Enable programmatic data access for eLTER-RI :: **CHEAT SHEET**

DEIMS-SDR entities

Functions to interact with the DEIMS-SDR (deims.org) content and API.

Site

```
site <- get_site_info(
  deimsid = "https://deims.org/f30007c4-8a6e-4f11-ab87-569db54638fe",
  categories = NA,
  show_map = TRUE,
  with_locations = TRUE
)
```

Basic site attributes (tibble) + embeddable map (leaflet)

categories = Affiliations + affiliation attributes

categories = Contacts + contact attributes

categories = EnvCharacts + environmental characteristics attributes

categories = General + general attributes

categories = Infrastructure + infrastructure attributes

categories = observedProperties + observed properties attributes

categories = RelateRes + related resources attributes

Location

```
location <- get_location_info(
  locationid = "https://deims.org/location/85dc6019-9654-4ba0-8338-08c4ffe8fe47",
  show_map = TRUE
)
```

Location attributes (tibble) + embeddable map (leaflet)

Activity

```
activity <- get_activity_info(
  activityid = "https://deims.org/activity/8786fc6d-5d70-495c-b901-42f480182845",
  show_map = TRUE
)
```

Activity attributes (tibble) + embeddable map (leaflet)

Dataset

```
dataset <- get_dataset_info(
  datasetid = "https://deims.org/dataset/38d604ef-decb-4d67-8ac3-cc843d10d3ef",
  show_map = TRUE
)
```

Dataset attributes (tibble) + embeddable map (leaflet)

Sensor

```
sensor <- get_sensor_info(
  sensorid = "https://deims.org/sensors/3845475c-4aec-4dd7-83b4-0ab6ba95db35",
  show_map = TRUE
)
```

Sensor attributes (tibble) + embeddable map (leaflet)

Network

```
sitesList <- get_network_sites(
  networkDEIMSID = "https://deims.org/network/7fef6b73-e5cb-4cd2-b438-ed32eb1504b3"
)
```

Table of network's site (tibble) + embeddable map (leaflet)

Installation

```
utils::install.packages("ReLTER",
repos = "https://ropensci.r-universe.dev")
```

Or

```
devtools::install_github("ropensci/ReLTER")
```

Graphs and Images

Functions to create picture of site, site' network or site's qrcode

```
map_LTERGermanSites <- produce_network_points_map(
  networkDEIMSID = "https://deims.org/networks/e904354a-f3a0-40ce-a9b5-61741f66c824",
  countryCode = "DEU"
)
```

```
siteMap <- produce_site_map(
  deimsid = "https://deims.org/f30007c4-8a6e-4f11-ab87-569db54638fe",
  scale_location = "bl",
  arrow_location = "tl",
  inset_position = "br"
)
```

```
qrcode <- produce_site_qrcode(
  deimsid = "https://deims.org/f30007c4-8a6e-4f11-ab87-569db54638fe"
)
```

Site's observed properties displayed with pie or waffle charts

```
produce_site_observedProperties_pie(
  deimsid = "https://deims.org/f30007c4-8a6e-4f11-ab87-569db54638fe"
)
```

```
produce_site_observedProperties_waffle(
  deimsid = "https://deims.org/f30007c4-8a6e-4f11-ab87-569db54638fe"
)
```

Docker container

```
docker pull ptagliolato/rocker_relter
```

```
docker run -d -e PASSWORD=yourpassword -p 8080:8787 ptagliolato/rocker_relter
```

Integrate

Functions to integrate and harmonize data from third-party repositories at various levels (e.g., format, units of measurement, time, semantic level, and taxonomic level)

Species occurrences

```
occ_SRC <- get_site_speciesOccurrences(
  deimsid = "https://deims.org/97ff6180-e5d1-45f2-a559-8a7872eb26b1",
  list_DS = c("gbif", "inat ", "obis"),
  show_map = TRUE,
  limit = 100
)
```

embeddable map (leaflet)

Darwin Core terms were used for the table fields.

Table of observations retrieved from the repositories specified in **list_DS**.

Taxonomic harmonization and entichment

Your 'table' includes a column (the third) with taxonomic information

```
output <- taxon_id_pesi(
  table = table,
  taxaColumn = 3
)
```

Authoritative taxonomic data from PESI are added to the 'table':
scientificNameAuthorship, synonyms, scientificNameID, url, nameAccordingTo, taxonomicStatus, and *taxonRank*

Your 'table' includes a column (the third) with taxonomic information

```
output <- taxon_id_worms(
  input = table,
  taxaColumn = 3,
  verbose = FALSE,
  refine = FALSE
)
```

Authoritative taxonomic data from WoRMS are added to the 'table':
scientificNameAuthorship, taxonID, taxonomicStatus, synonyms, taxonRank, scientificNameID, nOfWormsResults, wormsRecords

ReLTER is partially founded under H2020 eLTER-Plus Project GA n. 871128 • CC BY SA • alessandro.oggioni@cnr.it • Learn more at **docs.ropensci.org/ReLTER** • ReLTER 2.0.1 • Updated: 2025-06

