Package: GalaxyR (via r-universe)

November 3, 2026

Title 'Galaxy' API Implementation

Version 0.1.2

Description On 'Galaxy' platforms like 'Galaxy Europe' <https://usegalaxy.eu>, many tools and workflows can run directly on a high-performance computer. 'GalaxyR' connects R with 'Galaxy' platforms API <https://usegalaxy.eu/api/docs> and allows credential management, uploading data, invoking workflows or tools, checking their status, and downloading results.

URL https://github.com/JulFrey/GalaxyR

BugReports https://github.com/JulFrey/GalaxyR/issues

License GPL-3

Encoding UTF-8

Imports httr, jsonlite, methods

Roxygen list(markdown = TRUE)

Depends R (>= 4.1.0)

Config/roxygen2/version 8.0.0

RoxygenNote 7.3.3

Config/pak/sysreqs libssl-dev

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

Date/Publication 2026-07-06 09:19:22 UTC

RemoteUrl https://github.com/julfrey/galaxyr

RemoteRef HEAD

RemoteSha 23af5575091f01359d46df2b8e076b937cd9eac6

Contents

galaxy 2
Galaxy-class 3

galaxy_delete_dataset . . . 3
galaxy_delete_datasets . . . 4
galaxy_delete_history . . . 5
galaxy_download_result . . . 7
galaxy_download_rocrate . . . 8
galaxy_get_file_info . . . 9
galaxy_get_tool . . . 10
galaxy_get_tool_id . . . 11
galaxy_get_workflow . . . 12
galaxy_get_workflow_inputs . . . 12
galaxy_has_key . . . 13
galaxy_history_size . . . 14
galaxy_initialize . . . 15
galaxy_list_files . . . 16
galaxy_list_histories . . . 18
galaxy_list_invocations . . . 18
galaxy_list_tools . . . 19
galaxy_list_workflows . . . 20
galaxy_poll_tool . . . 21
galaxy_poll_workflow . . . 22
galaxy_print_tool_inputs . . . 23
galaxy_run_tool . . . 23
galaxy_set_credentials . . . 24
galaxy_start_workflow . . . 26
galaxy_upload_ftp . . . 27
galaxy_upload_https . . . 28

Index **31**

`galaxy` *Create a Galaxy session object*

Description

Constructor for a `Galaxy` S4 object used for pipe-based workflows. The returned object carries identifiers such as `history_id`, `input_dataset_id` and `invocation_id` through subsequent calls.

Usage

```
galaxy(history_name = "R API request", galaxy_url = "https://usegalaxy.eu")
```

Arguments

`history_name`	Character. Default name to give to a new history, stored in the object and used by `galaxy_initialize()` if you don't override it.
`galaxy_url`	Character. Base URL of the Galaxy instance. If the environment variable `GALAXY_URL` is set, it takes precedence.

Value

A Galaxy object in state "new".

Galaxy-class *Galaxy session object*

Description

An S4 class used to carry state across a pipe-based workflow against a Galaxy instance.

Slots

history_name Default name to give to a new history.

history_id Encoded ID of the history on the server.

input_dataset_id Encoded ID of the last uploaded input dataset.

inputs A list of tool/workflow inputs to be applied on the next call.

invocation_id Encoded ID of the last workflow invocation.

output_dataset_ids Character vector of encoded output dataset IDs.

state One of "new", "pending", "success" or "error".

galaxy_url Base URL of the Galaxy instance.

---

galaxy_delete_dataset *Delete a Galaxy dataset by ID*

Description

Delete a dataset (HDA) from a Galaxy instance using the Galaxy API.

Usage

```
galaxy_delete_dataset(
  dataset_id,
  purge = TRUE,
  verbose = FALSE,
  galaxy_url = "https://usegalaxy.eu"
)
```

Arguments

dataset_id	Character. The Galaxy dataset ID to delete.
purge	Logical. If TRUE the API call will include purge=true to permanently remove the dataset and free space. If FALSE the dataset may be only soft-deleted depending on Galaxy configuration. Default: TRUE.
verbose	Logical. If TRUE a message with the HTTP status code will be printed. Default: TRUE.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Details

This function performs an HTTP DELETE against the Galaxy /api/datasets/id endpoint. By default it requests a purge (permanent removal) by adding ?purge=true. The Galaxy API key is read from the environment variable GALAXY_API_KEY.

- Make sure Sys.getenv("GALAXY_API_KEY") is set to a valid API key..
- Use caution when running with purge = TRUE as this permanently removes data.

Value

A named list with elements:

success Logical. TRUE for 2xx responses, otherwise FALSE.

status Integer. HTTP status code returned by the API.

content Character. The raw response body (text).

