
rebase-helper

Release 0.18.1.dev21+g8685abf

Sep 16, 2019

Contents

1	How to get it running?	3
2	How to use it?	5
3	What do I get from it?	7
4	How does it work?	9
5	Video presentation	11
5.1	User Guide	12
5.2	API	17
5.3	Change Log	43
	Python Module Index	57
	Index	59

There are several steps that need to be done when rebasing a package. The goal of **rebase-helper** is to automate most of these steps.

CHAPTER 1

How to get it running?

rebase-helper is packaged in Fedora, so you can just install it with **dnf**.

If you wish to use the latest codebase, consult [installation instructions](#).

CHAPTER 2

How to use it?

After installation, execute **rebase-helper** from a directory containing SPEC file, sources and patches (usually cloned dist-git repository).

Without any arguments or configuration **rebase-helper** will attempt to determine the latest upstream version automatically. If that fails, or if you wish to rebase to some different version, you can specify it explicitly as an argument:

```
$ rebase-helper 3.1.10
```

or you can pass source tarball filename instead:

```
$ rebase-helper foo-4.2.tar.gz
```

For complete CLI reference see [usage](#).

Alternatively, you can run **rebase-helper** in a container:

```
$ docker run -it -e PACKAGE=foo rebasehelper/rebase-helper:latest
```

See [docker reference](#) for more information.

What do I get from it?

rebase-helper always creates *rebase-helper-results* directory containing the following items:

Path	Description
<i>report.txt</i>	summary report with all important information
<i>changes.patch</i>	diff against original files, directly applicable to dist-git repo
<i>logs/</i>	log files of various verbosity levels
<i>rebased-sources/</i>	git repository with all modified files
<i>checkers/</i>	reports from individual checkers that were run
<i>old-build/</i>	logs and results of old (original) version build
<i>new-build/</i>	logs and results of new (rebased) version build

How does it work?

The following steps describe a rebase process:

- **Preparation**

- *rebase-helper-workspace* and *rebase-helper-results* directories are created
- original SPEC file is copied to *rebase-helper-results/rebased-sources* directory and its Version tag is modified

- **Getting sources**

- old and new source tarballs are downloaded and extracted to *rebase-helper-workspace* directory
- old sources are downloaded from lookaside cache if possible

- **Downstream patches**

- new git repository is initialized and the old sources are extracted and committed
- each downstream patch is applied and changes introduced by it are committed
- new sources are extracted and added as a remote repository
- *git-rebase* is used to rebase the commits on top of new sources
- original patches are modified/deleted accordingly
- resulting files are stored in *rebase-helper-results/rebased-sources*
- diff against original files is saved to *rebase-helper-results/changes.patch*

- **Build**

- old and new source RPMs are created and built with selected build tool
- old SRPM and RPMs can also be downloaded from Koji to speed up the rebase

- **Comparison**

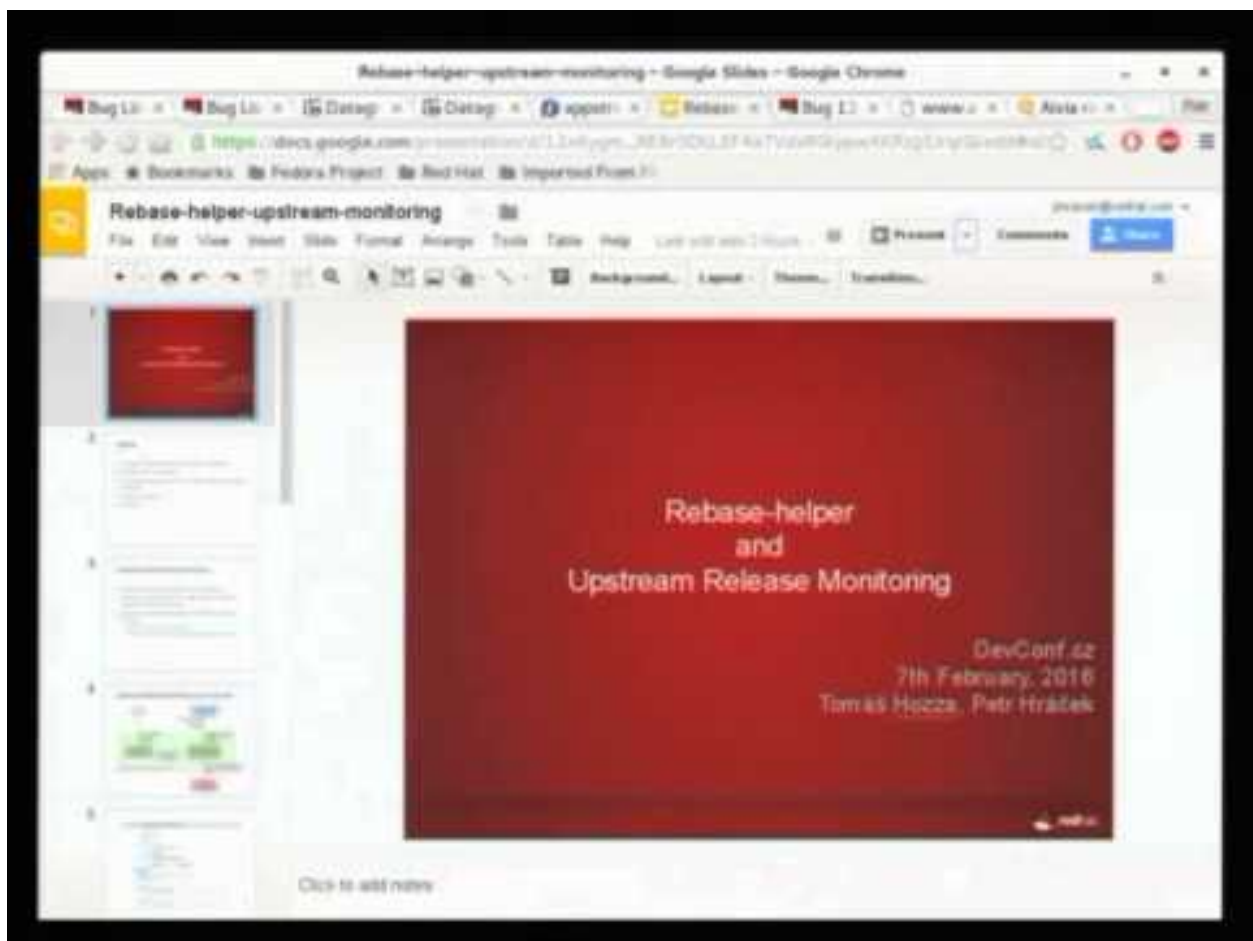
- multiple checker tools are run against both sets of packages and their output is stored in *rebase-helper-results/checkers* directory

- Cleanup
 - *rebase-helper-workspace* directory is removed

CHAPTER 5

Video presentation

At DevConf.cz 2016, Tomáš Hozza, Petr Hráček presented on rebase-helper.



5.1 User Guide

5.1.1 Installation

rebase-helper is packaged in Fedora, so you can just install it with **dnf**.

If you can't or don't want to use **rebase-helper** package, you have to install, apart from Python requirements listed in `get_requirements` function in `setup.py`, the following dependencies:

Dependency	Package name (in Fedora)	Notes
git	git	for rebasing of downstream patches
rpmbuild	rpm-build	for building SRPM and for rpmbuild build tool
mock	mock	for mock build tool, optional
koji	koji	for koji build tool, optional
rpmdiff	rpmlint	for rpmdiff checker, optional
abipkgdiff	libabigail	for abipkgdiff checker, optional
pkgdiff	pkgdiff	for pkgdiff checker, optional
csmock	csmock	for csmock checker, optional
licensecheck	licensecheck	for licensecheck checker, optional

rebase-helper is also published on [PyPI](#).

5.1.2 Logging

Since version 0.18.0 **rebase-helper** uses standard module-level loggers.

There are also 3 loggers with a special purpose:

Logger name	Purpose
<code>rebasehelper.traceback</code>	traceback generated by unhandled exceptions
<code>rebasehelper.summary</code>	messages forming the rebase summary
<code>rebasehelper.report</code>	messages forming the textual rebase report (used by <code>text</code> output tool)

rebase-helper uses `rebasehelper.logger.CustomLogger` logger class which provides extra logging levels.

`rebasehelper.logger.LoggerHelper` class provides 3 utility methods to manage default handlers:

- `rebasehelper.logger.LoggerHelper.create_stream_handlers()` sets up the default console handlers for `rebasehelper` and `rebasehelper.summary` loggers.
- `rebasehelper.logger.LoggerHelper.create_file_handlers()` sets up the default file handlers for `rebasehelper` logger, with INFO, VERBOSE and DEBUG levels. The respective log files are located in `rebase-helper-results/logs/`.
- `rebasehelper.logger.LoggerHelper.remove_file_handlers()` removes specified file handlers from `rebasehelper` logger.

