username and password. This is the same username and password that you've used to log in to the EIDITH website.

We'll look at changing your settings so that you don't have to type your username and password every time later on…

Note: It's possible that all of us trying this at once will cause some unexpected difficulties…

EcoHealth Alliance

EIDITH R Package Concept

IM Team EIDITH P2 Data

METABIOTA

API

Your Computer

Local conservation.
Global health.

EcoHealth Alliance

Deleting and Downloading Database

The package installation process should lead you through successfully downloading the EIDITH database to your local machine.

However, it might not work perfectly!

In which case you can use the `ed_db_delete()` function to remove your malformed local database and try a clean install using the `ed_db_download()` function.

`ed_db_delete()`
Cleans out and deletes your locally downloaded EIDITH database.

`ed_db_download()`
Will download *entire* P1 and P2 database to your machine when used with no arguments.

EcoHealth Alliance

Complete P2 EIDITH Structure

Local conservation.
Global health.

EcoHealth Alliance

EIDITH Tables and Puller Functions

Once the EIDITH database has been successfully downloaded onto your machine, you can access the different P2 EIDITH tables using specific functions for each table:

Event table: `ed2_events()`

Animal table: `ed2_animals()`

Specimen table: `ed2_specimens()`

Test table: `ed2_tests()`

Human table: `ed2_human()`

ExtractiveIndustry table: `ed2_extractive_industry()`

HumanZoo table: `ed2_human_zoo()`

NaturalAreas table: `ed2_natural_areas()`

TestDataSerology table: `ed2_test_serology()`

+ 16 more tables! So many tables!!

Let's take a look!

EcoHealth Alliance

Relational Databases

When the EIDITH R package downloads the EIDITH data onto your local machine, it is organized into a set of tables with unique identifiers in a relational database structure.

In a database like this, each table usually represents a logical unit of analysis. For instance, there will be an **Event** table which lists all events, an **Animal** table that lists all animals, and a **Specimen** table that lists all specimens.

In order for a database like this work efficiently, each table should really only contain information about it's own unit level – for instance, the **Animal** table should only contain information about *animals*, not information about *events*.

If the **Animal** table contained **Event** information, this information would be repeated for every animal from the same event. When you have lots of levels – you can see this duplication can really add up.

EcoHealth Alliance

Unique IDs and Joining

Another requirement of a well-tuned relational database is that each row is unique – a particular animal won't show up twice in the **Animal** database, and there won't be a row of totals or summary statistics at the bottom of the table.

In order to differentiate between entries, there has to be some column (or combination of columns) that is unique for each entry – this is called a *unique identifier* or a *primary key*.

When two tables are connected, they will contain their own unique identifier, and the unique identifier of their parent table. This allows us to combine information from different tables by "joining" or "merging" these tables.

EcoHealth Alliance

In this example, the **Animal** table's unique ID is **animal_id**, while the **Event** table's unique ID is **event_name**. The **Animal** table also includes **event_name**, which allows us to link each animal entry to its event.

In the **Event** table, each row must have a unique **event_name**. But in the **Animal** table, multiple animals can share the same **event_name** (this makes sense, because there can be animals from the same event!)

When two tables are related, they will contain both their own unique identifier, and the unique identifier of their parent table. This allows us to combine information from different tables by "joining" or "merging" these tables.

*Let's look at a toy example of joining small **Event** and **Animal** tables together:*

