Skip to main content

__Back to top

__ `Ctrl`+`K`

[![conda 25.1.2.dev29 documentation - Home](_static/conda_logo_full.svg)

](index.html)

  * [Conda](https://docs.conda.io/projects/conda/)
  * [Conda-build](https://docs.conda.io/projects/conda-build)
  * [Miniconda](https://docs.anaconda.com/free/miniconda/)
  * conda.org

  * [__ GitHub](https://github.com/conda/conda "GitHub")
  * [![Element](_static/element_logo.svg)](http://bit.ly/conda-chat-room "Element")
  * [__ Discourse](https://conda.discourse.group/ "Discourse")

  * [Conda](https://docs.conda.io/projects/conda/)
  * [Conda-build](https://docs.conda.io/projects/conda-build)
  * [Miniconda](https://docs.anaconda.com/free/miniconda/)
  * conda.org

  * [__ GitHub](https://github.com/conda/conda "GitHub")
  * [![Element](_static/element_logo.svg)](http://bit.ly/conda-chat-room "Element")
  * [__ Discourse](https://conda.discourse.group/ "Discourse")

Navigation

* [Using Custom Locations for Environment and Package Cache](user-guide/configuration/custom-env-and-pkg-locations.html)
    * [Improving interoperability with pip](user-guide/configuration/pip-interoperability.html)
    * [Using the free channel](user-guide/configuration/free-channel.html)
  * [Concepts](user-guide/concepts/index.html) __
    * [Commands](user-guide/concepts/conda-commands.html)
    * [Packages](user-guide/concepts/packages.html)
    * [Package specification](user-guide/concepts/pkg-specs.html)
    * [Package search and install specifications](user-guide/concepts/pkg-search.html)
    * [Channels](user-guide/concepts/channels.html)
    * [Environments](user-guide/concepts/environments.html)
    * [Installing with conda](user-guide/concepts/installing-with-conda.html)
    * [Performance](user-guide/concepts/conda-performance.html)
    * [Conda for data scientists](user-guide/concepts/data-science.html)
    * [Plugins](user-guide/concepts/conda-plugins.html)
  * [Troubleshooting](user-guide/troubleshooting.html)
  * [Cheatsheet](user-guide/cheatsheet.html)
* Configuration
* [Commands](commands/index.html) __
  * [` conda clean`](commands/clean.html)
  * [`conda compare`](commands/compare.html)
  * [`conda config`](commands/config.html)
  * [`conda create`](commands/create.html)
  * [`conda doctor`](commands/doctor.html)
  * [`conda env`](commands/env/index.html) __
    * [` conda env config`](commands/env/config/index.html) __
      * [` conda env config vars`](commands/env/config/vars/index.html) __

        * [` conda env config vars list`](commands/env/config/vars/list.html)
        * [`conda env config vars set`](commands/env/config/vars/set.html)
        * [`conda env config vars unset`](commands/env/config/vars/unset.html)
    * [`conda env create`](commands/env/create.html)
    * [`conda env export`](commands/env/export.html)
    * [`conda env list`](commands/env/list.html)
    * [`conda env remove`](commands/env/remove.html)
    * [`conda env update`](commands/env/update.html)
  * [`conda info`](commands/info.html)
  * [`conda init`](commands/init.html)
  * [`conda install`](commands/install.html)
  * [`conda list`](commands/list.html)
  * [`conda notices`](commands/notices.html)
  * [`conda package`](commands/package.html)
  * [`conda remove`](commands/remove.html)
  * [`conda rename`](commands/rename.html)
  * [`conda run`](commands/run.html)
  * [`conda search`](commands/search.html)
  * [`conda update`](commands/update.html)
* [Release notes](release-notes.html)
* [Glossary](glossary.html)
* [Developer guide](dev-guide/index.html) __
  * [Architecture](dev-guide/architecture.html)
  * [Contributing to conda](dev-guide/contributing.html)
  * [Development Environment](dev-guide/development-environment.html)
  * [Deep dives](dev-guide/deep-dives/index.html) __
    * [` conda install`](dev-guide/deep-dives/install.html)

  * [`conda init` and `conda activate`](dev-guide/deep-dives/activation.html)
  * [`conda config` and context](dev-guide/deep-dives/context.html)
  * [Solvers](dev-guide/deep-dives/solvers.html)
  * [Logging](dev-guide/deep-dives/logging.html)
* [Writing Tests](dev-guide/writing-tests/index.html) __
  * [Integration Tests](dev-guide/writing-tests/integration-tests.html)
* [Deprecations](dev-guide/deprecations.html)
* [Releasing](dev-guide/releasing.html)
* [Plugins](dev-guide/plugins/index.html) __
  * [Auth Handlers](dev-guide/plugins/auth_handlers.html)
  * [Health Checks](dev-guide/plugins/health_checks.html)
  * [Request Headers](dev-guide/plugins/request_headers.html)
  * [Post-commands](dev-guide/plugins/post_commands.html)
  * [Pre-commands](dev-guide/plugins/pre_commands.html)
  * [Reporter Backends](dev-guide/plugins/reporter_backends.html)
  * [Settings](dev-guide/plugins/settings.html)
  * [Solvers](dev-guide/plugins/solvers.html)
  * [Subcommands](dev-guide/plugins/subcommands.html)
  * [Virtual Packages](dev-guide/plugins/virtual_packages.html)
* [Specifications](dev-guide/specs/index.html) __
  * [Technical specification: solver state](dev-guide/specs/solver-state.html)
* [API](dev-guide/api.html) __
  * [` conda`](dev-guide/api/conda/index.html) __
    * [` __main__`](dev-guide/api/conda/__main__/index.html)
    * [`_vendor`](dev-guide/api/conda/_vendor/index.html) __
      * [` frozendict`](dev-guide/api/conda/_vendor/frozendict/index.html)
    * [`_version`](dev-guide/api/conda/_version/index.html)