Examples

```
input_file <- tempfile(fileext = ".txt")
test_text <- "This is an example \nfile."
writeLines(test_text,input_file)
history_id <- galaxy_initialize()
dataset_id <- galaxy_upload_https(input_file, history_id)

galaxy_delete_dataset(dataset_id)
```

---

galaxy_delete_datasets

Delete multiple Galaxy datasets by ID

Description

Convenience wrapper that deletes a vector of dataset IDs using galaxy_delete_dataset. Requests are paced with a small sleep between calls to avoid overwhelming the server.

Usage

```
galaxy_delete_datasets(
  output_ids,
  purge = TRUE,
  sleep = 0.2,
  galaxy_url = "https://usegalaxy.eu"
)
```

Arguments

output_ids	Character vector of dataset IDs to delete.
purge	Logical. Passed to galaxy_delete_dataset. Default: TRUE.
sleep	Numeric. Seconds to wait between API calls. Default: 0.2.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Value

A named list where each element is the return value from galaxy_delete_dataset for the corresponding dataset ID.

Examples

```
input_file <- tempfile(fileext = ".txt")
input_file2 <- tempfile(fileext = ".txt")
test_text <- "This is an example \nfile."
writeLines(test_text,input_file)
writeLines(test_text,input_file2)
history_id <- galaxy_initialize("test upload")
dataset_id <- galaxy_upload_https(input_file, history_id)
dataset_id2 <- galaxy_upload_https(input_file2, history_id)

galaxy_delete_datasets(list(output_ids = c(dataset_id, dataset_id2)))
```

---

galaxy_delete_history *Generic for deleting a Galaxy history*

Description

galaxy_delete_history() is an S4 generic. With x as a character scalar, it is treated as a history ID and the function returns API response metadata. With x as a Galaxy object, history_id and galaxy_url are taken from the object and the object state is updated.

Usage

```
galaxy_delete_history(
  x,
  purge = FALSE,
  galaxy_url = "https://usegalaxy.eu",
  verbose = FALSE
)

## S4 method for signature 'character'
galaxy_delete_history(
  x,
  purge = FALSE,
  galaxy_url = "https://usegalaxy.eu",
  verbose = FALSE
)

## S4 method for signature 'Galaxy'
galaxy_delete_history(x, purge = FALSE, verbose = FALSE)
```

Arguments

x	A history ID (character) or a Galaxy object.
purge	Logical. If TRUE, request permanent deletion (purge). Default: FALSE.
galaxy_url	Character. Base URL of the Galaxy instance, used by the character method. If GALAXY_URL is set, it takes precedence.
verbose	Logical. If TRUE, print request status. Default: FALSE.

Details

The function calls DELETE /api/histories/{history_id} with optional purge=TRUE to request permanent removal (server policy permitting).

Value

- For the character method: a named list with success, status, content.
- For the Galaxy method: the modified Galaxy object (state set to "success" on 2xx, otherwise "error").

Examples

```
## Not run:
# Character method
hid <- galaxy_initialize("history to delete")
galaxy_delete_history(hid, purge = FALSE)

# Galaxy method
g <- galaxy(history_name = "history to delete")
g <- galaxy_initialize(g)
```

```
g <- galaxy_delete_history(g, purge = FALSE)

## End(Not run)
```

`galaxy_download_result` *Generic for downloading files from a history*

Description

`galaxy_download_result()` is an S4 generic. With `x` as a character vector of HDA output IDs, all corresponding datasets are downloaded into `out_dir` using their Galaxy names; duplicate names are disambiguated by appending `_<i>` before the extension. Existing files are not overwritten if `overwrite = FALSE`, and a warning is issued when a name is adjusted. With `x` as a `Galaxy` object its `output_dataset_ids` and `galaxy_url` are used; the object is returned invisibly after performing the downloads.

Usage

```
galaxy_download_result(
  x,
  out_dir = ".",
  galaxy_url = "https://usegalaxy.eu",
  overwrite = FALSE
)

## S4 method for signature 'character'
galaxy_download_result(
  x,
  out_dir = ".",
  galaxy_url = "https://usegalaxy.eu",
  overwrite = FALSE
)

## S4 method for signature 'Galaxy'
galaxy_download_result(x, out_dir = ".", overwrite = FALSE)
```

Arguments

`x`	A vector of HDA output IDs (`character`), or a `Galaxy` object.
`out_dir`	Directory in which to save the downloaded files.
`galaxy_url`	Base URL of the Galaxy instance, used by the character method.
`overwrite`	Logical; if `FALSE` (default), do not overwrite existing files but choose unique names instead.

Value

For the character method, a list of `httr` responses; for the `Galaxy` method, the (unchanged) `Galaxy` object invisibly.