Examples


```
# use a single module and get logs from it

import logging

from rebasehelper.specfile import SpecFile

logger = logging.getLogger('rebasehelper.specfile')
logger.addHandler(logging.StreamHandler())

spec = SpecFile('test.spec')
spec.set_release_number(2)
```

```
# run a complete rebase with default log handlers

from rebasehelper.logger import LoggerHelper
from rebasehelper.config import Config
from rebasehelper.cli import CLI
from rebasehelper.application import Application

LoggerHelper.create_stream_handlers()
config = Config()
cli = CLI()
config.merge(cli)
exec_dir, res_dir = Application.setup(config)
# default file handlers are automatically created and removed by Application instance,
# unless disabled by passing create_logs=False
app = Application(config, exec_dir, res_dir)
app.run()
```

5.1.3 Rebasing in container

rebase-helper can be run in Docker container. The package to be rebased has to be specified in `PACKAGE` environment variable. Alternatively, you can set `REPOSITORY` environment variable and point it to URL of any dist-git repository. In both cases, you can reference a specific branch, tag or commit by appending it to the package name or the repository URL:

```
$ docker run -it -e PACKAGE=foo#branch=f26 rebasehelper/rebase-helper:latest
--outputtool json
```

Results of the rebase will be stored in an exported volume.

5.1.4 Usage

SYNOPSIS

```
rebase-helper [-h] [--version] [-v] [--color {always,never,auto}]
  [--background {dark,light,auto}] [--results-dir RESULTS_DIR]
  [--workspace-dir WORKSPACE_DIR]
  [--buildtool {copr,koji,mock,rpmbuild}]
  [--srpm-buildtool {mock,rpmbuild}]
  [--pkgcomparetool
  {abipkgdiff,csmock,licensecheck,pkgdiff,rpmdiff,sonamecheck}]
  [--outputtool {json,text}]
```

```
[--versioneer {anitya,cpan,hackage,npmjs,pypi,rubygems}]
[--versioneer-blacklist {anitya,cpan,hackage,npmjs,pypi,rubygems}]
[--spec-hook-blacklist
{commit-hash-updater,escape-macros,paths-to-rpm-macros,pypi-url-fix,
replace-old-version,ruby-helper,typo-fix}]
[--build-log-hook-blacklist {files}] [--non-interactive]
[--favor-on-conflict {downstream,upstream,off}]
[--not-download-sources] [-w] [--apply-changes]
[--disable-inapplicable-patches] [--skip-version-check]
[--update-sources] [--skip-upload] [--force-build-log-hooks]
[--builds-nowait] [--build-tasks OLD_TASK,NEW_TASK]
[--builder-options BUILDER_OPTIONS]
[--srpm-builder-options SRPM_BUILDER_OPTIONS]
[--changelog-entry CHANGELOG_ENTRY] [--config-file CONFIG_FILE]
[--replace-old-version-with-macro] [SOURCES]
```

DESCRIPTION

rebase-helper is a tool which helps package maintainers to rebase their packages to latest upstream versions.

It should be executed from a directory containing spec file, sources and patches (usually cloned dist-git repository).

The new version is specified by *SOURCES* argument, which can be either version number or filename of the new source archive. Starting with version 0.10.0, this argument can be omitted and the new version determined automatically using one of available *versioneers*.

OPTIONS

Positional arguments

SOURCES

version number or filename of the new source archive

Optional arguments

-h, --help

show help message and exit

--version

show rebase-helper version and exit

-v, --verbose

be more verbose

--color {always,never,auto}

colorize the output, defaults to auto

--background {dark,light,auto}

use color scheme for the given background, defaults to auto

--results-dir RESULTS_DIR

location where the rebase-helper-results directory will be created

```

--workspace-dir WORKSPACE_DIR
    location where the rebase-helper-workspace directory will be created

--buildtool {copr,koji,mock,rpmbuild}
    build tool to use, defaults to mock

--srpm-buildtool {mock,rpmbuild}
    SRPM build tool to use, defaults to rpmbuild

--pkgcomparetool {abipkgdiff,csmock,licensecheck,pkgdiff,rpmdiff,sonamecheck}
    set of tools to use for package comparison, defaults to abipkgdiff,licensecheck,pkgdiff,
    rpmdiff,sonamecheck if available

--outputtool {json,text}
    tool to use for formatting rebase output, defaults to text

--versioneer {anitya,cpan,hackage,npmjs,pypi,rubygems}
    tool to use for determining latest upstream version

--versioneer-blacklist {anitya,cpan,hackage,npmjs,pypi,rubygems}
    prevent specified versioneers from being run

--spec-hook-blacklist {commit-hash-updater,escape-macros,paths-to-rpm-macros,pypi-url-fix,
    prevent specified spec hooks from being run

--build-log-hook-blacklist {files}
    prevent specified build log hooks from being run

--non-interactive
    do not interact with user

--favor-on-conflict {downstream,upstream,off}
    favor downstream or upstream changes when conflicts appear

--not-download-sources
    do not download sources

-w, --keep-workspace
    do not remove workspace directory after finishing

--apply-changes
    apply changes.patch after a successful rebase

--disable-inapplicable-patches
    disable inapplicable patches in rebased SPEC file

--skip-version-check
    force rebase even if current version is newer than requested version

--update-sources
    update "sources" file and upload new sources to lookaside cache

--skip-upload
    skip uploading new sources to lookaside cache

--force-build-log-hooks
    enforce running of build log hooks (even in non-interactive mode)

--builds-nowait
    do not wait for remote builds to finish

--build-tasks OLD_TASK,NEW_TASK
    comma-separated remote build task ids

```

- builder-options** BUILDER_OPTIONS
enable arbitrary local builder option(s), enclose BUILDER_OPTIONS in quotes to pass more than one
- srpm-builder-options** SRPM_BUILDER_OPTIONS
enable arbitrary local srpm builder option(s), enclose SRPM_BUILDER_OPTIONS in quotes to pass more than one
- changelog-entry** CHANGELOG_ENTRY
text to use as changelog entry, can contain RPM macros, which will be expanded
- config-file** CONFIG_FILE
path to a configuration file, defaults to \$XDG_CONFIG_HOME/rebase-helper.cfg
- replace-old-version-with-macro**
replace old version string with %{version} instead of new version string

5.1.5 Writing plugins

Starting with version 0.10.0, **rebase-helper** is extensible through plugins.

You can implement your own build tool, checker, output tool, SPEC hook, build log hook or versioneer. All you have to do is to derive your plugin from corresponding base class, implement all necessary methods and register it using one of the following entry points:

Plugin type	Entry point	Base class
build tool	rebasehelper. build_tools	rebasehelper.plugins.build_tools.rpm. BuildToolBase
SRPM build tool	rebasehelper. srpm_build_tools	rebasehelper.plugins.build_tools.srpm. SRPMBuildToolBase
checker	rebasehelper.checkers	rebasehelper.plugins.checkers.BaseChecker
output tool	rebasehelper. output_tools	rebasehelper.plugins.output_tools. BaseOutputTool
SPEC hook	rebasehelper. spec_hooks	rebasehelper.plugins.spec_hooks. BaseSpecHook
build log hook	rebasehelper. build_log_hooks	rebasehelper.plugins.build_log_hooks. BaseBuildLogHook
versioneer	rebasehelper. versioneers	rebasehelper.plugins.versioneers. BaseVersioneer

Example

Listing 1: my_spec_hook/__init__.py

```

from rebasehelper.plugins.spec_hooks import BaseSpecHook

class MySpecHook(BaseSpecHook):

    NAME = 'MySpecHook'

    @classmethod
    def get_name(cls):
        return cls.NAME

```

(continues on next page)

(continued from previous page)

```

@classmethod
def run(cls, spec_file, rebase_spec_file):
    """
    This method is called after original SPEC file is processed

    :param spec_file: SpecFile object representing original SPEC file
    :param rebase_spec_file: SpecFile object representing rebased SPEC file
    """
    rebase_spec_file.spec_content.section('%package').insert(0, '# processed by {}
↪\n'.format(cls.NAME))
    rebase_spec_file.save()

```

Listing 2: setup.py

```

from setuptools import setup

setup(
    name='MySpecHook',
    version='0.1',
    description='Custom SPEC hook for rebase-helper',
    author='John Doe',
    install_requires=['rebasehelper>=0.10.0'],
    packages=['my_spec_hook'],
    entry_points={
        'rebasehelper.spec_hooks': ['my_spec_hook = my_spec_hook:MySpecHook']
    }
)

```

5.2 API

5.2.1 Helpers

Console helper module

class `rebasehelper.helpers.console_helper.ConsoleHelper`

Class for interacting with the command line.

class `Capturer` (*stdout=False, stderr=False*)

ContextManager for capturing stdout/stderr

static color_is_light (*rgb, bit_width*)

Determines whether a color is light or dark.

