

[Skip to main content](#)

[\\_\\_Back to top](#)

[\\_\\_`Ctrl`+`K`](#)

[ [!\[conda 25.1.2.dev29 documentation -](#)

[Home\]\(../../../../../\\_static/conda\\_logo\\_full.svg\) \]\(../../../../../index.html\)](#)

\* [\[Conda\]\(https://docs.conda.io/projects/conda/\)](#)

\* [\[Conda-build\]\(https://docs.conda.io/projects/conda-build\)](#)

\* [\[Miniconda\]\(https://docs.anaconda.com/free/miniconda/\)](#)

\* [\[conda.org\]\(https://conda.org\)](#)

\* [\[\\_\\_ GitHub\]\(https://github.com/conda/conda "GitHub"\)](#)

\* [\[!\[Element\]\(../../../../../\\_static/element\\_logo.svg\)\]\(http://bit.ly/conda-chat-room "Element"\)](#)

\* [\[\\_\\_ Discourse\]\(https://conda.discourse.group/ "Discourse"\)](#)

\* [\[Conda\]\(https://docs.conda.io/projects/conda/\)](#)

\* [\[Conda-build\]\(https://docs.conda.io/projects/conda-build\)](#)

\* [\[Miniconda\]\(https://docs.anaconda.com/free/miniconda/\)](#)

\* [\[conda.org\]\(https://conda.org\)](#)

\* [\[\\_\\_ GitHub\]\(https://github.com/conda/conda "GitHub"\)](#)

\* [\[!\[Element\]\(../../../../../\\_static/element\\_logo.svg\)\]\(http://bit.ly/conda-chat-room "Element"\)](#)

\* [\[\\_\\_ Discourse\]\(https://conda.discourse.group/ "Discourse"\)](#)

## Navigation

- \* [User guide](../././././user-guide/index.html) \_\_
- \* [Getting started with conda](../././././user-guide/getting-started.html)
- \* [Installing conda](../././././user-guide/install/index.html) \_\_
  - \* [Installing on Windows](../././././user-guide/install/windows.html)
  - \* [Installing on macOS](../././././user-guide/install/macos.html)
  - \* [Installing on Linux](../././././user-guide/install/linux.html)
  - \* [RPM and Debian Repositories for Miniconda](../././././user-guide/install/rpm-debian.html)
- \* [Tasks](../././././user-guide/tasks/index.html) \_\_
  - \* [Managing conda](../././././user-guide/tasks/manage-conda.html)
  - \* [Managing environments](../././././user-guide/tasks/manage-environments.html)
  - \* [Managing channels](../././././user-guide/tasks/manage-channels.html)
  - \* [Managing packages](../././././user-guide/tasks/manage-pkgs.html)
  - \* [Managing Python](../././././user-guide/tasks/manage-python.html)
  - \* [Managing virtual packages](../././././user-guide/tasks/manage-virtual.html)
  - \* [Creating custom channels](../././././user-guide/tasks/create-custom-channels.html)
  - \* [Creating projects](../././././user-guide/tasks/creating-projects.html)
  - \* [Viewing command-line help](../././././user-guide/tasks/view-command-line-help.html)
- \* [Configuration](../././././user-guide/configuration/index.html) \_\_
  - \* [Using the .condarc conda configuration file](../././././user-guide/configuration/use-condarc.html)
  - \* [Settings](../././././user-guide/configuration/settings.html)
    - \* [Administering a multi-user conda installation](../././././user-guide/configuration/admin-multi-user-install.html)
  - \* [Mirroring channels](../././././user-guide/configuration/mirroring.html)
  - \* [Disabling SSL verification](../././././user-guide/configuration/disable-ssl-verification.html)

- \* [Using non-standard certificates](../../../../../user-guide/configuration/non-standard-certs.html)
- \* [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](../../../../../configuration.html)
- \* [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](../.../index.html) \_\_
- \* [Architecture](../.../architecture.html)
- \* [Contributing to conda](../.../contributing.html)
- \* [Development Environment](../.../development-environment.html)
- \* [Deep dives](../.../deep-dives/index.html) \_\_