`galaxy_download_rocrate`

Generic for downloading a history as an RO-Crate

Description

`galaxy_download_rocrate()` is an S4 generic. With `x` as a history ID (`character`) it requests an export in RO-Crate format, polls until ready, and downloads the archive to `dest_file`. With `x` as a `Galaxy` object, its `history_id` and `galaxy_url` are used and the object is returned invisibly after performing the download.

Usage

```
galaxy_download_rocrate(
  x,
  dest_file = tempfile(fileext = ".zip"),
  galaxy_url = "https://usegalaxy.eu",
  format = "rocrate.zip",
  poll_interval = 5,
  timeout = 600
)

## S4 method for signature 'character'
galaxy_download_rocrate(
  x,
  dest_file = tempfile(fileext = ".zip"),
  galaxy_url = "https://usegalaxy.eu",
  format = "rocrate.zip",
  poll_interval = 30,
  timeout = 600
)

## S4 method for signature 'Galaxy'
galaxy_download_rocrate(
  x,
  dest_file = tempfile(fileext = ".zip"),
  format = format,
  poll_interval = 5,
  timeout = 600
)
```

Arguments

x	A history ID (character), or a Galaxy object.
dest_file	Path to save the downloaded RO-Crate (defaults to a temporary .zip file).
galaxy_url	Base URL of the Galaxy instance, used by the character method. If GALAXY_URL is set it takes precedence.
format	Format for the history export. Possible formats depend on the Galaxy server. Typical inputs are 'tgz', 'tar', 'tar.gz', 'bag.zip', 'bag.tar', 'bag.tgz', 'rocrate.zip' or 'bco.json'. Defaults to 'rocrate.zip'.
poll_interval	Seconds between status checks.
timeout	Maximum time to wait in seconds before giving up.

Value

For the character method, the path to the downloaded file. For the Galaxy method, the (unchanged) Galaxy object invisibly.

Examples

```
hid <- "0123456789abcdef"
crate <- galaxy_download_rocrate(hid, dest_file = "history_rocrate.zip")
g <- galaxy()
g <- galaxy_initialize(g)
g <- galaxy_download_rocrate(g, dest_file = "history_rocrate.zip")
```

---

galaxy_get_file_info *Get information for one or more Galaxy datasets*

Description

Retrieves metadata for one or more Galaxy history datasets (HDAs), including name, size, type, state, and deletion status.

Usage

```
galaxy_get_file_info(file_ids, galaxy_url = "https://usegalaxy.eu")
```

Arguments

file_ids	Character vector of Galaxy dataset IDs.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Details

This function queries the /api/datasets/{id} endpoint for each provided dataset ID. If a dataset cannot be retrieved, its fields are returned as NA.

Value

A data.frame with one row per dataset and the columns: `id`, `name`, `size_bytes`, `human_size`, `file_type`, `state`, `deleted`, `stringsAsFactors`.

Examples

```
tmp_dir <- tempdir()
f_name <- "iris.csv"
f_path <- paste(tmp_dir, f_name, sep = "\\")
write.csv(datasets::iris, f_path, row.names = FALSE)

history_id <- galaxy_initialize("IRIS")
file_id <- galaxy_upload_https(f_path, history_id)
galaxy_get_file_info(file_id)
```

---

galaxy_get_tool *Retrieve detailed metadata for a Galaxy tool*

Description

Retrieve detailed metadata for a Galaxy tool

Usage

```
galaxy_get_tool(
  tool_id,
  galaxy_url = "https://usegalaxy.eu",
  tool_version = NULL
)
```

Arguments

`tool_id`	Character. The Galaxy tool ID (for example `"toolshed.g2.bx.psu.edu/repos/devteam/fastqc/fast`
`galaxy_url`	Character. Base URL of the Galaxy instance (for example `"https://usegalaxy.eu"`). If the environment variable `GALAXY_URL` is set, it takes precedence.
`tool_version`	Optional character string to request a specific version. If `NULL`, Galaxy will return the default/latest version metadata.

Value

A list containing the tool metadata as returned by the Galaxy API (inputs, outputs, help text, etc.).

Examples

```
tool_id <- galaxy_get_tool_id("FastQC")[1]
fastqc_tool <- galaxy_get_tool(tool_id)
fastqc_tool$description
```

`galaxy_get_tool_id` *Retrieve Galaxy tool IDs by name*

Description

Retrieve Galaxy tool IDs by name

Usage

```
galaxy_get_tool_id(
  name,
  tools = NULL,
  ignore_case = TRUE,
  galaxy_url = "https://usegalaxy.eu",
  panel_id = NULL
)
```

Arguments

`name`	Character string to search for in tool names.
`tools`	Optional list as returned by `galaxy_list_tools`. If `NULL`, the function will fetch tools on the fly by calling `galaxy_list_tools`.
`ignore_case`	Logical. Whether matching should ignore case. Default: `TRUE`.
`galaxy_url`	Character. Base URL of the Galaxy instance (for example `"https://usegalaxy.eu"`). If the environment variable `GALAXY_URL` is set, it takes precedence.
`panel_id`	Optional character. Passed through to `galaxy_list_tools` when `tools` is `NULL` so you can restrict the search to a panel/section.

