Package: rgsrs (via r-universe)

June 5, 2026

Title Query the FDA Global Substance Registration System (GSRS) API

Version 0.1.0

Description Provides functions to query the FDA Global Substance Registration System (GSRS) REST API (<https://gsrs.ncats.nih.gov/api/v1/>). Enables programmatic access to substance records, UNII identifiers, synonyms, external codes, and chemical structures for over 170,000 registered substances.

License MIT + file LICENSE

URL https://c1au6i0.github.io/rgsrs/, https://github.com/c1au6i0/rgsrs

BugReports https://github.com/c1au6i0/rgsrs/issues

Depends R (>= 4.1)

Imports cli, httr2, janitor, pingr

Suggests fs, openxlsx, spelling, testthat (>= 3.0.0), withr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData false

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/pak/sysreqs libicu-dev libssl-dev

Repository https://c1au6i0.r-universe.dev

Date/Publication 2026-05-05 20:06:46 UTC

RemoteUrl https://github.com/c1au6i0/rgsrs

RemoteRef HEAD

RemoteSha db72e8d5b3f8963d785df331734580091b11e85e

Contents

gsrs_all 2
gsrs_browse 3
gsrs_chem_info 4
gsrs_codes 6
gsrs_hierarchy 7
gsrs_names 8
gsrs_search 9
gsrs_structure 11
gsrs_structure_search 12
gsrs_substance 13
gsrs_unii_from_name 14
gsrs_vocabularies 15
write_dataframes_to_excel 16

Index **17**

`gsrs_all` *Retrieve comprehensive GSRS data for a set of UNIIs*

Description

Convenience wrapper that calls `gsrs_substance()`, `gsrs_names()`, `gsrs_codes()`, `gsrs_structure()`, and `gsrs_hierarchy()` in sequence and returns a named list containing all five data frames. Each sub-function uses `with_graceful_exit` internally, so partial failures return `NULL` for that element without aborting the whole call.

Usage

```
gsrs_all(unii, verbose = TRUE, delay = 0.5)
```

Arguments

`unii`	Character vector of one or more UNII codes.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups. Default `0.5`.

Value

A named list with five elements:

substance Data frame from `gsrs_substance()`.

names Data frame from `gsrs_names()`.

codes Data frame from `gsrs_codes()`.

structure Data frame from `gsrs_structure()`.

hierarchy Data frame from `gsrs_hierarchy()`.

Returns `NULL` on error (with a warning).

See Also

gsrs_substance(), gsrs_names(), gsrs_codes(), gsrs_structure(), gsrs_hierarchy()

Examples

```
Sys.sleep(2)
out <- gsrs_all("R16CO5Y76E")  # aspirin
if (!is.null(out)) {
  print(out$substance)
  print(head(out$names))
  print(head(out$codes))
  print(out$structure[, c("smiles", "formula", "mwt", "inchi_key")])
  print(out$hierarchy[, c("depth", "type", "approval_id", "name")])
}
```

---

gsrs_browse *Browse all substance records in GSRS*

Description

Retrieves a paginated list of all substance records from `GET /api/v1/substances`. Useful for bulk workflows or building a local catalogue. Use `top` and `skip` to page through the ~170,000 available records, or set `top = Inf` to fetch all (slow — use with care).

Usage

```
gsrs_browse(top = 10L, skip = 0L, verbose = TRUE, delay = 0.5)
```

Arguments

`top`	Integer. Maximum number of records to return per request. Default `10`. Set to `NULL` or `Inf` to fetch all records (paginates automatically; large result sets will be slow).
`skip`	Integer. Number of records to skip (offset). Default `0`.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between paginated requests when `top = Inf`. Default `0.5`.

Value

A data frame with the same columns as gsrs_search(). Returns `NULL` on error (with a warning).

See Also

gsrs_search(), gsrs_substance()

Examples

```
Sys.sleep(2)
# Fetch the first 5 substance records
out <- gsrs_browse(top = 5, verbose = FALSE)
if (!is.null(out)) print(out[, c("approval_id", "preferred_name",
                                 "substance_class")])
```

`gsrs_chem_info` *Retrieve chemical structure information by substance name or CAS number*

Description

A convenience wrapper that resolves one or more substance identifiers to GSRS UNIIs and then fetches the embedded chemical structure data for each substance. The result is one wide row per input identifier containing both the resolved metadata and the full structure record.