* [`activate`](dev-guide/api/conda/activate/index.html)
* [`api`](dev-guide/api/conda/api/index.html)
* [`auxlib`](dev-guide/api/conda/auxlib/index.html) __
  * [` collection`](dev-guide/api/conda/auxlib/collection/index.html)
  * [`compat`](dev-guide/api/conda/auxlib/compat/index.html)
  * [`decorators`](dev-guide/api/conda/auxlib/decorators/index.html)
  * [`entity`](dev-guide/api/conda/auxlib/entity/index.html)
  * [`exceptions`](dev-guide/api/conda/auxlib/exceptions/index.html)
  * [`ish`](dev-guide/api/conda/auxlib/ish/index.html)
  * [`logz`](dev-guide/api/conda/auxlib/logz/index.html)
  * [`type_coercion`](dev-guide/api/conda/auxlib/type_coercion/index.html)
* [`base`](dev-guide/api/conda/base/index.html) __
  * [` constants`](dev-guide/api/conda/base/constants/index.html)
  * [`context`](dev-guide/api/conda/base/context/index.html)
* [`cli`](dev-guide/api/conda/cli/index.html) __
  * [` actions`](dev-guide/api/conda/cli/actions/index.html)
  * [`common`](dev-guide/api/conda/cli/common/index.html)
  * [`conda_argparse`](dev-guide/api/conda/cli/conda_argparse/index.html)
  * [`find_commands`](dev-guide/api/conda/cli/find_commands/index.html)
  * [`helpers`](dev-guide/api/conda/cli/helpers/index.html)
  * [`install`](dev-guide/api/conda/cli/install/index.html)
  * [`main`](dev-guide/api/conda/cli/main/index.html)
  * [`main_clean`](dev-guide/api/conda/cli/main_clean/index.html)
  * [`main_commands`](dev-guide/api/conda/cli/main_commands/index.html)
  * [`main_compare`](dev-guide/api/conda/cli/main_compare/index.html)
  * [`main_config`](dev-guide/api/conda/cli/main_config/index.html)
  * [`main_create`](dev-guide/api/conda/cli/main_create/index.html)

  * [`main_env`](dev-guide/api/conda/cli/main_env/index.html)
  * [`main_env_config`](dev-guide/api/conda/cli/main_env_config/index.html)
  * [`main_env_create`](dev-guide/api/conda/cli/main_env_create/index.html)
  * [`main_env_export`](dev-guide/api/conda/cli/main_env_export/index.html)
  * [`main_env_list`](dev-guide/api/conda/cli/main_env_list/index.html)
  * [`main_env_remove`](dev-guide/api/conda/cli/main_env_remove/index.html)
  * [`main_env_update`](dev-guide/api/conda/cli/main_env_update/index.html)
  * [`main_env_vars`](dev-guide/api/conda/cli/main_env_vars/index.html)
  * [`main_export`](dev-guide/api/conda/cli/main_export/index.html)
  * [`main_info`](dev-guide/api/conda/cli/main_info/index.html)
  * [`main_init`](dev-guide/api/conda/cli/main_init/index.html)
  * [`main_install`](dev-guide/api/conda/cli/main_install/index.html)
  * [`main_list`](dev-guide/api/conda/cli/main_list/index.html)
  * [`main_mock_activate`](dev-guide/api/conda/cli/main_mock_activate/index.html)
  * [`main_mock_deactivate`](dev-guide/api/conda/cli/main_mock_deactivate/index.html)
  * [`main_notices`](dev-guide/api/conda/cli/main_notices/index.html)
  * [`main_package`](dev-guide/api/conda/cli/main_package/index.html)
  * [`main_pip`](dev-guide/api/conda/cli/main_pip/index.html)
  * [`main_remove`](dev-guide/api/conda/cli/main_remove/index.html)
  * [`main_rename`](dev-guide/api/conda/cli/main_rename/index.html)
  * [`main_run`](dev-guide/api/conda/cli/main_run/index.html)
  * [`main_search`](dev-guide/api/conda/cli/main_search/index.html)
  * [`main_update`](dev-guide/api/conda/cli/main_update/index.html)
  * [`python_api`](dev-guide/api/conda/cli/python_api/index.html)
* [`common`](dev-guide/api/conda/common/index.html) __
  * [` _logic`](dev-guide/api/conda/common/_logic/index.html)
  * [`_os`](dev-guide/api/conda/common/_os/index.html) __

