eidosapi, un paquete para interactuar con la base de datos de biodiversidad EIDOS

Héctor Miranda-Cebrián\*1

1. Instituto Pirenaico de Ecología - CSIC

Autor de correspondencia\*: Héctor Miranda-Cebrián [[hectorm94@gmail.com](mailto:hectorm94@gmail.com)]

# Palabras clave

Taxonomía; Estandarización; Nomenclatura; Lista Patrón

# Keywords

Taxonomy; Standardization; Nomenclature; Checklist

El trabajo en ecología y cualquiera de sus disciplinas requiere información actualizada, o al menos estandarizada, sobre la nomenclatura de nuestros organismos de estudio (Patterson et al. (2010)). Para solventar este problema, se han ido desarrollando diferentes herramientas que permitan cotejar la nomenclatura de una especie contra bases de datos taxonómicas. Por un lado, tenemos herramientas como las webs de Flora Ibérica y Fauna Ibérica, que permiten consultar de forma manual sinónimos y nombres. Sin embargo, estas carecen de una interfaz de usuario que permita la consulta de un gran número de especies, acceder a ellas de forma automática o hacerlo desde entornos que permitan integrar los resultados en un flujo de trabajo más complejo. Por otro lado, existen herramientas que sí pueden consultarse a través de interfaces de programación de aplicaciones (API) desde entornos como R (R Core Team, 2025), facilitando la tarea de cotejar un gran número de especies a la vez así como su incorporación en nuestros flujos de trabajo. Algunos ejemplos de esto son el paquete *taxize* (Scott Chamberlain y Eduard Szocs, 2013; Chamberlain et al., 2020), con acceso a más de 20 bases de datos diferentes; o el paquete *WorldFlora* Kindt (2020) para acceder a *World Flora Online*, por citar algunas. La principal desventaja de estos servicios radica en que, a pesar de la gran cantidad de información que contienen, puede darse un desajuste entre su nomenclatura y la establecida a nivel nacional o regional. Esta limitación se hace más patente si, por ejemplo, queremos recopilar información sobre el estado de conservación de una especie presente en España, puesto que el nombre aceptado en una base de datos internacional puede no casar con el empleado en catálogos a nivel nacional o regional.

El [Inventario Español del Patrimonio Natural y de la Biodiversidad (IEPNB)](https://iepnb.gob.es/) supone un primer paso para solventar estas limitaciones a la hora de acceder a la información sobre las especies presentes en España. Este trabajo, todavía en proceso, recoge información de más de 63000 taxa, incluyendo su información taxonómica, sinonimia, categorías de amenaza o normativas legales que rigen su conservación, nutriéndose en gran medida de la Lista Patrón de Especies Silvestres presentes en España [(LP)](https://www.miteco.gob.es/es/biodiversidad/servicios/banco-datos-naturaleza/informacion-disponible/bdn_listas_patron.html#lista-patron-de-las-especies-silvestres-presentes-en-espana), que recoge información proveniente de multitud de sociedades científicas de ámbito nacional y especializadas en diversos grupos taxonómicos. A simple vista, la web del IEPNB sigue presentando las mismas limitaciones que otros recursos mencionados con anterioridad, como la imposibilidad de realizar consultas de forma no manual. Sin embargo, esta herramienta pone a nuestra disposición la API EIDOS, que nos permiten acceder a las bases de datos del IEPNB desde entornos de programación como R, y por tanto a automatizar los procesos de consulta.

# El paquete **eidosapi** para R

El paquete **eidosapi** se desarrolló para explotar las posibilidades ofrecidas por la API EIDOS y facilitar la tarea de cotejar y contrastar la taxonomía de múltiples especies al mismo tiempo desde el entorno de programación de R ([Figura 1](#fig-1)). El paquete cuenta con varias funciones que facilitan el acceso a las bases de datos del IEPNB, haciendo uso de del paquete **jsonlite** (Ooms, 2014) para interpretar los archivos en formato JSON que devuelve EIDOS al realizar consultas, así como del paquete **fuzzyjoin** (Robinson, 2025) para realizar consultas basadas en coincidencia difusa (*fuzzy matching*) que permitan buscar nombres en la LP aunque contengan algún error de escritura. A continuación se muestran varios ejemplos sencillos de uso, pero puede encontrase más información en el material suplementario.

|  |
| --- |
| Figura 1: Figura 1: Diagrama del flujo de trabajo de **eidosapi**. |

# Instalación y ejemplos básicos de uso de **eidosapi**

La instalación del paquete puede realizarse fácilmente clonando el repositorio disponible en GitHub (https://github.com/hmirceb/eidosapi).