Parameters

- **rgb** (*tuple*) – RGB tuple.
- **bit_width** – Number of bits defining the RGB.

Returns Whether a color is light or dark.

Return type `bool`

classmethod `cprint` (*message, fg=None, bg=None, style=None*)

Prints colored output if possible.

Parameters

- **message** (*str*) – String to be printed out.
- **fg** (*str*) – Foreground color.
- **bg** (*str*) – Background color.
- **style** (*str*) – Style to be applied to the printed message. Possible styles: bold, faint, italic, underline, blink, blink2, negative, concealed, crossed. Some styles may not be supported by every terminal, e.g. 'blink'. Multiple styles should be connected with a '+', e.g. 'bold+italic'.

static detect_background()

Detects terminal background color and decides whether it is light or dark.

Returns Whether to use dark or light color scheme.

Return type str

static exchange_control_sequence(query, timeout=0.05)

Captures a response of a control sequence from STDIN.

Parameters

- **query** (*str*) – Control sequence.
- **timeout** (*int, float*) – Time given to the terminal to react.

Returns Response of the terminal.

Return type str

static parse_rgb_device_specification(specification)

Parses RGB device specification.

Parameters **specification** (*str*) – RGB device specification.

Returns If the specification follows correct format, the first element is RGB tuple and the second is bit width of the RGB. Otherwise, both elements are None.

Return type tuple

classmethod should_use_colors(conf)

Determines whether ANSI colors should be used for CLI output.

Parameters **conf** (*rebasehelper.config.Config*) – Configuration object with arguments from the command line.

Returns Whether colors should be used.

Return type bool

use_colors = False

Copr helper module

```
class rebasehelper.helpers.copr_helper.CoprHelper
```

```
classmethod build(client, project, srpm)
```

```
classmethod create_project(client, project, chroot, description, instructions)
```

```
classmethod download_build(client, build_id, destination)
```

```

classmethod get_build_status (client, build_id)
classmethod get_build_url (client, build_id)
classmethod get_client ()
classmethod watch_build (client, build_id)

```

Download helper module

```
class rebasehelper.helpers.download_helper.DownloadHelper
```

Class for downloading files and performing HTTP requests.

```
static download_file (url, destination_path, blocksize=8192)
```

Downloads a file from HTTP, HTTPS or FTP URL.

Parameters

- **url** (*str*) – URL to be downloaded.
- **destination_path** (*str*) – Path to where the downloaded file will be stored.
- **blocksize** (*int*) – Block size in bytes.

```
static progress (download_total, downloaded, start_time)
```

Prints current progress and estimated remaining time of a download to the standard output.

Parameters

- **download_total** (*int*) – Total download size in bytes.
- **downloaded** (*int*) – Size of the already downloaded portion of a file in bytes.
- **start_time** (*float*) – Time when the download started in seconds since epoch.

```
static request (url, **kwargs)
```

Performs an HTTP request or an FTP RETR command.

Parameters

- **url** (*str*) – HTTP, HTTPS or FTP URL.
- ****kwargs** – Keyword arguments to be passed to requests.session.get().

Returns Response object.

Return type requests.Response

Git helper module

```
class rebasehelper.helpers.git_helper.GitHelper
```

Class which operates with git repositories

```
GIT_USER_EMAIL = 'rebase-helper@localhost.local'
```

```
GIT_USER_NAME = 'rebase-helper'
```

```
classmethod get_email ()
```

```
classmethod get_user ()
```

```
classmethod run_mergetool (repo)
```

Input helper module

class `rebasehelper.helpers.input_helper.InputHelper`

Class for command line interaction with the user.

static `get_message` (*message*, *default_yes=True*, *any_input=False*)

Prompts a user with yes/no message and gets the response.

Parameters

- **message** (*str*) – Prompt string.
- **default_yes** (*bool*) – If the default value should be YES.
- **any_input** (*bool*) – Whether to return default value regardless of input.

Returns True or False, based on user's input.

Return type bool

Koji Helper module

class `rebasehelper.helpers.koji_helper.KojiHelper`

classmethod `create_session` (*login=False*, *profile='koji'*)

Creates new Koji session and immediately logs in to a Koji hub.

Parameters

- **login** (*bool*) – Whether to perform a login.
- **profile** (*str*) – Koji profile to use.

Returns Newly created session instance.

Return type `koji.ClientSession`

Raises `RebaseHelperError` – If login failed.

classmethod `display_task_results` (*tasks*)

Prints states of Koji tasks.

Parameters *tasks* (*list*) – List of `koji.TaskWatcher` instances.

classmethod `download_build` (*session*, *build_id*, *destination*, *arches*)

Downloads RPMs and logs of a Koji build.

Parameters

- **session** (*koji.ClientSession*) – Active Koji session instance.
- **build_id** (*str*) – Koji build ID.
- **destination** (*str*) – Path where to download files to.
- **arches** (*list*) – List of architectures to be downloaded.

Returns List of downloaded RPMs and list of downloaded logs.

Return type tuple

Raises `DownloadError` – If download failed.

classmethod `download_task_results` (*session*, *tasklist*, *destination*)

Downloads packages and logs of finished Koji tasks.

Parameters

- **session** (*koji.ClientSession*) – Active Koji session instance.
- **tasklist** (*list*) – List of task IDs.
- **destination** (*str*) – Path where to download files to.

Returns List of downloaded RPMs and list of downloaded logs.

Return type tuple

Raises DownloadError – If download failed.

functional = False

classmethod **get_build** (*session, package, version*)

Looks up Koji build of a specific version of a package.

Parameters

- **session** (*koji.ClientSession*) – Active Koji session instance.
- **package** (*str*) – Package name.
- **version** (*str*) – Package version.

Returns Found latest package version and Koji build ID.

Return type tuple

classmethod **get_latest_build** (*session, package*)

Looks up latest Koji build of a package.

Parameters

- **session** (*koji.ClientSession*) – Active Koji session instance.
- **package** (*str*) – Package name.

Returns Found latest package version and Koji build ID.

Return type tuple

classmethod **get_old_build_info** (*package_name, package_version*)

Gets old build info from Koji.

Parameters

- **package_name** (*str*) – Package name from specfile.
- **package_version** (*str*) – Package version from specfile.

Returns Koji build id, package version.

Return type tuple

classmethod **get_task_url** (*session, task_id*)

classmethod **upload_srpm** (*session, srpm*)

Uploads SRPM to a Koji hub.

Parameters

- **session** (*koji.ClientSession*) – Active Koji session instance.
- **srpm** (*str*) – Valid path to SRPM.

Returns Remote path to the uploaded SRPM.

Return type list

classmethod `purge_macro` (*macro: str*) → None

static `substitute_path_with_macros` (*path, macros*)

Substitutes parts of a path with macros.

Parameters

- **path** (*str*) – Path to be changed.
- **macros** (*list*) – Macros which can be used as a substitution.

Returns Path expressed using macros.

Return type str

Path helper module

class `rebasehelper.helpers.path_helper.PathHelper`

Class for performing path related tasks.

static `file_available` (*filename*)

Checks if the given file exists.

Parameters **filename** (*str*) – Path to the file.

Returns Whether the file exists.

Return type bool

static `find_all_files` (*top_path, pattern*)

Recursively searches for all files matching the given pattern.

Parameters

- **top_path** (*str*) – Directory where to start the search.
- **pattern** (*str*) – Filename pattern.

Returns List containing absolute paths to all found files.

Return type list

static `find_all_files_current_dir` (*top_path, pattern*)

Searches for all files that match the given pattern inside a directory.

Parameters

- **top_path** (*str*) – Directory where to start the search.
- **pattern** (*str*) – Filename pattern.

Returns List containing absolute paths to all found files.

Return type list

static `find_first_dir_with_file` (*top_path, pattern*)

Recursively searches for a directory containing a file that matches the given pattern.

Parameters

- **top_path** (*str*) – Directory where to start the search.
- **pattern** (*str*) – Filename pattern.

Returns Full path to the directory containing the first occurrence of the searched file. None if there is no file matching the pattern.

Return type str

static find_first_file (*top_path*, *pattern*, *recursion_level=None*)
Recursively searches for a file that matches the given pattern.