    * [` linux`](dev-guide/api/conda/common/_os/linux/index.html)
    * [`osx`](dev-guide/api/conda/common/_os/osx/index.html)
    * [`unix`](dev-guide/api/conda/common/_os/unix/index.html)
    * [`windows`](dev-guide/api/conda/common/_os/windows/index.html)
  * [`compat`](dev-guide/api/conda/common/compat/index.html)
  * [`configuration`](dev-guide/api/conda/common/configuration/index.html)
  * [`constants`](dev-guide/api/conda/common/constants/index.html)
  * [`disk`](dev-guide/api/conda/common/disk/index.html)
  * [`io`](dev-guide/api/conda/common/io/index.html)
  * [`iterators`](dev-guide/api/conda/common/iterators/index.html)
  * [`logic`](dev-guide/api/conda/common/logic/index.html)
  * [`path`](dev-guide/api/conda/common/path/index.html) __
    * [` _cygpath`](dev-guide/api/conda/common/path/_cygpath/index.html)
    * [`directories`](dev-guide/api/conda/common/path/directories/index.html)
    * [`python`](dev-guide/api/conda/common/path/python/index.html)
    * [`windows`](dev-guide/api/conda/common/path/windows/index.html)
  * [`pkg_formats`](dev-guide/api/conda/common/pkg_formats/index.html) __
    * [` python`](dev-guide/api/conda/common/pkg_formats/python/index.html)
  * [`serialize`](dev-guide/api/conda/common/serialize/index.html)
  * [`signals`](dev-guide/api/conda/common/signals/index.html)
  * [`toposort`](dev-guide/api/conda/common/toposort/index.html)
  * [`url`](dev-guide/api/conda/common/url/index.html)
* [`core`](dev-guide/api/conda/core/index.html) __
  * [` envs_manager`](dev-guide/api/conda/core/envs_manager/index.html)
  * [`index`](dev-guide/api/conda/core/index/index.html)
  * [`initialize`](dev-guide/api/conda/core/initialize/index.html)
  * [`link`](dev-guide/api/conda/core/link/index.html)

  * [`package_cache_data`](dev-guide/api/conda/core/package_cache_data/index.html)
  * [`path_actions`](dev-guide/api/conda/core/path_actions/index.html)
  * [`portability`](dev-guide/api/conda/core/portability/index.html)
  * [`prefix_data`](dev-guide/api/conda/core/prefix_data/index.html)
  * [`solve`](dev-guide/api/conda/core/solve/index.html)
  * [`subdir_data`](dev-guide/api/conda/core/subdir_data/index.html)
* [`deprecations`](dev-guide/api/conda/deprecations/index.html)
* [`env`](dev-guide/api/conda/env/index.html) __
  * [` env`](dev-guide/api/conda/env/env/index.html)
  * [`installers`](dev-guide/api/conda/env/installers/index.html) __
    * [` base`](dev-guide/api/conda/env/installers/base/index.html)
    * [`conda`](dev-guide/api/conda/env/installers/conda/index.html)
    * [`pip`](dev-guide/api/conda/env/installers/pip/index.html)
  * [`pip_util`](dev-guide/api/conda/env/pip_util/index.html)
  * [`specs`](dev-guide/api/conda/env/specs/index.html) __
    * [` binstar`](dev-guide/api/conda/env/specs/binstar/index.html)
    * [`requirements`](dev-guide/api/conda/env/specs/requirements/index.html)
    * [`yaml_file`](dev-guide/api/conda/env/specs/yaml_file/index.html)
* [`exception_handler`](dev-guide/api/conda/exception_handler/index.html)
* [`exceptions`](dev-guide/api/conda/exceptions/index.html)
* [`exports`](dev-guide/api/conda/exports/index.html)
* [`gateways`](dev-guide/api/conda/gateways/index.html) __
  * [` anaconda_client`](dev-guide/api/conda/gateways/anaconda_client/index.html)
  * [`connection`](dev-guide/api/conda/gateways/connection/index.html) __
    * [` adapters`](dev-guide/api/conda/gateways/connection/adapters/index.html) __
      * [` ftp`](dev-guide/api/conda/gateways/connection/adapters/ftp/index.html)
      * [`http`](dev-guide/api/conda/gateways/connection/adapters/http/index.html)