- \* [`conda install``](../../deep-dives/install.html)
- \* [`conda init`` and `conda activate``](../../deep-dives/activation.html)
- \* [`conda config`` and context](../../deep-dives/context.html)
- \* [Solvers](../../deep-dives/solvers.html)
- \* [Logging](../../deep-dives/logging.html)
- \* [Writing Tests](../../writing-tests/index.html) \_\_
- \* [Integration Tests](../../writing-tests/integration-tests.html)
- \* [Deprecations](../../deprecations.html)
- \* [Releasing](../../releasing.html)
- \* [Plugins](../../plugins/index.html) \_\_
  - \* [Auth Handlers](../../plugins/auth\_handlers.html)
  - \* [Health Checks](../../plugins/health\_checks.html)
  - \* [Request Headers](../../plugins/request\_headers.html)
  - \* [Post-commands](../../plugins/post\_commands.html)
  - \* [Pre-commands](../../plugins/pre\_commands.html)
  - \* [Reporter Backends](../../plugins/reporter\_backends.html)
  - \* [Settings](../../plugins/settings.html)
  - \* [Solvers](../../plugins/solvers.html)
  - \* [Subcommands](../../plugins/subcommands.html)
  - \* [Virtual Packages](../../plugins/virtual\_packages.html)
- \* [Specifications](../../specs/index.html) \_\_
  - \* [Technical specification: solver state](../../specs/solver-state.html)
- \* [API](../../api.html) \_\_
  - \* [`conda``](../index.html) \_\_
    - \* [`__main__``](../\_\_main\_\_/index.html)
    - \* [`_vendor``](../\_vendor/index.html) \_\_
      - \* [`frozendict``](../\_vendor/frozendict/index.html)

- \* [`_version``](../\_version/index.html)
- \* [`activate``](../activate/index.html)
- \* [`api``](../api/index.html)
- \* [`auxlib``](../auxlib/index.html) \_\_
  - \* [`collection``](../auxlib/collection/index.html)
  - \* [`compat``](../auxlib/compat/index.html)
  - \* [`decorators``](../auxlib/decorators/index.html)
  - \* [`entity``](../auxlib/entity/index.html)
  - \* [`exceptions``](../auxlib/exceptions/index.html)
  - \* [`ish``](../auxlib/ish/index.html)
  - \* [`logz``](../auxlib/logz/index.html)
  - \* [`type_coercion``](../auxlib/type\_coercion/index.html)
- \* [`base``](../base/index.html) \_\_
  - \* [`constants``](../base/constants/index.html)
  - \* [`context``](../base/context/index.html)
- \* [`cli``](../cli/index.html) \_\_
  - \* [`actions``](../cli/actions/index.html)
  - \* [`common``](../cli/common/index.html)
  - \* [`conda_argparse``](../cli/conda\_argparse/index.html)
  - \* [`find_commands``](../cli/find\_commands/index.html)
  - \* [`helpers``](../cli/helpers/index.html)
  - \* [`install``](../cli/install/index.html)
  - \* [`main``](../cli/main/index.html)
  - \* [`main_clean``](../cli/main\_clean/index.html)
  - \* [`main_commands``](../cli/main\_commands/index.html)
  - \* [`main_compare``](../cli/main\_compare/index.html)
  - \* [`main_config``](../cli/main\_config/index.html)