Parameters

- **top_path** (*str*) – Directory where to start the search.
- **pattern** (*str*) – Filename pattern.
- **recursion_level** (*int*) – How deep in the directory tree the search can go.

Returns Path to the file matching the pattern or None if there is no file matching the pattern.

Return type str

static get_temp_dir ()
Creates a new temporary directory.

Returns Path to the created directory.

Return type str

Process helper module

class rebasehelper.helpers.process_helper.**ProcessHelper**
Class for executing subprocesses.

DEV_NULL = '/dev/null'

static run_subprocess (*cmd*, *input_file=None*, *output_file=None*, *ignore_stderr=False*)
Runs the specified command in a subprocess.

Parameters

- **cmd** (*iterable*) – A sequence of program arguments.
- **input_file** (*str*, *typing.BytesIO*) – File to read the input from.
- **output_file** (*str*, *typing.BytesIO*) – File to write the output of the command to.
- **ignore_stderr** (*bool*) – Whether to ignore stderr output.

Returns Exit code of the subprocess.

Return type int

static run_subprocess_cwd (*cmd*, *cwd=None*, *input_file=None*, *output_file=None*, *ignore_stderr=False*, *shell=False*)
Runs the specified command in a subprocess in a different working directory.

Parameters

- **cmd** (*iterable*) – A sequence of program arguments.
- **cwd** (*str*) – Working directory for the command.
- **input_file** (*str*, *typing.BytesIO*) – File to read the input from.
- **output_file** (*str*, *typing.BytesIO*) – File to write the output of the command to.

- **ignore_stderr** (*bool*) – Whether to ignore stderr output.
- **shell** (*bool*) – Whether to run the command in a shell.

Returns Exit code of the subprocess.

Return type int

```
static run_subprocess_cwd_env(cmd, cwd=None, env=None, input_file=None, out-
                             put_file=None, ignore_stderr=False, shell=False)
```

Runs the specified command in a subprocess in a different working directory with a redefined environment.

Parameters

- **cmd** (*iterable*) – A sequence of program arguments.
- **cwd** (*str*) – Working directory for the command.
- **env** (*dict*) – Environment variables for the new process.
- **input_file** (*str*, *typing.BytesIO*) – File to read the input from.
- **output_file** (*str*, *typing.BytesIO*) – File to write the output of the command to.
- **ignore_stderr** (*bool*) – Whether to ignore stderr output.
- **shell** (*bool*) – Whether to run the command in a shell.

Returns Exit code of the subprocess.

Return type int

```
static run_subprocess_env(cmd, env=None, input_file=None, output_file=None, ig-
                           nore_stderr=False, shell=False)
```

Runs the specified command in a subprocess with a redefined environment.

Parameters

- **cmd** (*iterable*) – A sequence of program arguments.
- **env** (*dict*) – Environment variables for the new process.
- **input_file** (*str*, *typing.BytesIO*) – File to read the input from.
- **output_file** (*str*, *typing.BytesIO*) – File to write the output of the command to.
- **ignore_stderr** (*bool*) – Whether to ignore stderr output.
- **shell** (*bool*) – Whether to run the command in a shell.

Returns Exit code of the subprocess.

Return type int

RPM helper module

```
class rebasehelper.helpers.rpm_helper.RpmHeader (hdr: <sphinx.ext.autodoc.importer._MockObject
                                                object at 0x7f9977a04320>)
```

```
class rebasehelper.helpers.rpm_helper.RpmHelper
    Class for working with RPM database and packages.
```

```
ARCHES = []
```

static all_packages_installed (*pkg_names=None*)
Checks if all specified packages are installed.

Parameters **pkg_names** (*list*) – List of package names to check.

Returns True if all packages are installed, False otherwise.

Return type bool

static get_arches ()
Gets list of all known architectures

static get_header_from_rpm (*rpm_name*)
Gets an RPM header from the given RPM package.

Parameters **rpm_name** (*str*) – Path to the package.

Returns Header object obtained from the package.

Return type *RpmHeader*

classmethod get_rpm_spec (*path*) → <sphinx.ext.autodoc.importer._MockObject object at 0x7f9977a04860>

static install_build_dependencies (*spec_path=None, assume_yes=False*)
Installs build dependencies of a package using dnf.

Parameters

- **spec_path** (*str*) – Absolute path to the SPEC file.
- **assume_yes** (*bool*) – Whether to automatically answer ‘yes’ to all questions.

Returns Exit code of the subprocess run.

Return type int

static is_package_installed (*pkg_name=None*)
Checks whether a package is installed.

Parameters **pkg_name** (*str*) – Name of the package.

Returns Whether the package is installed.

Return type bool

classmethod parse_spec (*path, flags=None*)

classmethod split_nevra (*s*)
Splits string into name, epoch, version, release and arch components

5.2.2 Plugins

Build Log Hooks package

class rebasehelper.plugins.build_log_hooks.**BaseBuildLogHook**
Base class for a build log hook.

CATEGORIES = []

classmethod format (*data*)
Formats build log hook output to a readable text form.

classmethod merge_two_results (*old, new*)

classmethod `run` (*spec_file*, *rebase_spec_file*, *results_dir*, ***kwargs*)

Runs the build log hook.

Parameters

- **spec_file** (`rebasehelper.specfile.SpecFile`) – Original SpecFile object.
- **rebase_spec_file** (`rebasehelper.specfile.SpecFile`) – Rebased SpecFile object.
- **kwargs** (*dict*) – Keyword arguments from instance of Application.

Returns The first element is a dict containing changes, the second is a bool whether the build process should be restarted.

Return type tuple

class `rebasehelper.plugins.build_log_hooks.BuildLogHookCollection` (*entrypoint*:
str, *manager*:
PluginManager)

run (*spec_file*, *rebase_spec_file*, *non_interactive*, *force_build_log_hooks*, ***kwargs*)

Runs all non-blacklisted build log hooks.

Parameters

- **spec_file** (`rebasehelper.specfile.SpecFile`) – Original SpecFile object.
- **rebase_spec_file** (`rebasehelper.specfile.SpecFile`) – Rebased SpecFile object.
- **non_interactive** (*bool*) – Whether rebase-helper is in non-interactive mode.
- **force_build_log_hooks** (*bool*) – Whether to run the hooks even in non-interactive mode.
- **kwargs** (*dict*) – Keyword arguments from instance of Application.

Returns Whether build log hooks made some changes to the SPEC file.

Return type bool

Build Tools package

class `rebasehelper.plugins.build_tools.BuildTemporaryEnvironment` (*sources*,
patches,
spec, *re-*
sults_dir)

Class representing temporary environment.

TEMPDIR_RESULTS = 'TEMPDIR_RESULTS'

TEMPDIR_SOURCES = 'TEMPDIR_SOURCES'

TEMPDIR_SPEC = 'TEMPDIR_SPEC'

TEMPDIR_SPECS = 'TEMPDIR_SPECS'

class `rebasehelper.plugins.build_tools.MockTemporaryEnvironment` (*sources*,
patches, *spec*,
results_dir)

Class representing temporary environment for MockBuildTool.

```
class rebasehelper.plugins.build_tools.RpmbuildTemporaryEnvironment (sources,  
patches,  
spec, re-  
sults_dir)
```

Class representing temporary environment for RpmbuildBuildTool.

```
TEMPDIR_BUILD = 'TEMPDIR_BUILD'  
TEMPDIR_BUILDROOT = 'TEMPDIR_BUILDROOT'  
TEMPDIR_RPMBUILD = 'TEMPDIR_RPMBUILD'  
TEMPDIR_RPMS = 'TEMPDIR_RPMS'  
TEMPDIR_SRPMS = 'TEMPDIR_SRPMS'
```

```
rebasehelper.plugins.build_tools.check_mock_privileges () → bool
```

Checkers package

```
class rebasehelper.plugins.checkers.BaseChecker  
Base class of package checkers.
```

DEFAULT

If True, the checker is run by default.

Type bool

CATEGORY

Category which determines when the checker is run.

Type *CheckerCategory*

results_dir

Path where the results are stored.

Type str

```
CATEGORY = None
```

```
DEFAULT = False
```

```
classmethod format (data)
```

Formats checker output to a readable text form.

```
classmethod get_category ()
```

```
classmethod get_checker_output_dir_short ()
```

Return short version of checker output directory

```
classmethod get_important_changes (checker_output)
```

Each checker has an opportunity to highlight some of its output. This function is optional, as not all checkers provide output with critical information.

Parameters **checker_output** (*dict*) – Dictionary with output from the given checker.

Returns List of strings to be output to CLI as warning messages.