Enable programmatic data access for eLTER-RI :: **CHEAT SHEET**

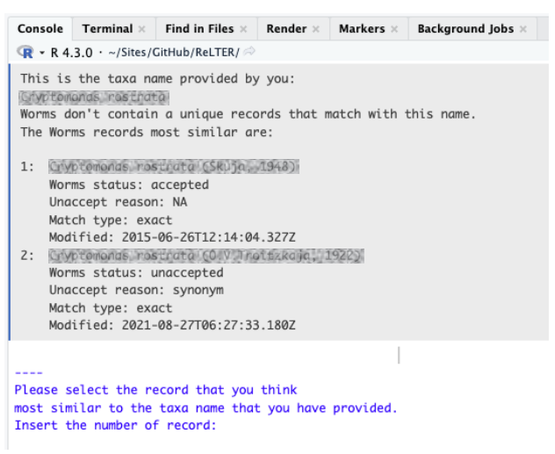
Integrate

Functions to integrate and harmonize data from third-party repositories at various levels (e.g., format, units of measurement, time, semantic level, and taxonomic level)

Taxonomic refine

Your 'table' includes a column (the third) with taxonomic information

```
output <- taxon_id_worms(
  input = table,
  taxaColumn = 3,
  verbose = FALSE,
  refine = TRUE
)
```



Authoritative taxonomic data from WoRMS are added to the 'table':
scientificNameAuthorship, taxonID, taxonomicStatus, synonyms, taxonRank, scientificNameID, nOfWormsResults, wormsRecords

Zenodo

```
record <- get_zenodo_data(
  doi = "10.5281/zenodo.7041152",
  rdata_exist = TRUE
)
```

record metadata (tibble) + file downloaded

… or to cut a piece of a large geo-referenced dataset using one or more geographical features from another dataset

MODIS

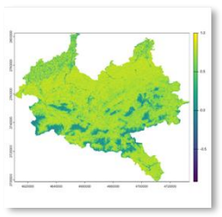
Gran Paradiso National Park - Italy, 1 year time series of LAI and aggregated map

```
download_list <- get_site_MODIS(
  deimsid = "https://deims.org/e33c983a-19ad-4f40-a6fd-1210ee0b3a4b",
  product = "LAI",
  from_date = "2020.01.01",
  to_date = "2020.12.31",
  output_dir = output_dir,
  output_proj = "3035",
  download_range = "Full",
  plot_ts = TRUE,
  show_map = "mean"
)
```

ODS

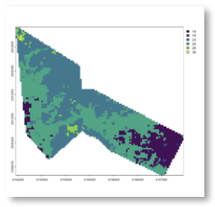
LTSER Platform Eisenwurzen (EW)

```
siteNDVI <- get_site_ODS(
  deimsid = "https://deims.org/d0a8da18-0881-4ebe-bccf-bc4cb4e25701",
  dataset = "ndvi_summer"
)
siteNDVI
terra::plot(siteNDVI)
```



Istituto Scientifico Angelo Mosso (MOSSO)

```
siteLandcover <- get_site_ODS(
  deimsid = "https://deims.org/17210eba-d832-4759-89fa-9ff127cbdf6e",
  dataset = "landcover"
)
siteLandcover
terra::plot(siteLandcover)
```



eLTER DRF

Functions to provide data using the Data Reporting Format (DRF)

```
research_object <- reporting_produce_data_object_v2.0(
  station = station,
  method = method,
  data = data,
  deimsid = "https://deims.org/8eda49e9-1f4e-4f3e-b58e-e0bb25dc32a6",
  data_topic = "VEG",
  variable_group = "SPECCOVER",
  time_span = 2015,
  version = "V20220907"
)

filename <- reporting_compose_file_name(
  deimsid = "https://deims.org/8eda49e9-1f4e-4f3e-b58e-e0bb25dc32a6",
  data_topic = "VEG",
  variable_group = "SPECCOVER",
  time_span = 2015,
  version = "V20220907"
)
```

Internal function

"AUT_LTER-Zöbelboden---Austria_VEG_SPECCOVER_2015_V20220907"

```
archive <- reporting_save_archive(
  x = research_object,
  filepath = ".",
  saveRDS = TRUE
)
```

ZIP

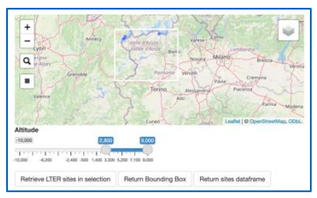
Shiny interface

Programmatically

get_sites_within_3d_bounding_box()
get_sites_within_radius()

Visually (Shiny intreface)

get_sites_interactive()



(sf) all sites within 3D bbox

Configuration

Functions to retrieve the current setting and, if needed, to configure a new one.

Get

get_deims_base_url()

https://deims.org/api

get_deims_API_version()

```
{
  "openapi": "3.0.0",
  "info": {
    "title": "DEIMS-SDR API",
    …
    "version": "1.1"
  },
```

Set

set_deims_base_url()

Citation

Oggioni, A., Silver, M., Tagliolato, P., & Karnieli, A. (2025). ReLTER: An R interface for environmental observation in long term ecological research. *Ecological Informatics*, *85*, 102915. https://doi.org/10.1016/j.ecoinf.2024.102915

Contribute

Propose an idea
Report a bug
Contribute code

ReLTER is partially founded under H2020 eLTER-Plus Project GA n. 871128 • • alessandro.oggioni@cnr.it • Learn more at **docs.ropensci.org/ReLTER** • ReLTER 2.0.1 • Updated: 2025-06