        * [`localfs`](dev-guide/api/conda/gateways/connection/adapters/localfs/index.html)
        * [`s3`](dev-guide/api/conda/gateways/connection/adapters/s3/index.html)
      * [`download`](dev-guide/api/conda/gateways/connection/download/index.html)
      * [`session`](dev-guide/api/conda/gateways/connection/session/index.html)
    * [`disk`](dev-guide/api/conda/gateways/disk/index.html) __
      * [` create`](dev-guide/api/conda/gateways/disk/create/index.html)
      * [`delete`](dev-guide/api/conda/gateways/disk/delete/index.html)
      * [`link`](dev-guide/api/conda/gateways/disk/link/index.html)
      * [`lock`](dev-guide/api/conda/gateways/disk/lock/index.html)
      * [`permissions`](dev-guide/api/conda/gateways/disk/permissions/index.html)
      * [`read`](dev-guide/api/conda/gateways/disk/read/index.html)
      * [`test`](dev-guide/api/conda/gateways/disk/test/index.html)
      * [`update`](dev-guide/api/conda/gateways/disk/update/index.html)
    * [`logging`](dev-guide/api/conda/gateways/logging/index.html)
    * [`repodata`](dev-guide/api/conda/gateways/repodata/index.html) __
      * [` jlap`](dev-guide/api/conda/gateways/repodata/jlap/index.html) __
        * [` core`](dev-guide/api/conda/gateways/repodata/jlap/core/index.html)
        * [`fetch`](dev-guide/api/conda/gateways/repodata/jlap/fetch/index.html)
        * [`interface`](dev-guide/api/conda/gateways/repodata/jlap/interface/index.html)
      * [`lock`](dev-guide/api/conda/gateways/repodata/lock/index.html)
    * [`subprocess`](dev-guide/api/conda/gateways/subprocess/index.html)
  * [`history`](dev-guide/api/conda/history/index.html)
  * [`instructions`](dev-guide/api/conda/instructions/index.html)
  * [`misc`](dev-guide/api/conda/misc/index.html)
  * [`models`](dev-guide/api/conda/models/index.html) __
    * [` channel`](dev-guide/api/conda/models/channel/index.html)
    * [`dist`](dev-guide/api/conda/models/dist/index.html)

* [`enums`](dev-guide/api/conda/models/enums/index.html)
* [`leased_path_entry`](dev-guide/api/conda/models/leased_path_entry/index.html)
* [`match_spec`](dev-guide/api/conda/models/match_spec/index.html)
* [`package_info`](dev-guide/api/conda/models/package_info/index.html)
* [`prefix_graph`](dev-guide/api/conda/models/prefix_graph/index.html)
* [`records`](dev-guide/api/conda/models/records/index.html)
* [`version`](dev-guide/api/conda/models/version/index.html)

* [`notices`](dev-guide/api/conda/notices/index.html) __
* [` cache`](dev-guide/api/conda/notices/cache/index.html)
* [`core`](dev-guide/api/conda/notices/core/index.html)
* [`fetch`](dev-guide/api/conda/notices/fetch/index.html)
* [`types`](dev-guide/api/conda/notices/types/index.html)
* [`views`](dev-guide/api/conda/notices/views/index.html)

* [`plan`](dev-guide/api/conda/plan/index.html)
* [`plugins`](dev-guide/api/conda/plugins/index.html) __
* [` hookspec`](dev-guide/api/conda/plugins/hookspec/index.html)
* [`manager`](dev-guide/api/conda/plugins/manager/index.html)
* [`post_solves`](dev-guide/api/conda/plugins/post_solves/index.html) __

* [`signature_verification`](dev-guide/api/conda/plugins/post_solves/signature_verification/index.html)

* [`reporter_backends`](dev-guide/api/conda/plugins/reporter_backends/index.html) __
* [` console`](dev-guide/api/conda/plugins/reporter_backends/console/index.html)
* [`json`](dev-guide/api/conda/plugins/reporter_backends/json/index.html)

* [`solvers`](dev-guide/api/conda/plugins/solvers/index.html)
* [`subcommands`](dev-guide/api/conda/plugins/subcommands/index.html) __
* [` doctor`](dev-guide/api/conda/plugins/subcommands/doctor/index.html) __

* [`

health_checks`](dev-guide/api/conda/plugins/subcommands/doctor/health_checks/index.html)

* [`types`](dev-guide/api/conda/plugins/types/index.html)
* [`virtual_packages`](dev-guide/api/conda/plugins/virtual_packages/index.html) __
  * [` archspec`](dev-guide/api/conda/plugins/virtual_packages/archspec/index.html)
  * [`conda`](dev-guide/api/conda/plugins/virtual_packages/conda/index.html)
  * [`cuda`](dev-guide/api/conda/plugins/virtual_packages/cuda/index.html)
  * [`freebsd`](dev-guide/api/conda/plugins/virtual_packages/freebsd/index.html)
  * [`linux`](dev-guide/api/conda/plugins/virtual_packages/linux/index.html)
  * [`osx`](dev-guide/api/conda/plugins/virtual_packages/osx/index.html)
  * [`windows`](dev-guide/api/conda/plugins/virtual_packages/windows/index.html)
* [`reporters`](dev-guide/api/conda/reporters/index.html)
* [`resolve`](dev-guide/api/conda/resolve/index.html)
* [`testing`](dev-guide/api/conda/testing/index.html) __
  * [` cases`](dev-guide/api/conda/testing/cases/index.html)
  * [`fixtures`](dev-guide/api/conda/testing/fixtures/index.html)
  * [`gateways`](dev-guide/api/conda/testing/gateways/index.html) __
    * [` fixtures`](dev-guide/api/conda/testing/gateways/fixtures/index.html)
  * [`helpers`](dev-guide/api/conda/testing/helpers/index.html)
  * [`integration`](dev-guide/api/conda/testing/integration/index.html)
  * [`notices`](dev-guide/api/conda/testing/notices/index.html) __
    * [` fixtures`](dev-guide/api/conda/testing/notices/fixtures/index.html)
    * [`helpers`](dev-guide/api/conda/testing/notices/helpers/index.html)
  * [`solver_helpers`](dev-guide/api/conda/testing/solver_helpers/index.html)
* [`trust`](dev-guide/api/conda/trust/index.html) __
  * [` constants`](dev-guide/api/conda/trust/constants/index.html)
  * [`signature_verification`](dev-guide/api/conda/trust/signature_verification/index.html)
* [`utils`](dev-guide/api/conda/utils/index.html)