Usage

```
gsrs_chem_info(
  identifiers,
  type = c("name", "cas", "unii", "inchikey", "smiles"),
  verbose = TRUE,
  delay = 0.5
)
```

Arguments

`identifiers`	Character vector of substance identifiers.
`type`	Character scalar. The identifier type. One of:
	`"name"` Common or systematic substance name (default).
	`"cas"` CAS Registry Number (e.g., `"50-78-2"`).
	`"unii"` FDA UNII / approval ID (e.g., `"R16CO5Y76E"`). Skips the search step and fetches the structure directly.
	`"inchikey"` Standard InChIKey (e.g., `"BSYNRYMUTXBXSQ-UHFFFAOYSA-N"`).
	`"smiles"` SMILES string. Uses an exact structure search to resolve to a UNII before fetching the structure record.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual API calls. Default `0.5`.

Value

A data frame with one row per input identifier and columns:

query The identifier supplied by the caller.

type The identifier type (`"name"` or `"cas"`).

unii Resolved UNII / approval ID.

preferred_name Preferred display name in GSRS.

substance_class Substance class (e.g., `"chemical"`).

smiles Canonical SMILES string.

formula Molecular formula (e.g., `"C9H8O4"`).

mwt Molecular weight (numeric).

inchi_key Standard InChIKey.

inchi Full InChI string.

stereochemistry Stereochemistry descriptor.

optical_activity Optical activity descriptor.

charge Formal charge (integer).

stereo_centers Number of stereocenters.

defined_stereo Number of defined stereocenters.

ez_centers Number of E/Z double-bond stereocenters.

molfile MDL molfile as a string.

date_retrieved Date the structure response was received.

Unresolved identifiers or non-chemical substances produce a row of `NA`s with `query` and `type` set. Returns `NULL` on error (with a warning).

See Also

`gsrs_structure()`, `gsrs_unii_from_name()`, `gsrs_codes()`, `gsrs_structure_search()`

Examples

```
Sys.sleep(2)
out <- gsrs_chem_info(c("aspirin", "ibuprofen"), type = "name")
if (!is.null(out)) print(out[, c("query", "unii", "formula", "mwt")])

Sys.sleep(2)
out_cas <- gsrs_chem_info(c("50-78-2", "15687-27-1"), type = "cas")
if (!is.null(out_cas)) print(out_cas[, c("query", "unii", "formula", "mwt")])

Sys.sleep(2)
out_unii <- gsrs_chem_info("R16CO5Y76E", type = "unii")
if (!is.null(out_unii)) print(out_unii[, c("query", "formula", "mwt")])

Sys.sleep(2)
out_ik <- gsrs_chem_info("BSYNRYMUTXBXSQ-UHFFFAOYSA-N", type = "inchikey")
```

```
if (!is.null(out_ik)) print(out_ik[, c("query", "unii", "formula")])

Sys.sleep(2)
out_smi <- gsrs_chem_info("CC(=O)Oc1ccccc1C(=O)O", type = "smiles")
if (!is.null(out_smi)) print(out_smi[, c("query", "unii", "formula")])
```

`gsrs_codes` *Retrieve external codes and identifiers for GSRS substances*

Description

For each supplied UNII, calls `GET /api/v1/substances(<UNII>)/codes` and returns all registered cross-references as a tidy data frame. These include CAS numbers, PubChem CIDs, ChEMBL IDs, WHO-ATC codes, NDF-RT codes, DrugBank IDs, and many more.

Usage

```
gsrs_codes(unii, code_system = NULL, verbose = TRUE, delay = 0.5)
```

Arguments

`unii`	Character vector of one or more UNII codes.
`code_system`	Character vector of code systems to filter on (e.g., `c("CAS", "PUBCHEM")`). Case-insensitive matching. Pass `NULL` (default) to return all code systems.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups when `unii` has multiple entries. Default `0.5`.

Value

A data frame with columns:

code_system External database / code system name (e.g., `"CAS"`, `"PUBCHEM"`, `"ChEMBL"`, `"WHO-ATC"`).

code The identifier in that system.

type `"PRIMARY"` or `"ALTERNATIVE"`.

url URL to the external record (when available).

comments Additional context for the code (e.g., ATC path).

is_classification Logical; `TRUE` for classification codes.

uuid Internal GSRS UUID for the code record.

date_retrieved Date the response was received.

query The UNII supplied by the caller.

Returns `NULL` on error (with a warning).

See Also

gsrs_substance(), gsrs_names(), gsrs_search()

Examples

```
Sys.sleep(2)
# All codes for aspirin
out <- gsrs_codes("R16CO5Y76E")
if (!is.null(out)) print(head(out))

Sys.sleep(2)
# Only CAS and PubChem codes
out_cas <- gsrs_codes("R16CO5Y76E", code_system = c("CAS", "PUBCHEM"))
if (!is.null(out_cas)) print(out_cas)
```

---

gsrs_hierarchy *Retrieve the relationship hierarchy for GSRS substances*

Description

For each supplied UNII, calls `GET /api/v1/substances(<UNII>)/@hierarchy` and returns the flat parent/child relationship tree as a tidy data frame. This is useful for navigating relationships such as salt forms to free base, active metabolites, or component substances.