EcoHealth Alliance

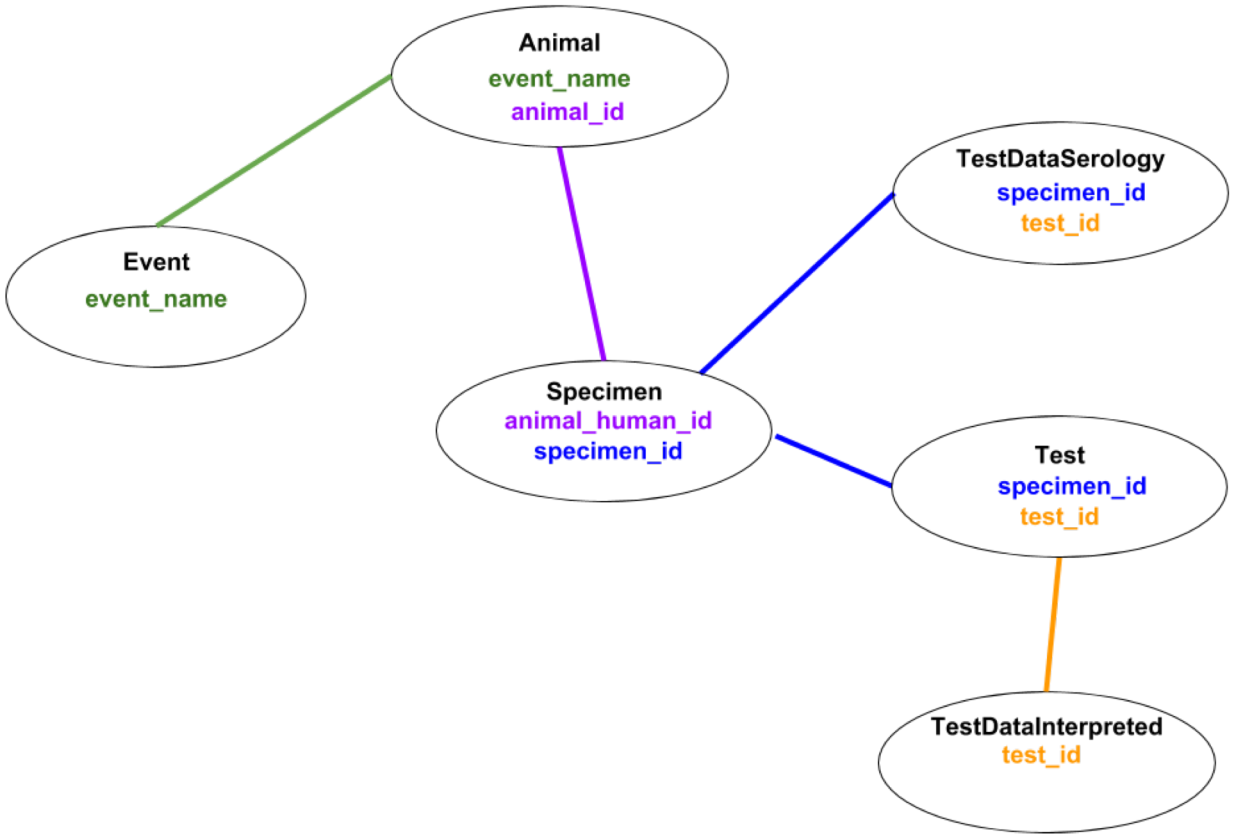
Complete P2 EIDITH Structure

Local conservation.
Global health.

EcoHealth Alliance

Animal EIDITH Structure

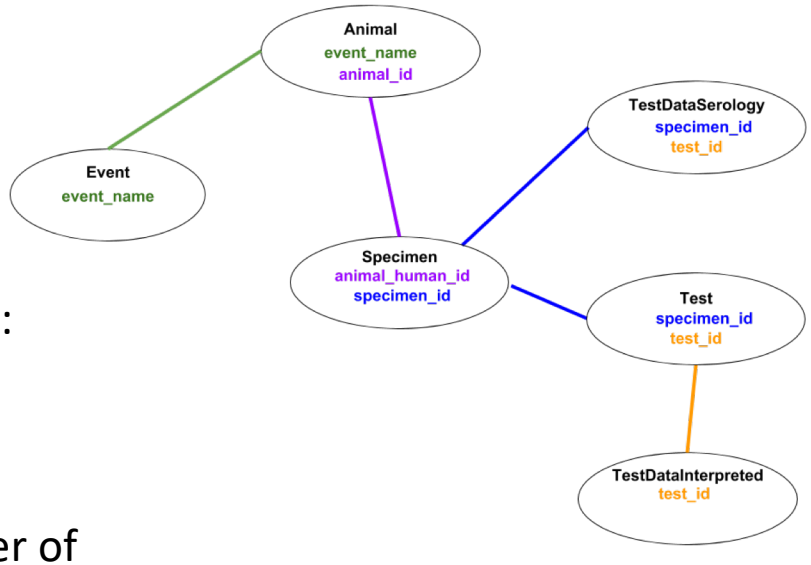
Each table is linked to the next via a unique identifier. The diagram below shows table linkages for animal sample information.