* [`conda_env`](dev-guide/api/conda_env/index.html) __

* [` cli`](dev-guide/api/conda_env/cli/index.html)

* [`installers`](dev-guide/api/conda_env/installers/index.html)

* [__](index.html)

* Configuration

Configuration#

```
# #############################################################
# ##              Channel Configuration                      ##
# #############################################################

# # channels (sequence: primitive)
# #   aliases: channel
# #   env var string delimiter: ','
# #   The list of conda channels to include for relevant operations.
# #
# channels: []

# # channel_alias (str)
# #   The prepended url location to associate with channel names.
# #
# channel_alias: https://conda.anaconda.org
```

```
# # channel_settings (sequence: map)
# #   env var string delimiter: ','
# #   A list of mappings that allows overriding certain settings for a
# #   single channel. Each list item should include at least the "channel"
# #   key and the setting you would like to override.
# #
# channel_settings: []


# # default_channels (sequence: primitive)
# #   env var string delimiter: ','
# #   The list of channel names and/or urls used for the 'defaults'
# #   multichannel.
# #
# default_channels:
#   - https://repo.anaconda.com/pkgs/main
#   - https://repo.anaconda.com/pkgs/r


# # override_channels_enabled (bool)
# #   Permit use of the --override-channels command-line flag.
# #
# override_channels_enabled: true


# # allowlist_channels (sequence: primitive)
# #   aliases: whitelist_channels
# #   env var string delimiter: ','
# #   The exclusive list of channels allowed to be used on the system. Use
# #   of any other channels will result in an error. If conda-build channels
```

are to be allowed, along with the --use-local command line flag, be

sure to include the 'local' channel in the list. If the list is empty

or left undefined, no channel exclusions will be enforced.

# #

allowlist_channels: []

denylist_channels (sequence: primitive)

env var string delimiter: ','

The list of channels that are denied to be used on the system. Use of

any of these channels will result in an error. If conda-build channels

are to be allowed, along with the --use-local command line flag, be

sure to not include the 'local' channel in the list. If the list is

empty or left undefined, no channel exclusions will be enforced.

# #

denylist_channels: []

custom_channels (map: primitive)

A map of key-value pairs where the key is a channel name and the value

is a channel location. Channels defined here override the default

'channel_alias' value. The channel name (key) is not included in the

channel location (value). For example, to override the location of

the 'conda-forge' channel where the url to repodata is

https://anaconda-repo.dev/packages/conda-forge/linux-64/repodata.json,

add an entry 'conda-forge: https://anaconda-repo.dev/packages'.

# #

custom_channels:

pkgs/pro: https://repo.anaconda.com

```
# # custom_multichannels (map: sequence)
# #   A multichannel is a metachannel composed of multiple channels. The two
# #   reserved multichannels are 'defaults' and 'local'. The 'defaults'
# #   multichannel is customized using the 'default_channels' parameter. The
# #   'local' multichannel is a list of file:// channel locations where
# #   conda-build stashes successfully-built packages.  Other multichannels
# #   can be defined with custom_multichannels, where the key is the
# #   multichannel name and the value is a list of channel names and/or
# #   channel urls.
# #
# custom_multichannels: {}

# # migrated_channel_aliases (sequence: primitive)
# #   env var string delimiter: ','
# #   A list of previously-used channel_alias values. Useful when switching
# #   between different Anaconda Repository instances.
# #
# migrated_channel_aliases: []

# # migrated_custom_channels (map: primitive)
# #   A map of key-value pairs where the key is a channel name and the value
# #   is the previous location of the channel.
# #
# migrated_custom_channels: {}

# # add_anaconda_token (bool)
```

aliases: add_binstar_token

In conjunction with the anaconda command-line client (installed with

`conda install anaconda-client`), and following logging into an

Anaconda Server API site using `anaconda login`, automatically apply a

matching private token to enable access to private packages and

channels.

# #

add_anaconda_token: true

allow_non_channel_urls (bool)

Warn, but do not fail, when conda detects a channel url is not a valid

channel.

# #

allow_non_channel_urls: false

restore_free_channel (bool)

"   Add the "free" channel back into defaults, behind

"main" in priority. The "free"   channel was removed

from the collection of default channels in conda 4.7.0.

# #

restore_free_channel: false

repodata_fns (sequence: primitive)

env var string delimiter: ','

Specify filenames for repodata fetching. The default is

('current_repodata.json', 'repodata.json'), which tries a subset of

the full index containing only the latest version for each package,

then falls back to repodata.json. You may want to specify something

else to use an alternate index that has been reduced somehow.