- \* [main\_create](../../cli/main\_create/index.html)
- \* [main\_env](../../cli/main\_env/index.html)
- \* [main\_env\_config](../../cli/main\_env\_config/index.html)
- \* [main\_env\_create](../../cli/main\_env\_create/index.html)
- \* [main\_env\_export](../../cli/main\_env\_export/index.html)
- \* [main\_env\_list](../../cli/main\_env\_list/index.html)
- \* [main\_env\_remove](../../cli/main\_env\_remove/index.html)
- \* [main\_env\_update](../../cli/main\_env\_update/index.html)
- \* [main\_env\_vars](../../cli/main\_env\_vars/index.html)
- \* [main\_export](../../cli/main\_export/index.html)
- \* [main\_info](../../cli/main\_info/index.html)
- \* [main\_init](../../cli/main\_init/index.html)
- \* [main\_install](../../cli/main\_install/index.html)
- \* [main\_list](../../cli/main\_list/index.html)
- \* [main\_mock\_activate](../../cli/main\_mock\_activate/index.html)
- \* [main\_mock\_deactivate](../../cli/main\_mock\_deactivate/index.html)
- \* [main\_notices](../../cli/main\_notices/index.html)
- \* [main\_package](../../cli/main\_package/index.html)
- \* [main\_pip](../../cli/main\_pip/index.html)
- \* [main\_remove](../../cli/main\_remove/index.html)
- \* [main\_rename](../../cli/main\_rename/index.html)
- \* [main\_run](../../cli/main\_run/index.html)
- \* [main\_search](../../cli/main\_search/index.html)
- \* [main\_update](../../cli/main\_update/index.html)
- \* [python\_api](../../cli/python\_api/index.html)
- \* [common](../../common/index.html) \_\_
- \* [\_logic](../../common/\_logic/index.html)

- \* [`_os``](../../common/\_os/index.html) \_\_
- \* [`linux``](../../common/\_os/linux/index.html)
- \* [`osx``](../../common/\_os/osx/index.html)
- \* [`unix``](../../common/\_os/unix/index.html)
- \* [`windows``](../../common/\_os/windows/index.html)
- \* [`compat``](../../common/compat/index.html)
- \* [`configuration``](../../common/configuration/index.html)
- \* [`constants``](../../common/constants/index.html)
- \* [`disk``](../../common/disk/index.html)
- \* [`io``](../../common/io/index.html)
- \* [`iterators``](../../common/iterators/index.html)
- \* [`logic``](../../common/logic/index.html)
- \* [`path``](../../common/path/index.html) \_\_
- \* [`_cygpath``](../../common/path/\_cygpath/index.html)
- \* [`directories``](../../common/path/directories/index.html)
- \* [`python``](../../common/path/python/index.html)
- \* [`windows``](../../common/path/windows/index.html)
- \* [`pkg_formats``](../../common/pkg\_formats/index.html) \_\_
- \* [`python``](../../common/pkg\_formats/python/index.html)
- \* [`serialize``](../../common/serialize/index.html)
- \* [`signals``](../../common/signals/index.html)
- \* [`toposort``](../../common/toposort/index.html)
- \* [`url``](../../common/url/index.html)
- \* [`core``](../../core/index.html) \_\_
- \* [`envs_manager``](../../core/envs\_manager/index.html)
- \* [`index``](../../core/index/index.html)
- \* [`initialize``](../../core/initialize/index.html)



- \* [`link``](../../core/link/index.html)
- \* [`package_cache_data``](../../core/package\_cache\_data/index.html)
- \* [`path_actions``](../../core/path\_actions/index.html)
- \* [`portability``](../../core/portability/index.html)
- \* [`prefix_data``](../../core/prefix\_data/index.html)
- \* [`solve``](../../core/solve/index.html)
- \* [`subdir_data``](../../core/subdir\_data/index.html)
- \* [`deprecations``](../../deprecations/index.html)
- \* [`env``](../../env/index.html) \_\_\_\_
- \* [ `env``](../../env/env/index.html)
- \* [`installers``](../../env/installers/index.html) \_\_\_\_
- \* [ `base``](../../env/installers/base/index.html)
- \* [`conda``](../../env/installers/conda/index.html)
- \* [`pip``](../../env/installers/pip/index.html)
- \* [`pip_util``](../../env/pip\_util/index.html)
- \* [`specs``](../../env/specs/index.html) \_\_\_\_
- \* [`binstar``](../../env/specs/binstar/index.html)
- \* [`requirements``](../../env/specs/requirements/index.html)
- \* [`yaml_file``](../../env/specs/yaml\_file/index.html)
- \* [`exception_handler``](../../exception\_handler/index.html)
- \* [`exceptions``](../../exceptions/index.html)
- \* [`exports``](../../exports/index.html)
- \* [`gateways``](../../gateways/index.html) \_\_\_\_
- \* [ `anaconda_client``](../../gateways/anaconda\_client/index.html)
- \* [`connection``](../../gateways/connection/index.html) \_\_\_\_
- \* [ `adapters``](../../gateways/connection/adapters/index.html) \_\_\_\_
- \* [ `ftp``](../../gateways/connection/adapters/ftp/index.html)