Usage

```
gsrs_hierarchy(unii, verbose = TRUE, delay = 0.5)
```

Arguments

`unii`	Character vector of one or more UNII codes.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups when `unii` has multiple entries. Default `0.5`.

Value

A data frame with columns:

node_id Node identifier within the hierarchy tree (string index).

parent_id Parent node identifier (`"#"` for root nodes).

depth Depth in the tree (0 = root).

type Node type (e.g., `"ROOT"`, `"ACTIVE MOIETY"`, `"SALT/SOLVATE"`).

text Human-readable label including UNII and name.

expandable Logical; `TRUE` if node has children.

approval_id UNII of the substance at this node.

name Preferred name at this node.

ref_uuid Internal GSRS UUID of the related substance.

substance_class Substance class at this node.

deprecated Logical; `TRUE` if the node substance is deprecated.

date_retrieved Date the response was received.

query The UNII supplied by the caller.

Returns `NULL` on error (with a warning).

See Also

`gsrs_substance()`, `gsrs_all()`

Examples

```
Sys.sleep(2)
out <- gsrs_hierarchy("R16CO5Y76E")  # aspirin
if (!is.null(out)) print(out[, c("depth", "type", "approval_id", "name")])
```

`gsrs_names` *Retrieve all names (synonyms) for GSRS substances*

Description

For each supplied UNII, calls `GET /api/v1/substances(<UNII>)/names` and returns every registered name record as a tidy data frame row.

Usage

```
gsrs_names(unii, verbose = TRUE, delay = 0.5)
```

Arguments

`unii`	Character vector of one or more UNII codes.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups when `unii` has multiple entries. Default `0.5`.

Value

A data frame with columns:

name The name string.

std_name Standardised (uppercased) name.

type Name type code (e.g., `"bn"` brand name, `"cn"` common name, `"sys"` systematic name, `"of"` official name).

preferred Logical; `TRUE` when this is the preferred name.

display_name Logical; `TRUE` when this name is shown by default.

languages Semicolon-separated language codes.

domains Semicolon-separated domain tags.

uuid Internal GSRS UUID for the name record.

date_retrieved Date the response was received.

query The UNII supplied by the caller.

Returns `NULL` on error (with a warning).

See Also

`gsrs_substance()`, `gsrs_codes()`, `gsrs_search()`

Examples

```
Sys.sleep(2)
out <- gsrs_names("R16CO5Y76E")  # aspirin
if (!is.null(out)) print(head(out))
```

`gsrs_search` *Search the GSRS substance database*

Description

Searches the FDA Global Substance Registration System (GSRS) using a free-text or Lucene-style field query. Returns a tidy data frame of matching substance records with key metadata fields.

Usage

```
gsrs_search(query, top = 10L, skip = 0L, verbose = TRUE, delay = 0.5)
```

Arguments

query	Character string. The search query. Supports: • Free text (e.g., "aspirin") • Lucene field syntax (e.g., "root_names:aspirin", "root_approvalID:R16CO5Y76E") • Wildcards (*, ?) as per GSRS documentation.
top	Integer. Maximum number of records to return per request. Default 10. Use NULL or Inf to attempt to retrieve all records (paginates automatically; large result sets may be slow).
skip	Integer. Number of records to skip (offset). Default 0.
verbose	Logical. If TRUE, emit progress messages. Default TRUE.
delay	Numeric. Seconds to wait between paginated requests. Default 0.5.

Value

A data frame with columns:

uuid Internal GSRS UUID of the substance.

approval_id FDA UNII / approval ID.

preferred_name Preferred display name.

substance_class Substance class (e.g., "chemical", "structurallyDiverse").

status Record status (e.g., "approved").

definition_type "PRIMARY" or "ALTERNATIVE".

definition_level "COMPLETE" or "INCOMPLETE".

version Record version string.

names_url URL to retrieve all names for this substance.

codes_url URL to retrieve all codes for this substance.

self_url Full URL for this substance record.

date_retrieved Date the response was received from the server.

Returns NULL on error (with a warning).