Local conservation.
Global health.

EcoHealth Alliance

Simple EIDITH Example



Let's look at a relatively simple example:

We want:

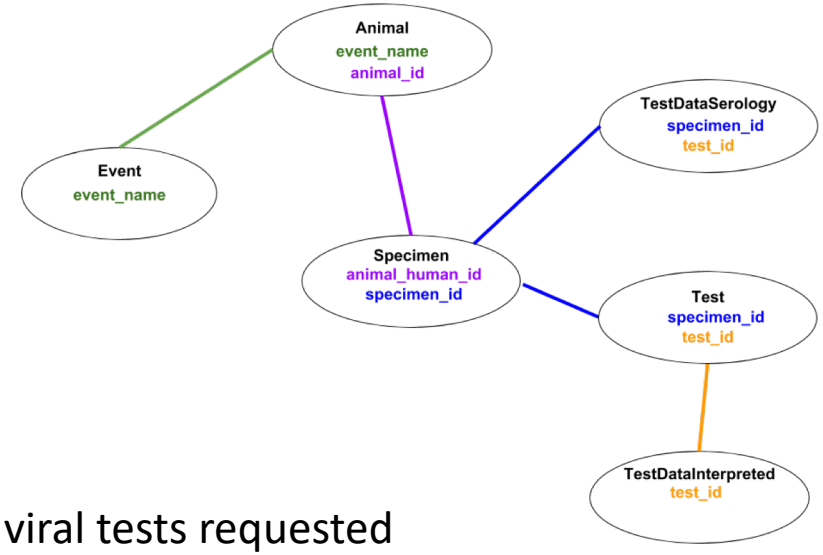
(1) To make a bar graph with the number of species sampled per country

(2) To make a bar graph with the number of animals sampled per country, perhaps even broken up by species

To do this, we will need to join the **Event** table (for information about events, like *country*) with the **Animal** table (for information about animals, like *species*)

EcoHealth Alliance

Another Joining Example



How about an example with more tables:

We want:

(1) To make a bar graph with the types of viral tests requested for each country

To do this, we will need to:

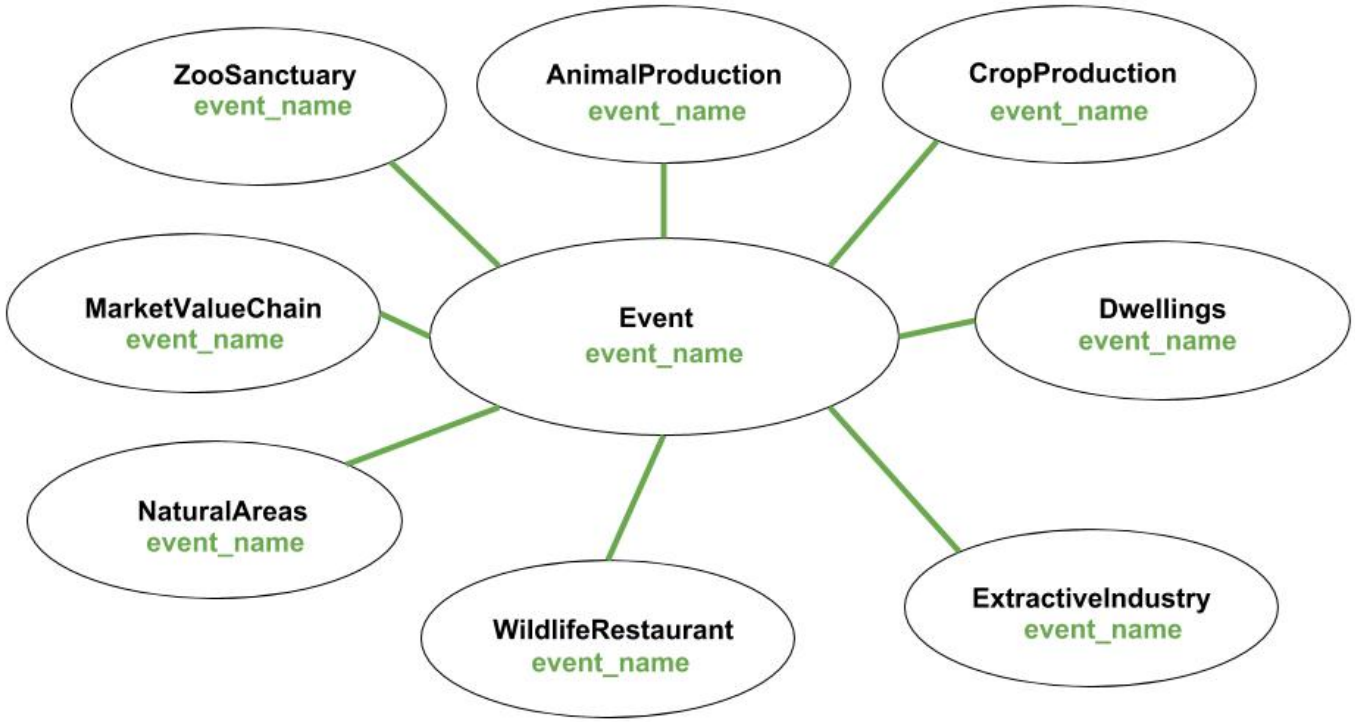
Join the **Event** table (for information about events, like *country*) with the **Animal** table, and then the **Specimen** table, and finally the **Test** table (for information about tests, like *test_requested*).

This is an example of having to join intermediate tables whose data we aren't necessarily using in order to link together non-adjacent tables.

EcoHealth Alliance

Site Characterization Modules

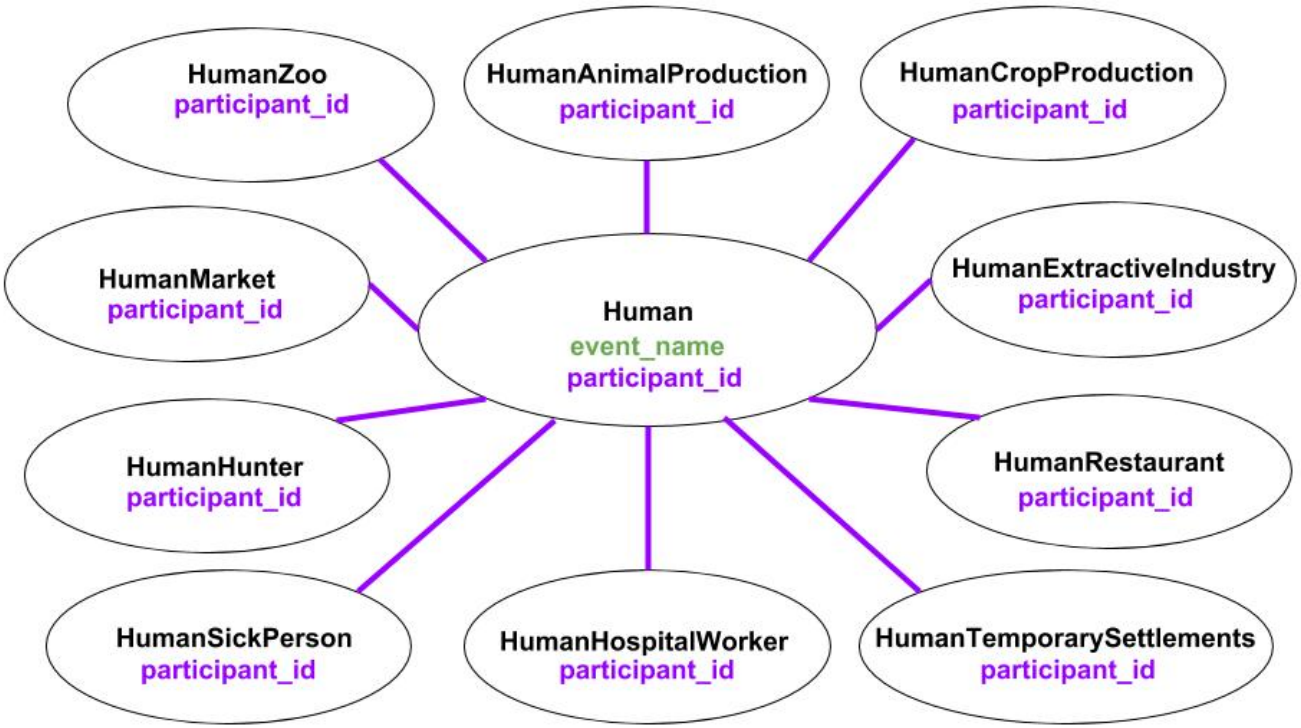
All modules are linked back to the main **Event** table via the **event_name** unique identifier:



EcoHealth Alliance

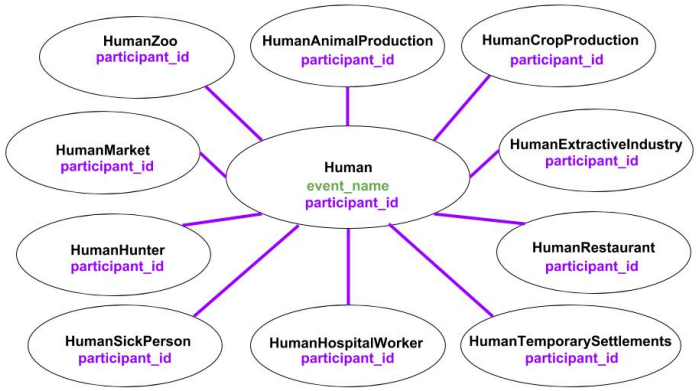
Human Questionnaire Modules

All livelihood-specific questionnaire modules are linked back to the main Human Questionnaire via the **participant_id** unique identifier:



EcoHealth Alliance

Human Questionnaire Example



We want:

(1) To make a bar graph of animals that questionnaire respondents work with at zoos.

(2) Include age / gender distinctions in the graph.

To do this, we will need to:

Join the **Human** table (that contains demographics like age / gender) with the **HumanZoo** table which has information about which animals respondents work with.

EcoHealth Alliance

Filtering a Table Pull

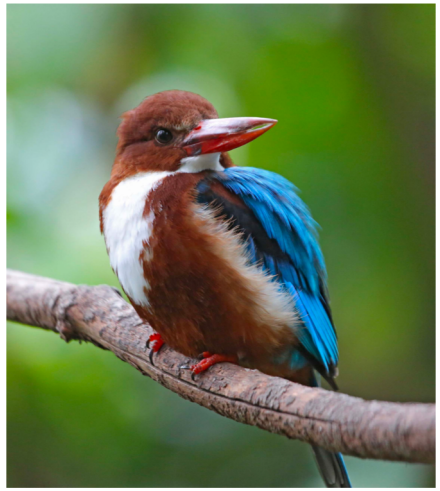
Let's say we want to quickly pull a filtered table – for instance we want only white-throated kingfishers from the **Animal** table. We can do this in one nifty step by entering:

```
kingfishers <- ed2_animals(species_scientific_name == 'Halcyon smyrnensis')
```

This filters the pull from our local database in one step, but obviously we need to know ahead of name of the field we want to filter.