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

- \* [`dist``](../dist/index.html)
- \* [`enums``](../enums/index.html)
- \* [`leased_path_entry``](../leased\_path\_entry/index.html)
- \* `match_spec``
- \* [`package_info``](../package\_info/index.html)
- \* [`prefix_graph``](../prefix\_graph/index.html)
- \* [`records``](../records/index.html)
- \* [`version``](../version/index.html)
- \* [`notices``](../notices/index.html) \_\_
- \* [`cache``](../notices/cache/index.html)
- \* [`core``](../notices/core/index.html)
- \* [`fetch``](../notices/fetch/index.html)
- \* [`types``](../notices/types/index.html)
- \* [`views``](../notices/views/index.html)
- \* [`plan``](../plan/index.html)
- \* [`plugins``](../plugins/index.html) \_\_
- \* [`hookspec``](../plugins/hookspec/index.html)
- \* [`manager``](../plugins/manager/index.html)
- \* [`post_solves``](../plugins/post\_solves/index.html) \_\_
- \* [`signature_verification``](../plugins/post\_solves/signature\_verification/index.html)
- \* [`reporter_backends``](../plugins/reporter\_backends/index.html) \_\_
- \* [`console``](../plugins/reporter\_backends/console/index.html)
- \* [`json``](../plugins/reporter\_backends/json/index.html)
- \* [`solvers``](../plugins/solvers/index.html)
- \* [`subcommands``](../plugins/subcommands/index.html) \_\_
- \* [`doctor``](../plugins/subcommands/doctor/index.html) \_\_
- \* [`health_checks``](../plugins/subcommands/doctor/health\_checks/index.html)

- \* [ `types` ](../../plugins/types/index.html)
- \* [ `virtual\_packages` ](../../plugins/virtual\_packages/index.html) \_\_
  - \* [ ` archspec` ](../../plugins/virtual\_packages/archspec/index.html)
  - \* [ `conda` ](../../plugins/virtual\_packages/conda/index.html)
  - \* [ `cuda` ](../../plugins/virtual\_packages/cuda/index.html)
  - \* [ `freebsd` ](../../plugins/virtual\_packages/freebsd/index.html)
  - \* [ `linux` ](../../plugins/virtual\_packages/linux/index.html)
  - \* [ `osx` ](../../plugins/virtual\_packages/osx/index.html)
  - \* [ `windows` ](../../plugins/virtual\_packages/windows/index.html)
- \* [ `reporters` ](../../reporters/index.html)
- \* [ `resolve` ](../../resolve/index.html)
- \* [ `testing` ](../../testing/index.html) \_\_
  - \* [ `cases` ](../../testing/cases/index.html)
  - \* [ `fixtures` ](../../testing/fixtures/index.html)
  - \* [ `gateways` ](../../testing/gateways/index.html) \_\_
    - \* [ ` fixtures` ](../../testing/gateways/fixtures/index.html)
  - \* [ `helpers` ](../../testing/helpers/index.html)
  - \* [ `integration` ](../../testing/integration/index.html)
  - \* [ `notices` ](../../testing/notices/index.html) \_\_
    - \* [ ` fixtures` ](../../testing/notices/fixtures/index.html)
    - \* [ `helpers` ](../../testing/notices/helpers/index.html)
  - \* [ `solver\_helpers` ](../../testing/solver\_helpers/index.html)
- \* [ `trust` ](../../trust/index.html) \_\_
  - \* [ ` constants` ](../../trust/constants/index.html)
  - \* [ `signature\_verification` ](../../trust/signature\_verification/index.html)
- \* [ `utils` ](../../utils/index.html)
- \* [ `conda\_env` ](../../conda\_env/index.html) \_\_

```
* [ `cli` ](../../conda_env/cli/index.html)
```

```
* [ `installers` ](../../conda_env/installers/index.html)
```

```
* [ ____ ](../../index.html)
```

```
* [Developer guide](../../index.html)
```

```
* ____
```

```
* [ `models` ](../index.html)
```

```
*
```

```
# `match_spec`#
```

Implements the query language for conda packages (a.k.a, MatchSpec).