# Instalación con remotes  
remotes::install\_github("https://github.com/hmirceb/eidosapi",  
 force = TRUE,   
 quiet = TRUE)  
# Cargamos el paquete  
library(eidosapi)

## Busqueda de especies por nombre

Un ejemplo básico sería buscar el nombre aceptado y sinónimos de una o varias especies. Para ellos vamos a emplear dos especies diferentes: la alondra ricotí (*Chersophilus duponti*), que no tiene sinónimos; y *Polygonum viviparum* que tiene varios. El procedimiento básico consiste en crear una tabla con el género y la especie de cada taxón. También podemos incluir una columna con la subespecie y la autoridad taxonómica que haya descrito el taxón. En el caso de que el taxón que nos interesa no tuviese subespecies o no conociésemos la autoría podemos omitir las columnas correspondientes o rellenarlas con *NA*. Puesto que el IEPNB todavía es un trabajo en marcha, algunos organismos no están presentes en la base de datos o carecen de información detallada. Para evitar errores al respecto, **eidosapi** ignora esos casos.

# Tabla ejemplo:  
taxa\_list = data.frame(genus = c("Chersophilus", "Polygonum", "Androsace"),  
 species = c("duponti", "viviparum", "cylindrica"),  
 subspecies = c(NA, NA, "hirtella"))  
eidos\_results = eidosapi::eidos\_taxon\_by\_name(  
 taxon\_list = taxa\_list  
 )  
  
# La tabla resultante tienes muchas columnas  
# Para facilitar la visualización se muestra solamente   
# el nombre que hemos buscado, su identificador,  
# si es un sinónimo o no y el id del taxón aceptado:  
knitr::kable(eidos\_results[c("supplied\_taxon",  
 "idtaxon", "nametype", "acceptednameid")])

| supplied\_taxon | idtaxon | nametype | acceptednameid |
| --- | --- | --- | --- |
| Chersophilus duponti | 11182 | Aceptado/válido | 11182 |
| Chersophilus duponti | 96736 | Aceptado/válido | 96736 |
| Polygonum viviparum | 7277 | Sinónimo | 32824 |
| Androsace cylindrica hirtella | 2146 | Aceptado/válido | 2146 |

## Estado de conservación

La columna *idtaxon* obtenida en el paso anterior contiene el identificador único para cada taxón de la base de datos. Si quisiéramos saber si la alondra ricotí tiene asociada alguna categoría de amenaza, solo tendríamos que introducir este indicador en la función eidos\_conservation\_by\_id.

# Buscamos el identificador por nombre:  
eidos\_results = eidosapi::eidos\_taxon\_by\_name(  
 taxon\_list = "Chersophilus duponti"  
 )  
  
# El identificador debería ser 11182:  
print(eidos\_results$idtaxon[1])

[1] 11182

# Accedemos a la información sobre su estado de conservación  
eidos\_cons = eidosapi::eidos\_conservation\_by\_id(  
 taxon\_id = eidos\_results$idtaxon  
 )  
  
# Mostramos solo algunas columnas básicas:  
knitr::kable(eidos\_cons[c("idtaxon", "anio", "categoriaconservacion", "aplicaa")])

| idtaxon | anio | categoriaconservacion | aplicaa |
| --- | --- | --- | --- |
| 11182 | 1992 | R (Rara) | España |
| 11182 | 2004 | EN (En peligro) | España |
| 11182 | 2016 | NT (Casi amenazado) | Mundial |
| 11182 | 2017 | NT (Casi amenazado) | Mundial |
| 11182 | 2020 | VU (Vulnerable) | Mundial |
| 11182 | 2021 | EN (En peligro) | Península |

En este caso, la alondra ricotí a nivel mundial pasó de “Casi amenazada” (NT) a “Vulnerable” (VU) en el año 2020, mientras que en España y la Península Ibérica se la considera “En peligro” (EN). Siguiendo este mismo procedimiento podríamos conocer las normativas específicas que rigen esas categoría para cada especie con la función eidos\_legal\_status\_by\_id.