# #

repodata_fns:

- current_repodata.json

- repodata.json

use_only_tar_bz2 (NoneType, bool)

A boolean indicating that only .tar.bz2 conda packages should be

downloaded. This is forced to True if conda-build is installed and

older than 3.18.3, because older versions of conda break when conda

feeds it the new file format.

# #

use_only_tar_bz2:

repodata_threads (int)

Threads to use when downloading and reading repodata. When not set,

defaults to None, which uses the default ThreadPoolExecutor behavior.

# #

repodata_threads: 0

fetch_threads (int)

Threads to use when downloading packages. When not set, defaults to

None, which uses the default ThreadPoolExecutor behavior.

# #

fetch_threads: 0

```
# # experimental (sequence: primitive)
# #   env var string delimiter: ','
# #   List of experimental features to enable.
# #
# experimental: []

# # no_lock (bool)
# #   Disable index cache lock (defaults to enabled).
# #
# no_lock: false

# # repodata_use_zst (bool)
# #   Disable check for `repodata.json.zst`; use `repodata.json` only.
# #
# repodata_use_zst: true


# ############################################################
# ##           Basic Conda Configuration           ##
# ############################################################

# # envs_dirs (sequence: primitive)
# #   aliases: envs_path
# #   env var string delimiter: ':'
# #   The list of directories to search for named environments. When
# #   creating a new named environment, the environment will be placed in
# #   the first writable location.
```

```
# #
# envs_dirs: []

# # pkgs_dirs (sequence: primitive)
# #   env var string delimiter: ','
# #   The list of directories where locally-available packages are linked
# #   from at install time. Packages not locally available are downloaded
# #   and extracted into the first writable directory.
# #
# pkgs_dirs: []

# # default_threads (int)
# #   Threads to use by default for parallel operations.  Default is None,
# #   which allows operations to choose themselves.  For more specific
# #   control, see the other *_threads parameters:     * repodata_threads -
# #   for fetching/loading repodata     * verify_threads - for verifying
# #   package contents in transactions     * execute_threads - for carrying
# #   out the unlinking and linking steps
# #
# default_threads: 0


# ############################################################
# ##              Network Configuration              ##
# ############################################################

# # client_ssl_cert (NoneType, str)
```

aliases: client_cert

A path to a single file containing a private key and certificate (e.g.

.pem file). Alternately, use client_ssl_cert_key in conjunction with

client_ssl_cert for individual files.

# #

client_ssl_cert:

client_ssl_cert_key (NoneType, str)

aliases: client_cert_key

Used in conjunction with client_ssl_cert for a matching key file.

# #

client_ssl_cert_key:

local_repodata_ttl (bool, int)

For a value of False or 0, always fetch remote repodata (HTTP 304

responses respected). For a value of True or 1, respect the HTTP

Cache-Control max-age header. Any other positive integer values is the

number of seconds to locally cache repodata before checking the remote

server for an update.

# #

local_repodata_ttl: 1

offline (bool)

Restrict conda to cached download content and file:// based urls.

# #

offline: false

proxy_servers (map: primitive)

A mapping to enable proxy settings. Keys can be either (1) a

scheme://hostname form, which will match any request to the given

scheme and exact hostname, or (2) just a scheme, which will match

requests to that scheme. Values are are the actual proxy server, and

are of the form 'scheme://[user:password@]host[:port]'. The optional

'user:password' inclusion enables HTTP Basic Auth with your proxy.

# #

proxy_servers: {}

remote_connect_timeout_secs (float)

The number seconds conda will wait for your client to establish a

connection to a remote url resource.

# #

remote_connect_timeout_secs: 9.15

remote_max_retries (int)

The maximum number of retries each HTTP connection should attempt.

# #

remote_max_retries: 3

remote_backoff_factor (int)

The factor determines the time HTTP connection should wait for

attempt.

# #

remote_backoff_factor: 1

```
# # remote_read_timeout_secs (float)
# #   Once conda has connected to a remote resource and sent an HTTP
# #   request, the read timeout is the number of seconds conda will wait for
# #   the server to send a response.
# #
# remote_read_timeout_secs: 60.0


# # ssl_verify (bool, str)
# #   aliases: verify_ssl
# #   Conda verifies SSL certificates for HTTPS requests, just like a web
# #   browser. By default, SSL verification is enabled, and conda operations
# #   will fail if a required url's certificate cannot be verified. Setting
# #   ssl_verify to False disables certification verification. The value for
# #   ssl_verify can also be (1) a path to a CA bundle file, (2) a path to a
# #   directory containing certificates of trusted CA, or (3) 'truststore'
# #   to use the operating system certificate store.
# #
# ssl_verify: true



# ##########################################################
# ##              Solver Configuration              ##
# ##########################################################

# # aggressive_update_packages (sequence: primitive)
# #   env var string delimiter: ','
# #   A list of packages that, if installed, are always updated to the
```

latest possible version.

# #

aggressive_update_packages:

- ca-certificates

- certifi

- openssl

auto_update_conda (bool)

aliases: self_update

Automatically update conda when a newer or higher priority version is

detected.