The MatchSpec is the conda package specification (e.g. conda==23.3, python<3.7, cryptography \* \*\_0) and is used to communicate the desired packages to install.

```
## Classes#
```

```
`MatchSpecType` |
```

```
---|---
```

```
`MatchSpec` | The query language for conda packages.
```

```
`MatchInterface` |
```

```
`_StrMatchMixin` |
```

```
`ExactStrMatch` |
```

```
`ExactLowerStrMatch` |
```

```
`GlobStrMatch` |
```

``GlobLowerStrMatch` |`

``SplitStrMatch` |`

``FeatureMatch` |`

``ChannelMatch` |`

``CaseInsensitiveStrMatch` |`

`## Functions#`

``_parse_version_plus_build`(v_plus_b) |` This should reliably pull the build string out of a version + build string combo.

`---|---`

``_parse_legacy_dist`(dist_str) |`

``_parse_channel`(channel_val) |`

``_parse_spec_str`(spec_str) |`

`## Attributes#`

``_PARSE_CACHE` |`

`---|---`

``_implementors` |`

`_class _MatchSpecType#`

Bases: [``type``](<https://docs.python.org/3/library/functions.html#type>) "`\(in Python v3.13\)`")

```
__call__(_spec_arg=None_, **kwargs_)#
```

Call self as a function.

```
_class _MatchSpec(_optional=False_, _target=None_, **kwargs_)#
```

The query language for conda packages.

Any of the fields that comprise a ``PackageRecord`` can be used to compose a ``MatchSpec``.

``MatchSpec`` can be composed with keyword arguments, where keys are any of the attributes of ``PackageRecord``. Values for keyword arguments are the exact values the attribute should match against. Many fields can also be matched against non-exact values--by including wildcard `*` and `>/<` ranges--where supported. Any non-specified field is the equivalent of a full wildcard match.

``MatchSpec`` can also be composed using a single positional argument, with optional keyword arguments. Keyword arguments also override any conflicting information provided in the positional argument. The positional argument can be either an existing ``MatchSpec`` instance or a string. Conda has historically supported more than one string representation for equivalent ``MatchSpec``

queries. This `MatchSpec` should accept any existing valid spec string, and correctly compose a `MatchSpec` instance.

A series of rules are now followed for creating the canonical string representation of a `MatchSpec` instance. The canonical string representation can generically be represented by

```
> (channel(/subdir):(namespace):)name(version(build))[key1=value1,key2=value2]
```

where `()` indicate optional fields. The rules for constructing a canonical string representation are:

1. name (i.e. "package name") is required, but its value can be `'*'`. Its position is always outside the key-value brackets.
2. If version is an exact version, it goes outside the key-value brackets and is prepended by `==`. If version is a "fuzzy" value (e.g. `1.11.*`), it goes outside the key-value brackets with the `.*` left off and is prepended by `=`. Otherwise version is included inside key-value brackets.
3. If version is an exact version, and build is an exact value, build goes outside key-value brackets prepended by a `=`. Otherwise, build goes inside key-value brackets. `build_string` is an alias for `build`.
4. The namespace position is being held for a future conda feature.
5. If channel is included and is an exact value, a `::` separator is used between channel and name. channel can either be a canonical channel name or a channel url. In the canonical string representation, the canonical channel name will always be used.



6. If channel is an exact value and subdir is an exact value, subdir is appended to channel with a / separator. Otherwise, subdir is included in the key-value brackets.

7. Key-value brackets can be delimited by comma, space, or comma+space. Value can optionally be wrapped in single or double quotes, but must be wrapped if value contains a comma, space, or equal sign. The canonical format uses comma delimiters and single quotes.