Value

Character vector of matching tool IDs in decreasing order (usually highest version first). Returns `character(0)` if no tools match.

Examples

```
# Fetch the full tool list once, then lookup
tools <- galaxy_list_tools()
galaxy_get_tool_id("FastQC", tools = tools)

# Or let the helper fetch on demand
galaxy_get_tool_id("FastQC")

# Exact, case-sensitive match inside a specific panel
galaxy_get_tool_id("Concatenate datasets",
ignore_case = FALSE, panel_id = "Text Manipulation")
```

galaxy_get_workflow *Receive workflow metadata from the API*

Description

Receive workflow metadata from the API

Usage

```
galaxy_get_workflow(workflow_id, galaxy_url = "https://usegalaxy.eu")
```

Arguments

workflow_id	Character. Galaxy workflow ID.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Value

a structured list with all metadata

Examples

```
## Not run:
galaxy_get_workflow("f2db41e1fa331b3e")

## End(Not run)
```

galaxy_get_workflow_inputs *Retrieve input definitions for a Galaxy workflow*

Description

Retrieves and summarizes the input steps required by a Galaxy workflow.

Usage

```
galaxy_get_workflow_inputs(workflow_id, galaxy_url = "https://usegalaxy.eu")
```

Arguments

workflow_id	Character. Galaxy workflow ID.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Details

This function queries `/api/workflows/{workflow_id}` and extracts workflow input steps (data and parameter inputs). The returned `step_id` values must be used as names in the `inputs` argument of `galaxy_start_workflow`.

Value

A data.frame with one row per workflow input and the columns: `step_id`, `name`, `type`, `optional`, `default`.

Examples

```
## Not run:
galaxy_get_workflow_inputs("f2db41e1fa331b3e")

## End(Not run)
```

`galaxy_has_key` *Check whether a Galaxy API key is available*

Description

Check whether the environment variable `GALAXY_API_KEY` is set and non-empty.

Usage

```
galaxy_has_key()
```

Value

Logical. `TRUE` if an API key is available, otherwise `FALSE`.

Examples

```
galaxy_has_key() # returns true if api key is set
```

galaxy_history_size *Galaxy history size Get the disk usage / size of a Galaxy history*

Description

The function first tries to read a size/disk_usage field from the history summary endpoint. If that is not present it fetches the history contents and sums dataset sizes (robust to a few different field names used by different Galaxy versions). Results are returned as a data.frame with bytes and a human-readable size.

Usage

```
galaxy_history_size(
  history_id,
  galaxy_url = "https://usegalaxy.eu",
  include_deleted = FALSE
)
```

Arguments

history_id	Galaxy history ID.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.
include_deleted	Logical; whether to include deleted datasets when summing (default FALSE)

Value

data.frame with columns history_id, bytes, human_size

Examples

```
histories <- galaxy_list_histories()
if(nrow(histories > 0)){
  galaxy_history_size(histories$history_id[1])
} else {
  message("No histories found for current user.")
}
```

`galaxy_initialize` *Create a new Galaxy history*

Description

`galaxy_initialize()` is an S4 generic. With no `x` supplied it creates a new history on the given Galaxy instance and returns its encoded ID. When called with a `Galaxy` object it uses the object's `history_name` and `galaxy_url`, creates the history, and updates the object with the new `history_id` and state `"pending"`.

Usage

```
galaxy_initialize(
  x,
  name = "R API request",
  galaxy_url = "https://usegalaxy.eu"
)

## S4 method for signature 'missing'
galaxy_initialize(name, galaxy_url)

## S4 method for signature 'Galaxy'
galaxy_initialize(x)
```

Arguments

`x`	A `Galaxy` object, or missing to use the default method.
`name`	Name of the history to create. Ignored when `x` is a `Galaxy`, in which case `x@history_name` is used.
`galaxy_url`	Base URL of the Galaxy instance. Ignored when `x` is a `Galaxy`, in which case `x@galaxy_url` is used.

Details

A valid Galaxy API key is required and must be available via the `GALAXY_API_KEY` environment variable.

Value

For the default method (`x` missing), a character scalar history ID. For the `Galaxy` method, the modified `Galaxy` object.

Examples

```
history_id <- galaxy_initialize("My history name")
g <- galaxy(history_name = "My history name")
g <- galaxy_initialize(g)
```

galaxy_list_files *List all files (datasets) in a Galaxy history*

Description

galaxy_list_files() is an S4 generic. With x as a character scalar, it is treated as a Galaxy history ID and returns a data frame of history datasets (files). With x as a Galaxy object, the history ID and URL are taken from the object.