See Also

gsrs_substance(), gsrs_names(), gsrs_codes()

Examples

```
Sys.sleep(2)
out <- gsrs_search("aspirin", top = 5)
if (!is.null(out)) print(head(out))
```

`gsrs_structure` *Retrieve chemical structure data for GSRS substances*

Description

For each supplied UNII, fetches the full substance record from `GET /api/v1/substances(<UNII>)` and extracts the embedded `structure` object, returning chemical identifiers and properties as a tidy data frame.

Usage

```
gsrs_structure(unii, verbose = TRUE, delay = 0.5)
```

Arguments

`unii`	Character vector of one or more UNII codes.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups when `unii` has multiple entries. Default `0.5`.

Value

A data frame with columns:

smiles Canonical SMILES string.

formula Molecular formula (e.g., `"C9H8O4"`).

mwt Molecular weight (numeric).

inchi_key Standard InChIKey.

inchi Full InChI string.

stereochemistry Stereochemistry descriptor (e.g., `"ACHIRAL"`, `"RACEMIC"`, `"ABSOLUTE"`).

optical_activity Optical activity (e.g., `"UNSPECIFIED"`, `"(+)"`, `"(-)"`).

charge Formal charge (integer).

stereo_centers Number of stereocenters.

defined_stereo Number of defined stereocenters.

ez_centers Number of E/Z double-bond stereocenters.

molfile MDL molfile as a string.

date_retrieved Date the response was received.

query The UNII supplied by the caller.

Non-chemical substances (proteins, polymers, etc.) return a row of `NA`s with `query` set. Returns `NULL` on error (with a warning).

See Also

gsrs_substance(), gsrs_structure_search(), gsrs_names(), gsrs_codes()

Examples

```
Sys.sleep(2)
out <- gsrs_structure("R16CO5Y76E")  # aspirin
if (!is.null(out)) print(out[, c("smiles", "formula", "mwt", "inchi_key")])
```

---

gsrs_structure_search *Search GSRS by chemical structure*

Description

Searches the FDA Global Substance Registration System for substances matching a chemical structure query supplied as a SMILES string. Supports substructure, similarity, exact-match, and flexible (disconnected moiety) search types.

Usage

```
gsrs_structure_search(
  smiles,
  type = c("sub", "sim", "exact", "flex"),
  cutoff = 0.8,
  top = 10L,
  verbose = TRUE
)
```

Arguments

smiles	Character string. A valid SMILES or SMARTS string describing the query structure (e.g., "CC(=O)Oc1ccccc1C(=O)O" for aspirin).
type	Character string. Search type. One of: "sub" Substructure search (default). Returns all substances whose structure contains the query as a substructure. "sim" Similarity search. Returns substances with Tanimoto similarity >= cutoff. Use cutoff to control threshold. "exact" Exact structure match (tautomer-aware, stereo-sensitive). "flex" Flexible (disconnected moiety) search; stereo-insensitive.
cutoff	Numeric in [0, 1]. Tanimoto similarity cutoff for type = "sim". Default 0.8. Ignored for other search types.
top	Integer. Maximum number of records to return. Default 10.
verbose	Logical. If TRUE, emit progress messages. Default TRUE.

Value

A data frame with the same columns as `gsrs_search()`, plus a `query_smiles` column recording the input SMILES. Returns `NULL` on error (with a warning).

See Also

`gsrs_structure()`, `gsrs_search()`

Examples

```
Sys.sleep(2)
# Exact match for aspirin
out <- gsrs_structure_search("CC(=O)Oc1ccccc1C(=O)O", type = "exact")
if (!is.null(out)) print(out[, c("approval_id", "preferred_name")])

Sys.sleep(2)
# Similarity search
out_sim <- gsrs_structure_search("CC(=O)Oc1ccccc1C(=O)O",
                                 type = "sim", cutoff = 0.7, top = 5)
if (!is.null(out_sim)) print(out_sim[, c("approval_id", "preferred_name")])
```

`gsrs_substance` *Fetch a GSRS substance record by UNII*

Description

Retrieves the top-level metadata for a single substance identified by its UNII (Unique Ingredient Identifier / approval ID). Internally this performs a filtered search using `root_approvalID:<unii>`.

Usage

```
gsrs_substance(unii, verbose = TRUE, delay = 0.5)
```

Arguments

`unii`	Character vector of one or more UNII codes (e.g., `"R16CO5Y76E"` for aspirin).
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups when `unii` has multiple entries. Default `0.5`.

Value

A data frame with the same columns as `gsrs_search()`, with one row per input UNII. Rows for unrecognised UNIIs will contain `NA` except for the `query` column (which is always set to the input UNII). Returns `NULL` on error (with a warning).