8. When constructing a ``MatchSpec`` instance from a string, any key-value pair given inside the key-value brackets overrides any matching parameter given outside the brackets.

When ``MatchSpec`` attribute values are simple strings, they are interpreted using the following conventions:

> \* If the string begins with ^ and ends with \$, it is converted to a regex.

>

> \* If the string contains an asterisk (\*), it is transformed from a glob to

> a regex.

>

> \* Otherwise, an exact match to the string is sought.

>

>

Examples

```
>>> str(MatchSpec(name='foo', build='py2*', channel='conda-forge'))
'conda-forge::foo[build=py2*]'
>>> str(MatchSpec('foo 1.0 py27_0'))
'foo==1.0=py27_0'
>>> str(MatchSpec('foo=1.0=py27_0'))
'foo==1.0=py27_0'
>>> str(MatchSpec('conda-forge::foo[version=1.0.*]'))
'conda-forge::foo=1.0'
>>> str(MatchSpec('conda-forge/linux-64::foo>=1.0'))
"conda-forge/linux-64::foo[version='>=1.0']"
>>> str(MatchSpec('*/linux-64::foo>=1.0'))
"foo[subdir=linux-64,version='>=1.0']"
```

To fully-specify a package with a full, exact spec, the fields

\* channel

\* subdir

\* name

\* version

\* build

must be given as exact values. In the future, the namespace field will be added to this list. Alternatively, an exact spec is given by

'\*[md5=12345678901234567890123456789012]' or

'\*[sha256=f453db4ffe2271ec492a2913af4e61d4a6c118201f07de757df0eff769b65d2e]'.

\_property \_is\_name\_only\_spec#

\_property \_optional#

\_property \_target#

\_property \_original\_spec\_str#

\_property \_name#

\_property \_strictness#

\_property \_spec#

\_property \_version#

\_property \_fn#

FIELD\_NAMES \_ = ('channel', 'subdir', 'name', 'version', 'build',  
'build\_number', 'track\_features', 'features',...\_#

FIELD\_NAMES\_SET#

\_MATCHER\_CACHE#

`_classmethod _from_dist_str(_dist_str_)`#

`get_exact_value(_field_name_)`#

`get_raw_value(_field_name_)`#

`get(_field_name_ , _default =None_)`#

`dist_str()`#

`match(_rec_)`#

Accepts a PackageRecord or a dict, and matches can pull from any field in that record. Returns True for a match, and False for no match.

`_match_individual(_record_ , _field_name_ , _match_component_)#`

`_is_simple()#`

`_is_single()#`

`_to_filename_do_not_use()#`

`__repr__()#`

Return repr(self).

`__str__()#`

Return str(self).

\_\_json\_\_()#

conda\_build\_form()#

\_\_eq\_\_(\_other\_)#

Return self==value.

\_\_hash\_\_()#

Return hash(self).

\_hash\_key()#

\_\_contains\_\_(\_field\_)#

`_build_components(_** kwargs_)#`

`_static __make_component(_field_name_ , _value_)#`

`_classmethod _merge(_match_specs_ , _union =False_)#`

`_classmethod _union(_match_specs_)#`

`_merge(_other_ , _union =False_)#`

`_parse_version_plus_build(_v_plus_b_)#`



This should reliably pull the build string out of a version + build string

combo. .. rubric:: Examples

```
>>> _parse_version_plus_build("=1.2.3 0")
('=1.2.3', '0')
>>> _parse_version_plus_build("1.2.3=0")
('1.2.3', '0')
>>> _parse_version_plus_build(">=1.0 , < 2.0 py34_0")
('>=1.0,<2.0', 'py34_0')
>>> _parse_version_plus_build(">=1.0 , < 2.0 =py34_0")
('>=1.0,<2.0', 'py34_0')
>>> _parse_version_plus_build("=1.2.3 ")
('=1.2.3', None)
>>> _parse_version_plus_build(">1.8,<2|==1.7")
('>1.8,<2|==1.7', None)
>>> _parse_version_plus_build("* openblas_0")
('*', 'openblas_0')
>>> _parse_version_plus_build("** *")
('*', '*')
```