A similar pull of just **Events** from Bangladesh would be:

```
bangladesh_events <- ed2_events(country == "Bangladesh")
```



EcoHealth Alliance

Snafus: Downloading

The most challenging part of developing this package has been getting it to effectively download tables without errors. There are a fair number of things that can go wrong:

- API failure (something wrong up at Metabiota)
- Expected fields are missing or their names have changed
- Internet connectivity issues

There is built-in error-catching to hopefully deal with these errors, but there will likely be some unexpected problems.

While we all break in the package, the best way to deal with errors is to contact Cale or Noam directly, or post an issue on the github issues page:

https://github.com/ecohealthalliance/eidith/issues

EcoHealth Alliance

Deleting and Downloading Database

The package installation process should lead you through successfully downloading the EIDITH database to your local machine.

However, it might not work perfectly!

In which case you can use the `ed_db_delete()` function to remove your malformed local database and try a clean install using the `ed_db_download()` function.

`ed_db_delete()`
Cleans out and deletes your locally downloaded EIDITH database.

`ed_db_download()`
Will download *entire* P1 and P2 database to your machine when used with no arguments. You can specify which tables you'd like to download using the `p1_tables` and `p2_tables` arguments. If you need to know what endpoints are available, you can try the `p1_api_endpoints()` and `p2_api_endpoints()` functions.

EcoHealth Alliance

Snafus: Note Columns

For most EIDITH tables, there is some sort of *notes* column for interviewers or recorders to add free text notes.

The EIDITH R Package checks to see whether each table has a notes column (that isn't entirely blank), and whenever you pull a local table with an `ed2_*` function, or print a table to a console, you'll get a `MESSAGE` about the presence of a non-empty notes column.

We can take a look at `ed2_events()` to see a notes column that contains some relevant notes…

EcoHealth Alliance

Snafus: Duplicate ID's

The database structure we have requires unique identifiers for each table. For example: one **animal_id** per animal in the **Animal** table.

However, there are sometimes data entry (?) errors in tables, resulting in duplicate unique IDs.

The EIDITH R package automatically checks for duplicate ID's, and whenever you pull a local table with an `ed2_*` function, or print a table to a console, you'll get an `IMPORTANT MESSAGE` about the duplication found.

Once again we can look at an `ed2_events()` pull to see this in action.

EcoHealth Alliance

Saving EIDITH Username/Password

Instead of entering your EIDITH Username and Password each time you download tables, you can save them for later use in R.

There is a nice package called **usethis** that lets you do this very easily:

```
install.packages("usethis")
usethis::edit_r_environ()
```

This will open up a text file in RStudio, where you'll want to enter your information:

```
EIDITH_USERNAME=your_username
EIDITH_PASSWORD=your_password
```

Save this file, restart R, and you're all set!

A guide to doing this is also in one of the EIDITH help files: `?ed_auth`

EcoHealth Alliance

Getting Help / Additional Resources

EIDITH R Package

- Internal help files: `?ed_auth`, `?ed_db_download`
- Searchable Metadata: `?ed_metadata` or `?ed2_metadata`
- EIDITH R Website: https://ecohealthalliance.github.io/eidith/index.html
- Talk to Cale / Noam!

Broader R Help:

- EHA M&A Handbook:
  - https://ecohealthalliance.github.io/eha-ma-handbook/
- # r-discuss Slack Channel

EcoHealth Alliance

Helper Functions

When we were doing the example on Human Questionnaire data we saw the helper function `expand_column_long()` which took a semicolon-separated field and expanded it into long form for easier plotting.

This function (or something similar) will be used so often that it really should be built into the EIDITH R package – so it will be!

During work on P1 EIDITH data we added a helper function `ed_fasta()` which takes virus sequences from the P1 **Viruses** table and converts them to a FASTA file.

EcoHealth Alliance

Discussion: Helpful Functions / Recipes

What are some useful functions or recipes for outputs that you'd like to see?

Some Ideas:

- Showing site locations with mapview()
- Key question response summaries
- Country goal processes
- Recreating / improving EIDITH website reports

More Ideas:

- https://hackmd.io/j7YLwrLNRta9j7V6noGn3g