## Búsqueda de especies con errores en la nomenclatura

Un problema común a la hora de trabajar con datos de especies son los errores de escritura. El paquete **eidosapi** incluye la función eidos\_fuzzy\_names que permite consultar en la LP, previa descarga con la función eidos\_clean\_checklist, los nombres que más se acerquen a la información que hayamos aportado. Podemos comprobarlo con algunos nombres muy mal escritos.

# Creamos la tabla con la información que queremos contrastar:  
taxa\_list = data.frame(genus = c("Vorderea", "Alytes"),  
 species = c("pyrenaica", "cisternasi"))  
  
# Obtendremos un error si no incluimos la LP como argumento:  
eidosapi::eidos\_fuzzy\_names(taxa\_list = taxa\_list)

Error in eidosapi::eidos\_fuzzy\_names(taxa\_list = taxa\_list): Checklist missing.  
 Please run eidos\_clean\_checklist() and  
 include result in argument checklist

# Descargarmos la LP y la guardamos en un objeto en el entorno de   
# trabajo de R:  
checklist = eidosapi::eidos\_clean\_checklist()

Downloading checklist and formatting, please wait...

# Ya podemos usar la función eidos\_fuzzy\_names:  
eidos\_result = eidosapi::eidos\_fuzzy\_names(taxa\_list = taxa\_list,   
 checklist = checklist)  
  
# Podemos comprobar que ha encontrado dos coincidencias:   
# Borderea pyrenaica y Alytes cisternasii:  
knitr::kable(eidos\_result[c("idtaxon", "taxon\_clean",   
 "ScientificName", "WithoutAutorship")])

| idtaxon | taxon\_clean | ScientificName | WithoutAutorship |
| --- | --- | --- | --- |
| 2700 | Borderea pyrenaica | Dioscorea pyrenaica Gren. | Dioscorea pyrenaica |
| 10909 | Alytes cisternasii | Alytes cisternasii Boscá, 1879 | Alytes cisternasii |

# Conclusiones

El paquete **eidosapi** permite acceder de forma sencilla a las bases de datos taxonómicas del Inventario del Patrimonio Natural y de la Biodiversidad, facilitando la estandarización de nombres científicos, obtener información detallada sobre sobre su estado de conservación y permitiéndonos incorporar esta información a otros flujos de trabajo.

# Contribución de los autores

Conceptualization, Software, Writing – original draft, Writing – review & editing: Héctor Miranda-Cebrián

# Agradecimientos

Esta nota se ha revisado siguiendo un proceso colaborativo y público disponible en (link al issue correspondiente).

# Referencias

Chamberlain, S., Szoecs, E., Foster, Z., Arendsee, Z., Boettiger, C., Ram, K., Bartomeus, I. et al. 2020. [*taxize: Taxonomic information from around the web*](https://github.com/ropensci/taxize).

Kindt, R. 2020. WorldFlora: An R package for exact and fuzzy matching of plant names against the World Flora Online taxonomic backbone data. *Applications in Plant Sciences* 8: e11388.

Ooms, J. 2014. [The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects](https://arxiv.org/abs/1403.2805). *arXiv:1403.2805 [stat.CO]*.

Patterson, D.J., Cooper, J., Kirk, P.M., Pyle, R.L., Remsen, D.P. 2010. [Names are key to the big new biology](https://doi.org/10.1016/j.tree.2010.09.004). *Trends in Ecology &amp; Evolution* 25: 686-691.

R Core Team. 2025. [*R: A Language and Environment for Statistical Computing*](https://www.R-project.org/). R Foundation for Statistical Computing, Vienna, Austria.

Robinson, D. 2025. [*fuzzyjoin: Join Tables Together on Inexact Matching*](https://doi.org/10.32614/CRAN.package.fuzzyjoin).

Scott Chamberlain, Eduard Szocs. 2013. [taxize - taxonomic search and retrieval in R](https://f1000research.com/articles/2-191/v2). *F1000Research*.