# #

auto_update_conda: true

channel_priority (ChannelPriority)

Accepts values of 'strict', 'flexible', and 'disabled'. The default

value is 'flexible'. With strict channel priority, packages in lower

priority channels are not considered if a package with the same name

appears in a higher priority channel. With flexible channel priority,

the solver may reach into lower priority channels to fulfill

dependencies, rather than raising an unsatisfiable error. With channel

priority disabled, package version takes precedence, and the

configured priority of channels is used only to break ties. In

previous versions of conda, this parameter was configured as either

True or False. True is now an alias to 'flexible'.

# #

channel_priority: flexible

create_default_packages (sequence: primitive)

env var string delimiter: ','

Packages that are by default added to a newly created environments.

# #

create_default_packages: []

disallowed_packages (sequence: primitive)

aliases: disallow

env var string delimiter: '&'

Package specifications to disallow installing. The default is to allow

all packages.

# #

disallowed_packages: []

force_reinstall (bool)

Ensure that any user-requested package for the current operation is

uninstalled and reinstalled, even if that package already exists in

the environment.

# #

force_reinstall: false

pinned_packages (sequence: primitive)

env var string delimiter: '&'

A list of package specs to pin for every environment resolution. This

parameter is in BETA, and its behavior may change in a future release.

# #

```
# pinned_packages: []


# # pip_interop_enabled (bool)
# #   Allow the conda solver to interact with non-conda-installed python
# #   packages.
# #
# pip_interop_enabled: false


# # track_features (sequence: primitive)
# #   env var string delimiter: ','
# #   A list of features that are tracked by default. An entry here is
# #   similar to adding an entry to the create_default_packages list.
# #
# track_features: []


# # solver (str)
# #   aliases: experimental_solver
# #   A string to choose between the different solver logics implemented in
# #   conda. A solver logic takes care of turning your requested packages
# #   into a list of specs to add and/or remove from a given environment,
# #   based on their dependencies and specified constraints.
# #
# solver: libmamba



# ##############################################################
# ##  Package Linking and Install-time Configuration  ##
```

```
# ##############################################################

# # allow_softlinks (bool)
# #   When allow_softlinks is True, conda uses hard-links when possible, and
# #   soft-links (symlinks) when hard-links are not possible, such as when
# #   installing on a different filesystem than the one that the package
# #   cache is on. When allow_softlinks is False, conda still uses hard-
# #   links when possible, but when it is not possible, conda copies files.
# #   Individual packages can override this setting, specifying that certain
# #   files should never be soft-linked (see the no_link option in the build
# #   recipe documentation).
# #
# allow_softlinks: false

# # always_copy (bool)
# #   aliases: copy
# #   Register a preference that files be copied into a prefix during
# #   install rather than hard-linked.
# #
# always_copy: false

# # always_softlink (bool)
# #   aliases: softlink
# #   Register a preference that files be soft-linked (symlinked) into a
# #   prefix during install rather than hard-linked. The link source is the
# #   'pkgs_dir' package cache from where the package is being linked.
# #   WARNING: Using this option can result in corruption of long-lived
```

```
# #   conda environments. Package caches are *caches*, which means there is
# #   some churn and invalidation. With this option, the contents of
# #   environments can be switched out (or erased) via operations on other
# #   environments.
# #
# always_softlink: false

# # path_conflict (PathConflict)
# #   The method by which conda handle's conflicting/overlapping paths
# #   during a create, install, or update operation. The value must be one
# #   of 'clobber', 'warn', or 'prevent'. The '--clobber' command-line flag
# #   or clobber configuration parameter overrides path_conflict set to
# #   'prevent'.
# #
# path_conflict: clobber

# # rollback_enabled (bool)
# #   Should any error occur during an unlink/link transaction, revert any
# #   disk mutations made to that point in the transaction.
# #
# rollback_enabled: true

# # safety_checks (SafetyChecks)
# #   Enforce available safety guarantees during package installation. The
# #   value must be one of 'enabled', 'warn', or 'disabled'.
# #
# safety_checks: warn
```

```
# # extra_safety_checks (bool)
# #   Spend extra time validating package contents.  Currently, runs sha256
# #   verification on every file within each package during installation.
# #
# extra_safety_checks: false

# # signing_metadata_url_base (NoneType, str)
# #   Base URL for obtaining trust metadata updates (i.e., the `*.root.json`
# #   and `key_mgr.json` files) used to verify metadata and (eventually)
# #   package signatures.
# #
# signing_metadata_url_base:

# # shortcuts (bool)
# #   Allow packages to create OS-specific shortcuts (e.g. in the Windows
# #   Start Menu) at install time.
# #
# shortcuts: true

# # shortcuts_only (sequence: primitive)
# #   env var string delimiter: ','
# #   Create shortcuts only for the specified package names.
# #
# shortcuts_only: []

# # non_admin_enabled (bool)
```

Allows completion of conda's create, install, update, and remove

operations, for non-privileged (non-root or non-administrator) users.

# #

non_admin_enabled: true

separate_format_cache (bool)

Treat .tar.bz2 files as different from .conda packages when filenames

are otherwise similar. This defaults to False, so that your package

cache doesn't churn when rolling out the new package format. If you'd

rather not assume that a .tar.bz2 and .conda from the same place

represent the same content, set this to True.