Usage

```
galaxy_list_files(
  x,
  include_deleted = FALSE,
  include_hidden = FALSE,
  galaxy_url = "https://usegalaxy.eu",
  limit = 500L
)

## S4 method for signature 'character'
galaxy_list_files(
  x,
  include_deleted = FALSE,
  include_hidden = FALSE,
  galaxy_url = "https://usegalaxy.eu",
  limit = 500L
)

## S4 method for signature 'Galaxy'
galaxy_list_files(
  x,
  include_deleted = FALSE,
  include_hidden = FALSE,
  limit = 500L
)
```

Arguments

x	A history ID (character) or a Galaxy object.
include_deleted	Logical; if TRUE, include deleted datasets. Default: FALSE.
include_hidden	Logical; if TRUE, include hidden datasets. Default: FALSE.
galaxy_url	Character. Base URL of the Galaxy instance, used by the character method. If GALAXY_URL is set, it takes precedence.
limit	Integer page size for API pagination. Default: 500L.

Details

The function queries `/api/histories/{history_id}/contents` using pagination and returns one row per history content item, restricted to datasets (`history_content_type == "dataset"` when available).

Value

A `data.frame` with one row per dataset and columns:

id Encoded dataset ID.

name Dataset name.

history_id Encoded history ID.

history_content_type Usually `"dataset"`.

type Galaxy item type (if provided by server).

state Dataset state (e.g. `"ok"`, `"running"`, `"error"`).

deleted Logical deletion flag.

hidden Logical hidden flag.

file_ext Datatype extension (if available).

file_size Dataset size in bytes (if available).

create_time Creation timestamp (if available).

update_time Update timestamp (if available).

If no datasets are found, an empty data frame with these columns is returned.

Examples

```
# Character method (history_id)
hid <- galaxy_initialize("List files example")
files_df <- galaxy_list_files(hid)
head(files_df)

# Galaxy-object method
g <- galaxy(history_name = "List files example")
g <- galaxy_initialize(g)
files_df2 <- galaxy_list_files(g)
head(files_df2)
```

galaxy_list_histories *List Galaxy histories (name and history id)*

Description

List Galaxy histories (name and history id)

Usage

```
galaxy_list_histories(galaxy_url = "https://usegalaxy.eu")
```

Arguments

galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Value

data.frame with columns: history_name, history_id

Examples

```
histories <- galaxy_list_histories()
```

galaxy_list_invocations *List workflow invocations for a given workflow*

Description

List workflow invocations for a given workflow

Usage

```
galaxy_list_invocations(workflow_id, galaxy_url = "https://usegalaxy.eu")
```

Arguments

workflow_id	The Galaxy workflow ID to list invocations for.
galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.

Value

data.frame with columns: invocation_id, workflow_id, history_id, state, create_time, update_time, stringsAsFactors

galaxy_list_tools *List tools installed on a Galaxy instance*

Description

List tools installed on a Galaxy instance

Usage

```
galaxy_list_tools(
  galaxy_url = "https://usegalaxy.eu",
  in_panel = FALSE,
  panel_id = NULL
)
```

Arguments

galaxy_url	Character. Base URL of the Galaxy instance (for example "https://usegalaxy.eu"). If the environment variable GALAXY_URL is set, it takes precedence.
in_panel	Logical. If TRUE, return the tool panel structure (sections/categories). If FALSE, return the flat list of all tools as supplied by Galaxy. Default: FALSE.
panel_id	Optional character. When supplied, only tools from the matching panel (section/category) are returned. The value is matched against both the panel id and name. Supplying panel_id automatically requests the panelized structure, regardless of the value of in_panel.

Value

A list corresponding to the parsed JSON returned by Galaxy. If panel_id is provided, a list of tool entries belonging to the requested panel is returned (each entry is the raw tool metadata as provided by Galaxy).

Examples

```
# All tools (flat list)
tools_list <- galaxy_list_tools()
length(tools_list)

# Panel structure
panel_list <- galaxy_list_tools(in_panel = TRUE)
length(panel_list)

# Tools from a specific panel (match by id or name)
tools_list <- galaxy_list_tools(panel_id = "Get Data")
length(tools_list)
```

`galaxy_list_workflows` *List workflows available to the user*

Description

Retrieves workflows accessible to the authenticated user from a Galaxy instance. Optionally includes public (published) workflows if supported by the Galaxy server.

Usage

```
galaxy_list_workflows(
  include_public = FALSE,
  galaxy_url = "https://usegalaxy.eu"
)
```

Arguments

`include_public`	Logical. If `TRUE`, attempt to also include published public workflows. Default: `FALSE`.
`galaxy_url`	Character. Base URL of the Galaxy instance (for example `"https://usegalaxy.eu"`). If the environment variable `GALAXY_URL` is set, it takes precedence.