`_parse_legacy_dist(_dist_str_)`#

## Examples

```
>>> _parse_legacy_dist("_license-1.1-py27_1.tar.bz2")  
  
('_license', '1.1', 'py27_1')  
  
>>> _parse_legacy_dist("_license-1.1-py27_1")  
  
('_license', '1.1', 'py27_1')
```

```
_parse_channel(_channel_val_)#
```

```
_PARSE_CACHE#
```

```
_parse_spec_str(_spec_str_)#
```

```
_class _MatchInterface(_value_)#
```

```
_property _raw_value#
```

`_abstract property _exact_value#`

If the match value is an exact specification, returns the value. Otherwise returns None.

`_abstract _match(_other_)#`

`matches(_value_)#`

`merge(_other_)#`

`union(_other_)#`

`_class __StrMatchMixin#`

`_property _exact_value#`

`__str__()`#

Return `str(self)`.

`__repr__()`#

Return `repr(self)`.

`__eq__(_other_)`#

Return `self==value`.

`__hash__()`#

Return hash(self).

`_class _ExactStrMatch(_value_)#`

Bases: ``_StrMatchMixin`, `MatchInterface``

`__slots__ = ('_raw_value',)_#`

`match(_other_)#`

`_class _ExactLowerStrMatch(_value_)#`

Bases: ``ExactStrMatch``

`match(_other_)#`

```
_class _GlobStrMatch(_value_)#
```

```
Bases: `_StrMatchMixin`, `MatchInterface`
```

```
_property _exact_value#
```

If the match value is an exact specification, returns the value. Otherwise returns None.

```
_property _matches_all#
```

```
__slots__ = ('_raw_value', '_re_match')_#
```

```
match(_other_)#
```

```
merge(_other_)#
```

`_class _GlobLowerStrMatch(_value_)#`

Bases: ``GlobStrMatch``

`_class _SplitStrMatch(_value_)#`

Bases: ``MatchInterface``

`_property _exact_value#`

If the match value is an exact specification, returns the value. Otherwise  
returns None.

`__slots__ = ('_raw_value',)_#`

`_convert(_value_)#`

match(\_other\_)#

\_\_repr\_\_()#

Return repr(self).

\_\_str\_\_()#

Return str(self).

\_\_eq\_\_(\_other\_)#

Return self==value.

\_\_hash\_\_()#



Return hash(self).

`_class _FeatureMatch(_value_)`#

Bases: ``MatchInterface``

`_property _exact_value`#

If the match value is an exact specification, returns the value. Otherwise  
returns None.

`__slots__ = ('_raw_value',)_`#

`_convert(_value_)`#

`match(_other_)`#

`__repr__()`#

Return repr(self).

`__str__()`#

Return str(self).

`__eq__(_other_)`#

Return self==value.

`__hash__()`#

Return hash(self).

`_class _ChannelMatch(_value_)`#

Bases: ``GlobStrMatch``

`match(_other_)`#

`__str__()`#

Return `str(self)`.

`__repr__()`#

Return `repr(self)`.