# #

separate_format_cache: false

verify_threads (int)

Threads to use when performing the transaction verification step.

When not set, defaults to 1.

# #

verify_threads: 0

execute_threads (int)

Threads to use when performing the unlink/link transaction. When not

set, defaults to 1. This step is pretty strongly I/O limited, and you

may not see much benefit here.

# #

execute_threads: 0

```
# ##########################################################
# ##           Conda-build Configuration            ##
# ##########################################################

# # bld_path (str)
# #   The location where conda-build will put built packages. Same as
# #   'croot', but 'croot' takes precedence when both are defined. Also used
# #   in construction of the 'local' multichannel.
# #
# bld_path: ''

# # croot (str)
# #   The location where conda-build will put built packages. Same as
# #   'bld_path', but 'croot' takes precedence when both are defined. Also
# #   used in construction of the 'local' multichannel.
# #
# croot: ''

# # anaconda_upload (NoneType, bool)
# #   aliases: binstar_upload
# #   Automatically upload packages built with conda build to anaconda.org.
# #
# anaconda_upload:

# # conda_build (map: primitive)
# #   aliases: conda-build
```

```
# #   General configuration parameters for conda-build.
# #
# conda_build: {}


# ############################################################
# ##  Output, Prompt, and Flow Control Configuration  ##
# ############################################################

# # always_yes (NoneType, bool)
# #   aliases: yes
# #   Automatically choose the 'yes' option whenever asked to proceed with a
# #   conda operation, such as when running `conda install`.
# #
# always_yes:

# # auto_activate_base (bool)
# #   Automatically activate the base environment during shell
# #   initialization.
# #
# auto_activate_base: true

# # auto_stack (int)
# #   Implicitly use --stack when using activate if current level of nesting
# #   (as indicated by CONDA_SHLVL environment variable) is less than or
# #   equal to specified value. 0 or false disables automatic stacking, 1 or
# #   true enables it for one level.
```

```
# #
# auto_stack: 0

# # changeps1 (bool)
# #   When using activate, change the command prompt ($PS1) to include the
# #   activated environment.
# #
# changeps1: true

# # env_prompt (str)
# #   Template for prompt modification based on the active environment.
# #   Currently supported template variables are '{prefix}', '{name}', and
# #   '{default_env}'. '{prefix}' is the absolute path to the active
# #   environment. '{name}' is the basename of the active environment
# #   prefix. '{default_env}' holds the value of '{name}' if the active
# #   environment is a conda named environment ('-n' flag), or otherwise
# #   holds the value of '{prefix}'. Templating uses python's str.format()
# #   method.
# #
# env_prompt: '({default_env}) '

# # json (bool)
# #   Ensure all output written to stdout is structured json.
# #
# json: false

# # console (str)
```

Configure different backends to be used while rendering normal console

output. Defaults to "classic".

# #

console: classic

notify_outdated_conda (bool)

Notify if a newer version of conda is detected during a create,

install, update, or remove operation.

# #

notify_outdated_conda: true

quiet (bool)

Disable progress bar display and other output.

# #

quiet: false

report_errors (NoneType, bool)

Opt in, or opt out, of automatic error reporting to core maintainers.

Error reports are anonymous, with only the error stack trace and

information given by `conda info` being sent.

# #

report_errors:

show_channel_urls (NoneType, bool)

Show channel URLs when displaying what is going to be downloaded.

# #

show_channel_urls:

verbosity (int)

aliases: verbose

Sets output log level. 0 is warn. 1 is info. 2 is debug. 3 is trace.

# #

verbosity: 0

unsatisfiable_hints (bool)

A boolean to determine if conda should find conflicting packages in

the case of a failed install.

# #

unsatisfiable_hints: true

unsatisfiable_hints_check_depth (int)

An integer that specifies how many levels deep to search for

unsatisfiable dependencies. If this number is 1 it will complete the

unsatisfiable hints fastest (but perhaps not the most complete). The

higher this number, the longer the generation of the unsat hint will

take. Defaults to 3.

# #

unsatisfiable_hints_check_depth: 2

number_channel_notices (int)

Sets the number of channel notices to be displayed when running

commands the "install", "create", "update", "env create", and "env

update" . Defaults to 5. In order to completely suppress channel

notices, set this to 0.

```
# #
# number_channel_notices: 5


# # envvars_force_uppercase (bool)
# #   Force uppercase for new environment variable names. Defaults to True.
# #
# envvars_force_uppercase: true



# ############################################################
# ##              Plugin Configuration              ##
# ############################################################

# # no_plugins (bool)
# #   Disable all currently-registered plugins, except built-in conda
# #   plugins.
# #
# no_plugins: false
```

[__Edit on

GitHub](https://github.com/conda/conda/edit/main/docs/source/configuration.rst)

[__Show Source](_sources/configuration.rst.txt)

Created using [Sphinx](https://www.sphinx-doc.org/) 7.4.7.

[Analytics Dashboard __](https://docs-conda-io.goatcounter.com "Analytics Dashboard")

Built with the [PyData Sphinx Theme](https://pydata-sphinx-theme.readthedocs.io/en/stable/index.html) 0.15.4.