Return type list

```
classmethod get_underlined_title (text, separator='')
```

```
classmethod is_available ()
```

```
results_dir = None
```



```
classmethod run_check (results_dir, **kwargs)
    Perform the check itself and return results.
```

```
class rebasehelper.plugins.checkers.CheckerCategory
    An enumeration.
```

```
RPM = 3
```

```
SOURCE = 1
```

```
SRPM = 2
```

```
class rebasehelper.plugins.checkers.CheckerCollection (entrypoint: str, manager: PluginManager)
```

```
    Class representing the process of running various checkers on final packages.
```

```
get_default_plugins (return_one: bool = False) → Union[List[Type[rebasehelper.plugins.plugin.Plugin]],
    Type[rebasehelper.plugins.plugin.Plugin]]
```

```
get_supported_plugins () → List[Type[rebasehelper.plugins.plugin.Plugin]]
```

```
run (results_dir: str, checker_name: str, **kwargs) → Optional[Dict[str, Any]]
    Runs a particular checker and returns the results.
```

Parameters

- **results_dir** – Path to a directory in which the checker should store the results.
- **checker_name** – Name of the checker to be run.

Raises `NotImplementedError` – If a checker with the given name doesn't exist.

Returns Results of the checker.

Output Tools package

```
class rebasehelper.plugins.output_tools.BaseOutputTool
```

```
    Base class for OutputTools. print_cli_summary must be overridden in order to produce different CLI output
```

```
DEFAULT = False
```

```
EXTENSION = ''
```

```
classmethod get_report_path (app)
```

```
classmethod prepend_results_dir_name (*path_members)
```

```
classmethod print_cli_summary (app)
```

```
    Print report of the rebase
```

Parameters **app** – Application instance

```
classmethod print_important_checkers_output ()
```

```
    Iterates over all checkers output to highlight important checkers warning
```

```
classmethod print_patches_cli ()
```

```
    Print info about patches
```

```
classmethod print_patches_section_cli (logger_method, patch_type)
```

```
    Print info about one of the patches key section
```

Parameters

- **logger_method** – method to be used for logging
- **patch_type** – string containing key for the `patch_dict`

```
classmethod print_report_file_path()
```

Print path to the report file

```
classmethod run(logs, app)
```

```
class rebasehelper.plugins.output_tools.OutputToolCollection(
    entrypoint: str,
    manager: PluginManager)
```

Class representing the process of running various output tools.

```
run(tool, logs=None, app=None)
```

Runs specified output tool.

Parameters

- **tool** (*str*) – Tool to run.
- **logs** (*list*) – Build logs containing information about the failed rebase.
- **app** (`rebasehelper.application.Application`) – Application class instance.

Plugin Collection module

```
class rebasehelper.plugins.plugin_collection.PluginCollection(
    entrypoint: str,
    manager: PluginManager)
```

```
get_all_plugins() → List[Type[rebasehelper.plugins.plugin.Plugin]]
```

```
get_default_plugins(return_one: bool = False) → Union[Type[rebasehelper.plugins.plugin.Plugin],
    List[Type[rebasehelper.plugins.plugin.Plugin]]]
```

```
get_options() → List[Union[Dict[str, Any], List[Dict[str, Any]]]]
```

Gets options of all plugins of one type.

Returns List of plugins' options.

Return type list

```
get_plugin(tool: str) → Type[rebasehelper.plugins.plugin.Plugin]
```

```
get_supported_plugins() → List[Type[rebasehelper.plugins.plugin.Plugin]]
```

Plugin Manager module

```
class rebasehelper.plugins.plugin_manager.PluginManager
```

```
COLLECTIONS = [<class 'rebasehelper.plugins.build_log_hooks.BuildLogHookCollection'>, ...]
```

```
get_options()
```

Gets options of all plugins.

Returns List of plugins' options.

Return type list

RPM Build Tools package

class `rebasehelper.plugins.build_tools.rpm.BuildToolBase`

Build tool base class.

DEFAULT

If True, the build tool is default tool.

Type bool

ACCEPTS_OPTIONS

If True, the build tool accepts additional options passed via `--builder-options`.

Type bool

CREATES_TASKS

If True, the build tool creates remote tasks.

Type bool

ACCEPTS_OPTIONS = False

CREATES_TASKS = False

DEFAULT = False

classmethod `build(spec, results_dir, srpm, **kwargs)`

Build binaries from the sources.

Keyword arguments: `spec` – path to a SPEC file `sources` – list with absolute paths to SOURCES `patches` – list with absolute paths to PATCHES `results_dir` – path to DIR where results should be stored

Returns: dict with: `'srpm'` -> absolute path to SRPM `'rpm'` -> list of absolute paths to RPMs `'logs'` -> list of absolute paths to logs

static `get_builder_options(**kwargs)`

classmethod `get_detached_task(task_id, results_dir)`

Gets packages and logs for specified task

Parameters

- `task_id` – detached task id
- `results_dir` – path to DIR where results should be stored

Returns tuple with: list of absolute paths to RPMs list of absolute paths to logs

classmethod `get_logs()`

Get logs from previously failed build Returns: dict with `'logs'` -> list of absolute paths to logs

classmethod `get_task_info(build_dict)`

Gets information about detached remote task

Parameters `build_dict` – build data

Returns task info

classmethod `prepare(spec, conf)`

Prepare for building.

Parameters `spec` – spec file object

classmethod `wait_for_task(build_dict, task_id, results_dir)`

Waits until specified task is finished

Parameters

- **build_dict** – build data
- **results_dir** – path to DIR where results should be stored

Returns tuple with: list of absolute paths to RPMs list of absolute paths to logs

class `rebasehelper.plugins.build_tools.rpm.BuildToolCollection` (*entrypoint: str, manager: PluginManager*)

Collection of RPM build tools.

SPEC Hooks package

class `rebasehelper.plugins.spec_hooks.BaseSpecHook`

Base class for a spec hook

CATEGORIES = []

classmethod `run` (*spec_file, rebase_spec_file, **kwargs*)

Runs a spec hook.

Parameters

- **spec_file** – Original spec file object
- **rebase_spec_file** – Rebased spec file object
- **kwargs** – Keyword arguments from Application instance

class `rebasehelper.plugins.spec_hooks.SpecHookCollection` (*entrypoint: str, manager: PluginManager*)

Class representing the process of running various spec file hooks.

run (*spec_file, rebase_spec_file, **kwargs*)

Runs all non-blacklisted spec hooks.

Parameters

- **spec_file** (`rebasehelper.specfile.SpecFile`) – Original SpecFile object.
- **rebase_spec_file** (`rebasehelper.specfile.SpecFile`) – Rebased SpecFile object.
- ****kwargs** – Keyword arguments from Application instance.

SRPM Build Tools package

class `rebasehelper.plugins.build_tools.srpm.SRPMBuildToolBase`

SRPM build tool base class.

DEFAULT

If True, the build tool is default tool.

Type bool

DEFAULT = False

classmethod `build` (*spec, results_dir, **kwargs*)

Build SRPM with chosen SRPM Build Tool

Parameters

- **spec** – SpecFile object

- **results_dir** – absolute path to DIR where results should be stored

Returns absolute path to SRPM, list with absolute paths to logs

classmethod `get_logs()`

Get logs from previously failed build Returns: dict with 'logs' -> list of absolute paths to logs

static `get_srpm_builder_options(**kwargs)`

class `rebasehelper.plugins.build_tools.srpm.SRPMBuildToolCollection` (*entrypoint: str, manager: PluginManager*)

Collection of SRPM build tools.

Versioneers package

class `rebasehelper.plugins.versioneers.BaseVersioneer`

Base class for a versioneer

CATEGORIES = []

classmethod `run(package_name)`

Runs a versioneer.

Parameters `package_name` – Name of a package

Returns Latest upstream version of a package

class `rebasehelper.plugins.versioneers.VersioneerCollection` (*entrypoint: str, manager: PluginManager*)

run (*versioneer, package_name, category, versioneer_blacklist=None*)

Runs the specified versioneer or all versioneers subsequently until one of them succeeds.

Parameters

- **versioneer** (*str*) – Name of a versioneer.
- **package_name** (*str*) – Name of a package.
- **category** (*str*) – Package category.
- **versioneer_blacklist** (*list*) – List of versioneers that will be skipped.

Returns Latest upstream version of a package.

Return type str

5.2.3 Application module

class `rebasehelper.application.Application` (*cli_conf: rebasehelper.config.Config, execution_dir: str, results_dir: str, create_logs: bool = True*)

apply_changes ()

build_binary_packages ()

Function calls build class for building packages

build_source_packages ()

static extract_archive (*archive_path*, *destination*)

Extracts given archive into the destination and handle all exceptions.