Details

By default, only workflows owned by or shared with the current user are returned. When `include_public = TRUE`, the function will attempt to request published workflows as well. Availability of public workflows depends on the Galaxy instance and version.

Value

A data.frame with one row per workflow and columns including: `id`, `name`, `published`, `owner`.

Examples

```
workflows <- galaxy_list_workflows(TRUE)
head(workflows)
```

`galaxy_poll_tool` *Generic for galaxy_poll_tool*

Description

Generic for galaxy_poll_tool

Wait for a Galaxy job to complete

S4 method to poll the status of a tool invocation

Usage

```
galaxy_poll_tool(
  x,
  galaxy_url = "https://usegalaxy.eu",
  poll_interval = 3,
  timeout = 600
)

## S4 method for signature 'character'
galaxy_poll_tool(
  x,
  galaxy_url = "https://usegalaxy.eu",
  poll_interval = 3,
  timeout = 600
)

## S4 method for signature 'Galaxy'
galaxy_poll_tool(x, poll_interval = 3, timeout = 600)
```

Arguments

`x`	A job ID (`character`) or a `Galaxy` object.
`galaxy_url`	Base URL of the Galaxy instance, used by the character method.
`poll_interval`	Seconds between status checks.
`timeout`	Maximum time to wait in seconds.

Value

For the character method, the final job object; for the `Galaxy` method, the modified `Galaxy` object.

`galaxy_poll_workflow` *Generic for polling workflows*

Description

`galaxy_poll_workflow()` is an S4 generic. With `x` as a character vector it is treated as a workflow invocation ID; the invocation is polled until it completes and a list of output dataset IDs is returned. With `x` as a `Galaxy` object, the `invocation_id` and `galaxy_url` are taken from the object, and the object is updated with the resulting `output_dataset_ids` and state.

Usage

```
galaxy_poll_workflow(
  x,
  galaxy_url = "https://usegalaxy.eu",
  poll_interval = 30,
  ...
)

## S4 method for signature 'character'
galaxy_poll_workflow(
  x,
  galaxy_url = "https://usegalaxy.eu",
  poll_interval = 30,
  ...
)

## S4 method for signature 'Galaxy'
galaxy_poll_workflow(
  x,
  galaxy_url = "https://usegalaxy.eu",
  poll_interval = 30,
  ...
)
```

Arguments

`x`	A workflow invocation ID (`character`) or a `Galaxy` object.
`galaxy_url`	Base URL of the Galaxy instance, used by the character method. If `GALAXY_URL` is set it takes precedence.
`poll_interval`	Time in seconds between polling attempts.
`...`	not in use

Value

For the character method, a list with elements `success` and `output_ids`. For the `Galaxy` method, the modified `Galaxy` object.

Examples

```
invocation_id <- "abc123"
galaxy_poll_workflow(invocation_id)
```

---

galaxy_print_tool_inputs *Print tool inputs*

Description

Print tool inputs

Usage

```
galaxy_print_tool_inputs(inputs)
```

Arguments

inputs	A list of tool inputs as returned by galaxy_get_tool.

Value

A character vector with formatted input descriptions.

Examples

```
tool_id <- galaxy_get_tool_id("FastQC")[1]
fastqc_tool <- galaxy_get_tool(tool_id)
print_tool_inputs(fastqc_tool)
```

---

galaxy_run_tool *Generic run tool*

Description

galaxy_run_tool() is an S4 generic. With x as a character vector it is treated as a history ID; the specified tool is invoked in that history and the job ID is returned. With x as a Galaxy object, the history ID and URL are taken from the object and the object is updated with the job ID.

Usage

```
galaxy_run_tool(
  x,
  tool_id,
  inputs = NULL,
  dataset_id = NULL,
  galaxy_url = "https://usegalaxy.eu"
)

## S4 method for signature 'character'
galaxy_run_tool(
  x,
  tool_id,
  inputs = NULL,
  dataset_id = NULL,
  galaxy_url = "https://usegalaxy.eu"
)

## S4 method for signature 'Galaxy'
galaxy_run_tool(x, tool_id, inputs = NULL, dataset_id = NULL)
```

Arguments

x	A history ID (character) or a Galaxy object.
tool_id	Tool identifier to execute.
inputs	Named list of tool inputs.
dataset_id	ID of the input dataset (HDA).
galaxy_url	Base URL of the Galaxy instance, used by the character method.

Value

For the character method, a job ID; for the Galaxy method, the modified Galaxy object.