`_class_ _CaseInsensitiveStrMatch(_value_)`#

Bases: ``GlobLowerStrMatch``

match(\_other\_)#

\_implementors#

\_\_\_On this page

- \* Classes

- \* Functions

- \* Attributes

- \* `MatchSpecType`

- \* `MatchSpecType.\_\_call\_\_()`

- \* `MatchSpec`

- \* `MatchSpec.is\_name\_only\_spec`

- \* `MatchSpec.optional`

- \* `MatchSpec.target`

- \* `MatchSpec.original\_spec\_str`

- \* `MatchSpec.name`

- \* `MatchSpec.strictness`

- \* `MatchSpec.spec`

- \* `MatchSpec.version`

- \* `MatchSpec.fn`

- \* `MatchSpec.FIELD\_NAMES`

- \* `MatchSpec.FIELD\_NAMES\_SET`

- \* ``MatchSpec._MATCHER_CACHE``
- \* ``MatchSpec.from_dist_str()``
- \* ``MatchSpec.get_exact_value()``
- \* ``MatchSpec.get_raw_value()``
- \* ``MatchSpec.get()``
- \* ``MatchSpec.dist_str()``
- \* ``MatchSpec.match()``
- \* ``MatchSpec._match_individual()``
- \* ``MatchSpec._is_simple()``
- \* ``MatchSpec._is_single()``
- \* ``MatchSpec._to_filename_do_not_use()``
- \* ``MatchSpec.__repr__()``
- \* ``MatchSpec.__str__()``
- \* ``MatchSpec.__json__()``
- \* ``MatchSpec.conda_build_form()``
- \* ``MatchSpec.__eq__()``
- \* ``MatchSpec.__hash__()``
- \* ``MatchSpec._hash_key()``
- \* ``MatchSpec.__contains__()``
- \* ``MatchSpec._build_components()``
- \* ``MatchSpec._make_component()``
- \* ``MatchSpec.merge()``
- \* ``MatchSpec.union()``
- \* ``MatchSpec._merge()``
- \* ``_parse_version_plus_build()``
- \* ``_parse_legacy_dist()``
- \* ``_parse_channel()``

- \* ``_PARSE_CACHE``
- \* ``_parse_spec_str()``
- \* ``MatchInterface``
  - \* ``MatchInterface.raw_value``
  - \* ``MatchInterface.exact_value``
  - \* ``MatchInterface.match()``
  - \* ``MatchInterface.matches()``
  - \* ``MatchInterface.merge()``
  - \* ``MatchInterface.union()``
- \* ``_StrMatchMixin``
  - \* ``_StrMatchMixin.exact_value``
  - \* ``_StrMatchMixin.__str__()``
  - \* ``_StrMatchMixin.__repr__()``
  - \* ``_StrMatchMixin.__eq__()``
  - \* ``_StrMatchMixin.__hash__()``
- \* ``ExactStrMatch``
  - \* ``ExactStrMatch.__slots__``
  - \* ``ExactStrMatch.match()``
- \* ``ExactLowerStrMatch``
  - \* ``ExactLowerStrMatch.match()``
- \* ``GlobStrMatch``
  - \* ``GlobStrMatch.exact_value``
  - \* ``GlobStrMatch.matches_all``
  - \* ``GlobStrMatch.__slots__``
  - \* ``GlobStrMatch.match()``
  - \* ``GlobStrMatch.merge()``
- \* ``GlobLowerStrMatch``

- \* ``SplitStrMatch``
  - \* ``SplitStrMatch.exact_value``
  - \* ``SplitStrMatch.__slots__``
  - \* ``SplitStrMatch._convert()``
  - \* ``SplitStrMatch.match()``
  - \* ``SplitStrMatch.__repr__()``
  - \* ``SplitStrMatch.__str__()``
  - \* ``SplitStrMatch.__eq__()``
  - \* ``SplitStrMatch.__hash__()``
- \* ``FeatureMatch``
  - \* ``FeatureMatch.exact_value``
  - \* ``FeatureMatch.__slots__``
  - \* ``FeatureMatch._convert()``
  - \* ``FeatureMatch.match()``
  - \* ``FeatureMatch.__repr__()``
  - \* ``FeatureMatch.__str__()``
  - \* ``FeatureMatch.__eq__()``
  - \* ``FeatureMatch.__hash__()``
- \* ``ChannelMatch``
  - \* ``ChannelMatch.match()``
  - \* ``ChannelMatch.__str__()``
  - \* ``ChannelMatch.__repr__()``
- \* ``CaseInsensitiveStrMatch``
  - \* ``CaseInsensitiveStrMatch.match()``
- \* ``_implementors``

guide/api/conda/models/match\_spec/index.rst)

[ \_\_Show Source](../../../../\_sources/dev-

guide/api/conda/models/match\_spec/index.rst.txt)

© Copyright 2017, Anaconda, Inc.

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.