Parameters

- **archive_path** – path to the archive to be extracted
- **destination** – path to a destination, where the archive should be extracted to

Returns

static extract_sources (*archive_path*, *destination*)

Function extracts a given Archive and returns a full dirname to sources

generate_patch ()

Generates patch to the results_dir containing all needed changes for the rebased package version

get_new_build_logs ()

patch_sources (*sources*)

prepare_next_run (*results_dir*)

prepare_sources ()

Function prepares a sources.

Returns

print_summary (*exception=None*)

Save rebase-helper result and print the summary using output tools. :param exception: Error message from rebase-helper :return:

print_task_info (*builder*)

run ()

run_package_checkers (*results_dir*, ***kwargs*)

Runs checkers on packages and stores results in a given directory.

Parameters

- **results_dir** (*str*) – Path to directory in which to store the results.
- **category** (*str*) – checker type(SOURCE/SRPM/RPM)

Returns None

static setup (*cli_conf*)

5.2.4 Archive module

class rebasehelper.archive.**Archive** (*filename=None*)

Class representing an archive with sources

extract_archive (*path=None*)

Extracts the archive into the given path

Parameters **path** – Path where to extract the archive to.

Returns

classmethod **get_supported_archives** ()

Return list of supported archive types

```

class rebasehelper.archive.ArchiveTypeBase
    Base class for various archive types

    EXTENSION = ''

    classmethod extract (archive=None, filename=None, path=None)
        Extracts the archive into the given path

        Parameters path – Path where to extract the archive to.

        Returns

    classmethod match (filename=None)
        Checks if the filename matches the archive type. If yes, returns True, otherwise returns False.

    classmethod open (filename=None)
        Opens archive with the given filename and returns the proper archive type object.

class rebasehelper.archive.Bz2ArchiveType
    .bz2 archive type

    EXTENSION = '.bz2'

    classmethod extract (archive=None, filename=None, path=None)
        Extracts the archive into the given path

        Parameters path – Path where to extract the archive to.

        Returns

    classmethod open (filename=None)
        Opens archive with the given filename and returns the proper archive type object.

class rebasehelper.archive.TarArchiveType
    .tar archive type

    EXTENSION = '.tar'

class rebasehelper.archive.TarBz2ArchiveType
    .tar.bz2 archive type

    EXTENSION = '.tar.bz2'

class rebasehelper.archive.TarGzArchiveType
    .tar.gz archive type

    EXTENSION = '.tar.gz'

    classmethod extract (archive=None, filename=None, path=None)
        Extracts the archive into the given path

        Parameters path – Path where to extract the archive to.

        Returns

    classmethod open (filename=None)
        Opens archive with the given filename and returns the proper archive type object.

class rebasehelper.archive.TarXzArchiveType
    .tar.xz archive type

    EXTENSION = '.tar.xz'

    classmethod extract (archive=None, filename=None, path=None)
        Extracts the archive into the given path

        Parameters path – Path where to extract the archive to.

```

Returns

classmethod `open` (*filename=None*)

Opens archive with the given filename and returns the proper archive type object.

class `rebasehelper.archive.TgzArchiveType`
.tgz archive type

EXTENSION = `' .tgz '`

class `rebasehelper.archive.ZipArchiveType`
.zip archive type

EXTENSION = `' .zip '`

classmethod `extract` (*archive=None, filename=None, path=None*)

Extracts the archive into the given path

Parameters `path` – Path where to extract the archive to.

Returns

classmethod `match` (*filename=None*)

Checks if the filename matches the archive type. If yes, returns True, otherwise returns False.

classmethod `open` (*filename=None*)

Opens archive with the given filename and returns the proper archive type object.

`rebasehelper.archive.register_archive_type` (*archive*)

5.2.5 CLI module

class `rebasehelper.cli.CLI` (*args=None*)

Class for processing data from commandline

static `build_parser` (*available_choices_only=False*)

class `rebasehelper.cli.CliHelper`

static `run` ()

5.2.6 Exception module

exception `rebasehelper.exceptions.BinaryPackageBuildError` (**args, **kwargs*)

Error indicating failure during the build of binary package.

exception `rebasehelper.exceptions.CheckerNotFound`

Error indicating not being able to find checker binary.

exception `rebasehelper.exceptions.DownloadError`

Exception indicating that download of a file failed.

exception `rebasehelper.exceptions.LookasideCacheError`

Exception indicating a problem in accessing lookaside cache.

exception `rebasehelper.exceptions.ParseError`

exception `rebasehelper.exceptions.RebaseHelperError` (**args, **kwargs*)

Class representing Error raised inside rebase-helper after intentionally catching some expected and well known exception/error.

exception `rebasehelper.exceptions.SourcePackageBuildError` (**args, **kwargs*)
 Error indicating failure during the build of source package

5.2.7 Logger module

class `rebasehelper.logger.ColorizingStreamHandler` (*stream=None*)

colors = {'dark': {10: {'bg': 'default', 'fg': 'brightblack', 'style': None}, 11:

emit (*record*)

Emit a record.

If a formatter is specified, it is used to format the record. The record is then written to the stream with a trailing newline. If exception information is present, it is formatted using `traceback.print_exception` and appended to the stream. If the stream has an 'encoding' attribute, it is used to determine how to do the output to the stream.

set_terminal_background (*background*)

terminal_background = 'dark'

class `rebasehelper.logger.CustomLogger` (*name, level=0*)

HEADING = 26

IMPORTANT = 27

SUCCESS = 25

TRACE = 11

VERBOSE = 12

class `rebasehelper.logger.LoggerHelper`

Helper class for setting up a logger.

static add_file_handler (*logger: logging.Logger, path: str, formatter: Optional[logging.Formatter] = None, level: Optional[int] = None*) → *Optional[logging.FileHandler]*

Adds file handler to the given logger.

Parameters

- **logger** – Logger object to add the handler to.
- **path** – Path to a log file.
- **formatter** – Formatter object used to format logged messages.
- **level** – Severity threshold.

Returns Created file handler instance or None if creation failed.

static add_stream_handler (*logger: logging.Logger, level: Optional[int] = None, formatter: Optional[logging.Formatter] = None*) → `rebasehelper.logger.ColorizingStreamHandler`

Adds stream handler to the given logger.

Parameters

- **logger** – Logger object to add the handler to.
- **level** – Severity threshold.

- **formatter** – Formatter object used to format logged messages.

Returns Created stream handler instance.

classmethod create_file_handlers (*results_dir: str*) → List[logging.FileHandler]
Creates rebase-helper file handlers.

Parameters results_dir – Path to rebase-helper-results directory.

Returns List of created file handler instances.

classmethod create_stream_handlers () → Tuple[rebasehelper.logger.ColorizingStreamHandler, rebasehelper.logger.ColorizingStreamHandler]

classmethod remove_file_handlers (*handlers: List[logging.FileHandler]*) → None
Removes rebase-helper file handlers.

Parameters handlers – List of file handlers to remove.

5.2.8 Patcher module

class rebasehelper.patcher.Patcher

Class for git command used for patching old and new sources

classmethod apply_old_patches ()
Function applies a patch to a old/new sources

classmethod apply_patch (*repo, patch_object*)
Function applies patches to old sources It tries apply patch with am command and if it fails then with command -apply

static decorate_patch_name (*patch_name*)

classmethod extract_patch_name (*message*)

classmethod init_git (*directory*)
Function initialize old and new Git repository

classmethod insert_patch_name (*message, patch_name*)

new_repo = None

new_sources = None

non_interactive = False

old_repo = None

old_sources = None

output_data = None

classmethod patch (*old_dir, new_dir, rest_sources, patches, **kwargs*)
The function can be used for patching one directory against another

patches = []

source_dir = None

classmethod strip_patch_name (*diff, patch_name*)

5.2.9 Specfile module

class `rebasehelper.specfile.PackageCategory`

An enumeration.

`R = re.compile('^R-')`

`haskell = re.compile('^ghc-')`

`nodejs = re.compile('^nodejs-')`

`perl = re.compile('^perl-')`

`php = re.compile('^php-')`

`python = re.compile('^python[23]?-')`

`ruby = re.compile('^rubygem-')`

`rust = re.compile('^rust-')`

class `rebasehelper.specfile.PatchList`

class `rebasehelper.specfile.PatchObject` (*path, index, strip*)

Class represents set of information about patches

`get_patch_name()`

class `rebasehelper.specfile.SpecContent` (*content*)

Class representing content of a SPEC file.