---

galaxy_set_credentials

Set Galaxy connection parameters for the current R session

Description

Set Galaxy connection parameters for the current R session

Usage

```
galaxy_set_credentials(
  api_key = NULL,
  username = NULL,
  password = NULL,
  galaxy_url = "https://usegalaxy.eu",
  overwrite = TRUE
)
```

Arguments

api_key	Character. Galaxy API key.
username	Character. Galaxy username (only required for FTP uploads).
password	Character. Galaxy password (only required for FTP uploads).
galaxy_url	Character. Base URL of the Galaxy instance (e.g. "https://usegalaxy.eu"). If set all galaxy_url arguments of functions will be ignored.
overwrite	Logical. Whether to overwrite existing environment variables. Default: TRUE.

Details

This helper is intended for interactive sessions. It sets the following environment variables using Sys.setenv():

- GALAXY_API_KEY
- GALAXY_URL
- GALAXY_USERNAME
- GALAXY_PASSWORD

Only arguments that are provided (non-NULL) are set.

Value

Invisibly returns a named list of values that were set.

Examples

```
# This requires valid credentials to your galaxy instance
## Not run:
galaxy_set_credentials(
  api_key = "your-secret-key",
  username = "your-username",
  password = "your-password",
  galaxy_url = "https://usegalaxy.eu"
)

## End(Not run)
```

galaxy_start_workflow *Generic start workflow*

Description

galaxy_start_workflow() is an S4 generic. With x as a character vector it is treated as a history ID: the given workflow is invoked in that history and the invocation ID is returned. With x as a Galaxy object, the history ID and URL are taken from the object; the workflow is started and the object is updated with the resulting invocation_id.

Usage

```
galaxy_start_workflow(
  x,
  workflow_id,
  inputs = NULL,
  dataset_id = NULL,
  galaxy_url = "https://usegalaxy.eu"
)

## S4 method for signature 'character'
galaxy_start_workflow(
  x,
  workflow_id,
  inputs = NULL,
  dataset_id = NULL,
  galaxy_url = "https://usegalaxy.eu"
)

## S4 method for signature 'Galaxy'
galaxy_start_workflow(x, workflow_id, inputs = NULL, dataset_id = NULL)
```

Arguments

x	A Galaxy object, or a history ID (character) to use the default method.
workflow_id	Character. Galaxy workflow ID.
inputs	Named list. Optional workflow input mapping; keys are workflow input step IDs, values are lists describing datasets/parameters.
dataset_id	Character. ID of the input dataset (HDA). Ignored if inputs is supplied. When x is a Galaxy and dataset_id is missing, x@input_dataset_id is used.
galaxy_url	Base URL of the Galaxy instance, used by the character method. If GALAXY_URL is set it takes precedence.

Value

For the character method, a character scalar invocation ID. For the Galaxy method, the modified Galaxy object.

`galaxy_upload_ftp` *Generic upload ftp*

Description

`galaxy_upload_ftp()` is an S4 generic. With no `x` supplied it uploads a local file via FTP and registers it in the specified history, returning the encoded dataset ID. When called with a `Galaxy` object it uses the object's `history_id` and `galaxy_url` and updates the object with the new `input_dataset_id`.

Usage

```
galaxy_upload_ftp(
  x,
  input_file,
  galaxy_ftp = "ftp.usegalaxy.eu",
  galaxy_url = "https://usegalaxy.eu",
  ...
)

## S4 method for signature 'character'
galaxy_upload_ftp(
  x,
  input_file,
  galaxy_ftp = "ftp.usegalaxy.eu",
  galaxy_url = "https://usegalaxy.eu",
  ...
)

## S4 method for signature 'Galaxy'
galaxy_upload_ftp(
  x,
  input_file,
  galaxy_ftp = "ftp.usegalaxy.eu",
  galaxy_url = "https://usegalaxy.eu",
  ...
)
```

Arguments

`x`	A `Galaxy` object, or a `history_id` to use the default method.
`input_file`	Path to the local file to upload.
`galaxy_ftp`	FTP server address of the Galaxy instance.
`galaxy_url`	Base URL of the Galaxy instance, used by the default method. If `GALAXY_URL` is set it takes precedence.
`...`	not in use

Details

A valid API key (GALAXY_API_KEY) and FTP credentials (GALAXY_USERNAME, GALAXY_PASSWORD) must be available in the environment.

Value

For the default method, a character scalar dataset ID. For the Galaxy method, the modified Galaxy object.

Examples

```
galaxy_ftp <- "ftp.usegalaxy.eu"
input_file <- tempfile(fileext = ".txt")
writeLines("Example", input_file)
hid <- galaxy_initialize("test upload")
did <- galaxy_upload_ftp(input_file, hid, galaxy_ftp)
g <- galaxy()
g <- galaxy_initialize(g)
g <- galaxy_upload_ftp(g, input_file, galaxy_ftp = galaxy_ftp)
```

---

galaxy_upload_https *Generic upload file with https*

Description

galaxy_upload_https() is an S4 generic. With no x supplied it uploads a local file via HTTPS to the specified history and returns the encoded dataset ID. When called with a Galaxy object it uses the object's history_id and galaxy_url, uploads the file, and updates the object with the new input_dataset_id.