See Also

gsrs_search(), gsrs_names(), gsrs_codes()

Examples

```
Sys.sleep(2)
out <- gsrs_substance("R16CO5Y76E")  # aspirin
if (!is.null(out)) print(out)
```

`gsrs_unii_from_name` *Look up UNII codes for substance names*

Description

For each supplied name, queries GSRS using `root_names:<name>` and returns the best-matching UNII together with the preferred substance name and substance class. This is useful for converting common or systematic names to the canonical FDA UNII identifier.

Usage

```
gsrs_unii_from_name(names, top = 1L, verbose = TRUE, delay = 0.5)
```

Arguments

`names`	Character vector of substance names to resolve.
`top`	Integer. Maximum number of candidate records to consider per name query. Default 1 returns only the top hit. Increase to inspect multiple candidates.
`verbose`	Logical. If `TRUE`, emit progress messages. Default `TRUE`.
`delay`	Numeric. Seconds to wait between individual lookups. Default `0.5`.

Value

A data frame with columns:

unii The UNII / approval ID of the matched substance.

preferred_name Preferred display name in GSRS.

substance_class Substance class (e.g., `"chemical"`).

status Record status.

uuid Internal GSRS UUID.

date_retrieved Date the response was received.

query The name supplied by the caller.

Unresolved names produce a row of `NA`s with `query` set. Returns `NULL` on error (with a warning).

See Also

gsrs_substance(), gsrs_search(), gsrs_names()

Examples

```
Sys.sleep(2)
out <- gsrs_unii_from_name(c("aspirin", "ibuprofen"))
if (!is.null(out)) print(out)
```

---

gsrs_vocabularies *Retrieve controlled vocabulary terms from GSRS*

Description

Fetches all (or a page of) controlled vocabulary entries from GET /api/v1/vocabularies. The result is one row per vocabulary term, with the parent domain and type attached to every row. This is useful for understanding allowed values for fields such as name type, substance class, relationship type, code system, and more.

Usage

```
gsrs_vocabularies(top = NULL, verbose = TRUE, delay = 0.5)
```

Arguments

top	Integer. Maximum number of vocabulary *domains* to return per request. Default NULL fetches all domains (paginates automatically).
verbose	Logical. If TRUE, emit progress messages. Default TRUE.
delay	Numeric. Seconds to wait between paginated requests. Default 0.5.

Value

A data frame with columns:

domain Vocabulary domain name (e.g., "NAME_TYPE", "SUBSTANCE_CLASS", "RELATIONSHIP_TYPE").

term_type Vocabulary term type identifier.

editable Logical; TRUE if the vocabulary can be extended.

filterable Logical; TRUE if the vocabulary supports filtering.

value The controlled term value (used in the API/data).

display Human-readable display label for the term.

hidden Logical; TRUE if the term is hidden from the UI.

selected Logical; TRUE if the term is selected by default.

date_retrieved Date the response was received.

Returns NULL on error (with a warning).

See Also

gsrs_search(), gsrs_codes()

Examples

```
Sys.sleep(2)
vocab <- gsrs_vocabularies(verbose = FALSE)
if (!is.null(vocab)) {
  # See all name type values
  print(vocab[vocab$domain == "NAME_TYPE", c("value", "display")])
}
```

`write_dataframes_to_excel` *Write a named list of data frames to an Excel workbook*

Description

Each element of `df_list` is written to its own sheet. Requires the `openxlsx` package (listed in `Suggests`).

Usage

```
write_dataframes_to_excel(df_list, filename)
```

Arguments

`df_list`	A named list of data frames.
`filename`	Character string. Path to the output `.xlsx` file.

Value

Invisible `filename`.

Examples

```
tmp <- tempfile(fileext = ".xlsx")
write_dataframes_to_excel(list(sheet1 = mtcars, sheet2 = iris), tmp)
```

Index

gsrs_all, 2
gsrs_all(), *8*
gsrs_browse, 3
gsrs_chem_info, 4
gsrs_codes, 6
gsrs_codes(), *2, 3*, *5*, *9, 10*, *12*, *14*, *16*
gsrs_hierarchy, 7
gsrs_hierarchy(), *2, 3*
gsrs_names, 8
gsrs_names(), *2, 3*, *7*, *10*, *12*, *14, 15*
gsrs_search, 9
gsrs_search(), *3*, *7*, *9*, *13–16*
gsrs_structure, 11
gsrs_structure(), *2, 3*, *5*, *13*
gsrs_structure_search, 12
gsrs_structure_search(), *5*, *12*
gsrs_substance, 13
gsrs_substance(), *2, 3*, *7–10*, *12*, *15*
gsrs_unii_from_name, 14
gsrs_unii_from_name(), *5*
gsrs_vocabularies, 15

write_dataframes_to_excel, 16