`DISALLOW_INLINE_COMMENTS = ['%package', '%patchlist', '%sourcelist', '%description', '%prep']`

`SECTION_HEADERS = ['%package', '%prep', '%build', '%install', '%check', '%clean', '%prereq']`

classmethod `get_comment_span` (*line: str, section: str*) → Tuple[int, int]

Gets span of a comment depending on the section.

Parameters

- **line** – Line to find the comment in.
- **section** – Section the line is in.

Returns Span of the comment. If no comment is found, both tuple elements are equal to the length of the line for convenient use in a slice.

replace_section (*name, content*)

Replaces content of a section.

In case there are multiple sections with the same name, the first one is replaced.

Parameters

- **name** (*str*) – Section name.
- **content** (*list*) – Section content as a list of lines.

Returns False if section was not found else True.

Return type bool

section (*name*)

Gets content of a section.

In case there are multiple sections with the same name, the first one is returned.

Parameters **name** (*str*) – Section name.

Returns Section content as a list of lines.

Return type list

class `rebasehelper.specfile.SpecFile` (*path*: str, *sources_location*: str = "")

Class representing a SPEC file

copy (*new_path*)

Creates a copy of the current object and copies the SPEC file to a new location.

Parameters **new_path** (*str*) – Path to copy the new SPEC file to.

Returns The created SpecFile instance.

Return type *SpecFile*

download_remote_sources ()

Method that iterates over all sources and downloads ones, which contain URL instead of just a file.

Returns None

static extract_version_from_archive_name (*archive_path*, *source_string* = "")

Method extracts the version from archive name based on the source string from SPEC file. It extracts also an extra version such as 'b1', 'rc1', ...

Parameters

- **archive_path** – archive name or path with archive name from which to extract the version
- **source_string** – Source string from SPEC file used to construct version extraction regex

Returns tuple of strings with (extracted version, extra version) or (None, None) if extraction failed

find_archive_target_in_prep (*archive*)

Tries to find a command that is used to extract the specified archive and attempts to determine target path from it. 'tar' and 'unzip' commands are supported so far.

Parameters **archive** – Path to archive

Returns Target path relative to builddir or None if not determined

get_NVR () → str

get_applied_patches ()

Method returns list of all applied patches.

Returns list of PatchObject

get_archive ()

Method returns the basename of first Source in SPEC file a.k.a. Source0

Returns basename of first Source in SPEC file

Return type str

get_extra_version () → str

get_main_files_section ()

Finds the exact name of the main %files section.

Returns Name of the main files section.

Return type str

get_new_log (*changelog_entry*)
 Constructs a new changelog entry.

Parameters **changelog_entry** (*str*) – Message to use in the entry.

Returns List of lines of the new entry.

Return type list

get_not_used_patches ()
 Method returns list of all unplied patches.

Returns list of PatchObject

get_patches ()
 Method returns list of all applied and not applied patches

Returns list of PatchObject

get_prep_section ()
 Function returns whole prep section

get_release_number () → str

get_setup_dirname ()
 Get dirname from %setup or %autosetup macro arguments

Returns dirname

get_sources ()
 Method returns dictionary with local sources list.

Returns list of Sources with absolute path

Return type list of str

static get_subpackage_name (*files_section*)
 Gets subpackage name based on the %files section.

get_version () → str

is_test_suite_enabled ()
 Returns whether test suite is enabled during the build time

Returns True if enabled or False if not

redefine_release_with_macro (**args, **kwargs*)

reload ()
 Reloads the whole Spec file.

revert_redefine_release_with_macro (**args, **kwargs*)

save ()
 Saves changes made to SpecContent and updates the internal state.

set_extra_version (**args, **kwargs*)

set_release_number (*release*)
 Method to set release number

Parameters **release** –

Returns

set_tag (**args, **kwargs*)

set_version (*version*)

Method to update the version in the SPEC file

Parameters **version** – string with new version

Returns None

set_version_using_archive (*archive_path*)

Method to update the version in the SPEC file using a archive path. The version is extracted from the archive name.

Parameters **archive_path** –

Returns

static split_version_string (*version_string=""*)

Method splits version string into version and possibly extra string as 'rc1' or 'b1', ...

Parameters **version_string** – version string such as '1.1.1' or '1.2.3b1', ...

Returns tuple of strings with (extracted version, extra version, separator) or (None, None, None) if extraction failed

update_changelog (**args, **kwargs*)

update_paths_to_patches (**args, **kwargs*)

update_setup_dirname (**args, **kwargs*)

write_updated_patches (**args, **kwargs*)

`rebasehelper.specfile.get_rebase_name` (*dir_name, name*)

Function returns a name in results directory

Parameters

- **dir_name** –
- **name** –

Returns full path to results dir with name

`rebasehelper.specfile.saves` (*func*)

Decorator for saving the SpecFile after a method is run.

5.2.10 Temporary environment module

class `rebasehelper.tmp_environment.TemporaryEnvironment` (*exit_callback=None*)

Class representing a temporary environment (directory) that can be used as a workspace.

Works as a context manager.

TEMPDIR = 'TEMPDIR'

env ()

Gets a copy of `_env` dictionary.

Returns Copy of `_env` dictionary.

Return type dict

path ()

Gets the path to the temporary environment.

Returns Absolute path to the temporary environment.

Return type str

5.3 Change Log

5.3.1 [Unreleased]

5.3.2 [0.18.0] - 2019-08-21

Added

- Added workaround for missing *mock* group in Fedora Rawhide

Fixed

- Moved setup dependencies from `install_requires` to `setup_requires`

Changed

- Refactored logging, see [logging documentation](#) for details
- Reason of build failure is now always logged

Removed

- Removed unused `Application` methods and attributes
- Removed unused testing files

5.3.3 [0.17.2] - 2019-08-09

Added

- Added tests for `SpecContent` class
- Enabled and configured [Packit-as-a-Service](#)

Fixed

- Added exception handling to PyPI release webhook endpoint
- Fixed `TestCLI.test_cli_unit()` test
- Updated `MANIFEST.in` to include all necessary files

Changed

- Improved tests for `Application` class
- `SpecFile.update_changelog()` now creates `%changelog` section if it doesn't exist
- **rebase-helper** now uses `setuptools-scm` to determine version from git
- `setup.py sdist` now supports overriding distribution base name with `--base-name` option

5.3.4 [0.17.1] - 2019-08-01

Fixed

- Fixed PyPI release webhook endpoint

Changed

- Removed direct dependencies preventing PyPI release

5.3.5 [0.17.0] - 2019-07-31

Added

- Added possibility for plugins to specify their own arguments
- Added basic type hints and enabled mypy linter
- Added `rust` package category
- **replace-old-version** SPEC hook can now replace old version with `%{version}` macro
- **replace-old-version** SPEC hook can now replace also parts of version in Sources and Patches

Fixed

- Fixed broken `ansicolors` dependency
- Fixed printing of output of unavailable checkers
- Fixed determining unmatched quotation in `%prep` section
- Made **files** build log hook handle conditions and macros in `%files` section
- **files** build log hook now adds man pages in a way that follows [Fedora Packaging Guidelines](#)
- **files** build log hook now ignores `debuginfo` files
- Fixed issues with polluted global macro namespace when parsing multiple different SPEC files
- Fixed handling of remote downstream patches (URLs)
- Fixed parsing of patch strip options in `%prep`
- Fixed wrapping of extra long lines in the usage documentation
- Sources from lookaside cache are now downloaded to a proper location

Changed

- Restructured the code layout of plugins
- Refactored parts of SpecFile class
- Disabled removing of “unused” patches
- `mock` is now automatically run with superuser privileges if necessary
- Local builder is now used if `--get-old-build-from-koji` is specified and the build can't be downloaded from Koji
- checkers (even the default ones) are now skipped if they are not available
- Sources are now copied to destination if they cannot be extracted
- Reimplemented downloading Koji builds
- Updated documentation

Removed

- Removed Python 2 support

5.3.6 [0.16.3] - 2019-05-03

Fixed

- Fixed handling of SPEC files with conditionalized sections
- Fixed **replace-old-version** SPEC hook not to update version in *%changelog* and local sources
- Fixed capturing RPM error output during SPEC parsing
- Fixed handling of absolute *%license* and *%doc* paths in **files** build log hook
- Fixed logging of SRPM and RPM build errors

Changed

- Updated packit configuration for packit 0.2.0
- Adapted to upcoming change in RPM python API
- Made `SpecFile` class more suitable for external use

5.3.7 [0.16.2] - 2019-03-07

Added

- Added support for `packit`
- Added SPEC hook for replacing old version string

Fixed

- Fixed documentation building by mocking `requests-gssapi`
- Fixed `TestOutputTool` for checkers