Usage

```
galaxy_upload_https(
  x,
  input_file,
  wait = FALSE,
  wait_timeout = 600,
  galaxy_url = "https://usegalaxy.eu",
  file_type = "auto",
  dbkey = "?",
  ...
)

## S4 method for signature 'character'
galaxy_upload_https(
  x,
  input_file,
```

```
  wait = FALSE,
  wait_timeout = 600,
  galaxy_url = "https://usegalaxy.eu",
  file_type = "auto",
  dbkey = "?",
  ...
)

## S4 method for signature 'Galaxy'
galaxy_upload_https(
  x,
  input_file,
  wait = FALSE,
  wait_timeout = 600,
  galaxy_url = "https://usegalaxy.eu",
  file_type = "auto",
  dbkey = "?",
  ...
)
```

Arguments

x	A Galaxy object, or a history_id to use the default method.
input_file	Path to the local file to upload.
wait	Logical. Whether to wait for Galaxy to finish processing.
wait_timeout	Time in seconds until wait times out with an error.
galaxy_url	Base URL of the Galaxy instance, used by the default method. If GALAXY_URL is set it takes precedence.
file_type	Galaxy datatype identifier (e.g. "auto", "fastq", "bam").
dbkey	Reference genome identifier (e.g. "?" or "hg38").
...	not in use

Details

This uses Galaxy's built-in upload1 tool and performs a multipart form POST. Large files may still require FTP depending on server configuration. A valid API key (GALAXY_API_KEY) must be available in the environment.

Value

For the default method, a character scalar dataset ID. For the Galaxy method, the modified Galaxy object.

Examples

```
hid <- galaxy_initialize("test upload")
test_file <- tempfile(fileext = ".txt")
```

```
writeLines("This is an example test file.", test_file)
file_id <- galaxy_upload_https(hid, test_file)
g <- galaxy()
g <- galaxy_initialize(g)
g <- galaxy_upload_https(g, test_file)
```

Index

galaxy, 2
Galaxy-class, 3
galaxy_delete_dataset, 3
galaxy_delete_datasets, 4
galaxy_delete_history, 5
galaxy_delete_history,character-method
(galaxy_delete_history), 5
galaxy_delete_history,Galaxy-method
(galaxy_delete_history), 5
galaxy_download_result, 7
galaxy_download_result,character-method
(galaxy_download_result), 7
galaxy_download_result,Galaxy-method
(galaxy_download_result), 7
galaxy_download_rocrate, 8
galaxy_download_rocrate,character-method
(galaxy_download_rocrate), 8
galaxy_download_rocrate,Galaxy-method
(galaxy_download_rocrate), 8
galaxy_get_file_info, 9
galaxy_get_tool, 10
galaxy_get_tool_id, 11
galaxy_get_workflow, 12
galaxy_get_workflow_inputs, 12
galaxy_has_key, 13
galaxy_history_size, 14
galaxy_initialize, 15
galaxy_initialize,Galaxy-method
(galaxy_initialize), 15
galaxy_initialize,missing-method
(galaxy_initialize), 15
galaxy_list_files, 16
galaxy_list_files,character-method
(galaxy_list_files), 16
galaxy_list_files,Galaxy-method
(galaxy_list_files), 16
galaxy_list_histories, 18
galaxy_list_invocations, 18
galaxy_list_tools, 19
galaxy_list_workflows, 20
galaxy_poll_tool, 21
galaxy_poll_tool,character-method
(galaxy_poll_tool), 21
galaxy_poll_tool,Galaxy-method
(galaxy_poll_tool), 21
galaxy_poll_workflow, 22
galaxy_poll_workflow,character-method
(galaxy_poll_workflow), 22
galaxy_poll_workflow,Galaxy-method
(galaxy_poll_workflow), 22
galaxy_print_tool_inputs, 23
galaxy_run_tool, 23
galaxy_run_tool,character-method
(galaxy_run_tool), 23
galaxy_run_tool,Galaxy-method
(galaxy_run_tool), 23
galaxy_set_credentials, 24
galaxy_start_workflow, 26
galaxy_start_workflow,character-method
(galaxy_start_workflow), 26
galaxy_start_workflow,Galaxy-method
(galaxy_start_workflow), 26
galaxy_upload_ftp, 27
galaxy_upload_ftp,character-method
(galaxy_upload_ftp), 27
galaxy_upload_ftp,Galaxy-method
(galaxy_upload_ftp), 27
galaxy_upload_https, 28
galaxy_upload_https,character-method
(galaxy_upload_https), 28
galaxy_upload_https,Galaxy-method
(galaxy_upload_https), 28