Changed

- Build log hooks are now run only if build of **new** binary packages fails
- It is now possible to use `--get-old-build-from-koji` option without FAS

5.3.8 [0.16.1] - 2019-02-28

Fixed

- Made `GitPatchTool` auto-skip empty commits caused by new rebase implementation in **git 2.20**
- Fixed `TestGitHelper` to work on real systems with existing git configuration

5.3.9 [0.16.0] - 2019-02-27

Added

- Added category for *R* packages
- Added `make test-podman` as an alternative to `make test-docker`
- Added `--skip-upload` option (to be used in conjunction with `--update-sources`)
- Added check that all sources for the new version are present
- Added SPEC hook for escaping macros in comments

Changed

- `--get-old-build-from-koji` now tries to get specific version build (as opposed to the latest one)
- Implemented parsing of multiline macros and shell expansions in SPEC files
- **rebase-helper** can now handle multiline enquoted strings in *%prep* section
- Refactored `GitPatchTool` to make the rebase process more robust and to preserve as much of the original downstream patches as possible
- `git mergetool` is now run again if there are some unresolved conflicts left
- Associated comments are now removed along with patches

Fixed

- Fixed populating list of logs on build failures
- Added missing abort after failed `git am`
- Fixed processing SPEC files without *%prep* section

- Fixed several issues in **ruby-helper** SPEC hook
- Fixed unwanted expansion of `%autosetup` macro
- Fixed automatic rebuilding based on build log hooks result
- Fixed removal of `%doc` and `%license` files in subpackages

Removed

- Removed `requests-kerberos` support and switched to `requests-gssapi` exclusively

5.3.10 [0.15.0] - 2018-12-21

Added

- Implemented build log hooks and added **files** hook to detect and fix missing/unpackaged files

Changed

- Refactored and simplified all plugins

Fixed

- Fixed not listing all argument choices while generating documentation
- Fixed error in parsing `rpmdiff` output
- Fixed insertion of extra blank lines to a SPEC file after removing patches

Removed

- Removed unneeded packages from base Docker image

5.3.11 [0.14.0] - 2018-10-04

Added

- Added **PathsToRPMMacros** SPEC hook for transforming paths in `%files` section
- Added `--favor-on-conflict` option to prefer upstream or downstream changes with conflicting patches

Changed

- Extended **PyPIURLFix** SPEC hook to incorporate the new <https://pypi.org> website
- Made processing of patches in a SPEC file more robust
- Rewritten functional test to use an artificial package designed to check most aspects of the rebase process
- `pylint` is now run with Python 3 only, as Python 2 variant is no longer supported
- Code refactoring, simplified `SpecFile` class

- Checkers are no longer required for **rebase-helper** to run, only available checkers are used

Fixed

- Fixed bug in **licensecheck** checker when used with **json** output tool
- Fixed SPEC hook tests
- Fixed strangely acting lookaside cache upload progressbar
- Fixed downloading of SRPMs with `--get-old-build-from-koji`
- Fixed building usage documentation

5.3.12 [0.13.2] - 2018-05-18

Added

- Added **licensecheck** checker for detecting license changes
- Added another *not-so-verbose* verbosity level

Changed

- Refactored `utils` module

Fixed

- Fixed **abipkgdiff** detecting changes in only one object file
- Fixed uploads to lookaside cache
- Fixed broken consequent build retries

5.3.13 [0.13.1] - 2018-04-19

Added

- Added `--apply-changes` option to apply *changes.patch* after successful rebase
- Implemented *.gitignore* update with `--update-sources`

Changed

- Extended `README.md`
- Cleaned up constants

Fixed

- Fixed crash after failed rebase when no checkers were run

5.3.14 [0.13.0] - 2018-03-29

Added

- Added possibility to make changes to specfile between build retries
- Added **CommitHashUpdater** SPEC hook
- Added **hackage** versioneer
- Added support for uncompressed tar archives
- Created integration environment for test suite to isolate it from the internet
- Added `--update-sources` option to update *sources* file and upload new sources to lookaside cache

Changed

- Switched to `requests` library for downloads
- Made error messages from **Koji** builds more useful
- Reworked handling of downstream patches
- Changed package build process to build first SRPMs and then RPMs
- Divided checkers into categories running at different phases of rebase
- **Koji** build tool refactored to be better adjustable and extensible
- Colorized **rebase-helper** output and enhanced log messages
- Significantly improved rebase summary and report

Fixed

- Fixed `TestConsoleHelper.test_get_message()` test
- Fixed bug in **rpmdiff** output analysis
- Fixed some code styling errors and a large number of issues found by static analysis

5.3.15 [0.12.0] - 2017-12-19

Added

- Added **npmjs** and **cpan** versioneers
- Added possibility to specify custom `py.test` arguments
- Added possibility to customize changelog entry
- Added version check to abort rebase if requested version is not newer than current
- Added separate tox tasks for linting
- Implemented **rpmbuild** and **mock** SRPM build tools
- Added possibility to configure rebase-helper with configuration file
- Added possibility to blacklist certain SPEC hooks or versioneers

- Created `rebasehelper/rebase-helper` Docker Hub repository

Changed

- Made several speed optimizations in the test suite
- Tests requiring superuser privileges are now automatically skipped if necessary
- Simplified build analysis and made related log messages more useful

Fixed

- Fixed documentation builds on `readthedocs.org` broken by `rpm distribution` requirement
- Fixed reading username and e-mail from git configuration
- Added missing dependencies to Dockerfile
- Fixed processing of custom builder options
- Added workarounds for RPM bugs related to `%sources` and `%patches`
- Fixed several unhandled exceptions
- Fixed parsing tarball filename containing certain characters

5.3.16 [0.11.0] - 2017-10-04

Added

- Added `rpm-py-installer` to install `rpm-python` from pip
- Implemented detection of package category (`python`, `perl`, `ruby`, `nodejs`, `php`)
- Added **RubyGems** versioneer
- Added **RubyHelper** SPEC hook for getting additional sources based on instructions in SPEC file comments

Changed

- Value of `Version` and `Release` tags is now preserved if there are any macros that can be modified instead
- Versioneers and SPEC hooks are now run only for matching package categories
- Bash completion is now generated from source code, so it is always up-to-date

Fixed

- Prevented unwanted modifications of `%prep` section
- Fixed unexpected removal of rpms and build logs after last build retry
- Added files are no longer listed as removed in **rpmdiff** report

5.3.17 [0.10.1] - 2017-08-30

Added

- Added `--version` argument

Changed

- **Anitya** versioneer now primarily searches for projects using Fedora mapping
- Python dependencies moved from `requirements.txt` to `setup.py`

Fixed

- Made `CustomManPagesBuilder` work with Sphinx \geq 1.6
- `%prep` section parser is now able to handle backslash-split lines

5.3.18 [0.10.0] - 2017-08-25

Added

- Implemented extensible SPEC hooks and versioners
- Added **PyPI** SPEC hook for automatic fixing of Source URL of Python packages
- Added **Anitya** and **PyPI** versioners for determining latest upstream version of a package
- Added possibility to download old version build of a package from Koji
- Added support for test suite to be run in Docker containers
- Implemented functional tests for automatic testing of whole rebase process
- Diff against original source files is now generated as *changes.patch*

Changed

- Introduced plugin system for extending build tools, checkers and output tools
- Updated for **Koji 1.13** which finally brings Python 3 support
- Improved output information and reports
- Added colorized output
- Improved project documentation

Fixed

- Pre-configured git username and e-mail address is now used if available
- Fixed several issues in **rpmdiff** and especially **abipkgdiff** checkers
- Fixed several test suite related issues

5.3.19 [0.9.0] - 2017-01-05

Added

- Old sources are now downloaded from Fedora lookaside cache
- Auto-generated and improved CLI documentation and man page
- Added support for downloading files of unknown size

Changed

- `SpecFile` class preparation for pre-download hooks
- Code cleanup and refactorization

Fixed

- Fixed regexp for getting release number from SPEC
- Fixed functionality of `--results-dir` option
- Several upstream monitoring fixes
- Fixed issues caused by Fedora Flag Day

5.3.20 [0.8.0] - 2016-07-31

Added

- Added support for JSON output format
- Added support for **copr** build tool
- Added support for passing arbitrary extra arguments to local builders (**mock**, **rpmbuild**) with `--builder-options`.
- Added new option `--build-retries` allows the user to specify number of build retries (by default 2)
- Added support for **csmock** check tool

Changed

- Renamed **fedpkg** build tool to **koji** to make it more clear
- Downloading of files is now done only using standard Python library and not using PyCURL

Fixed

- Many bug fixes and code clean up

5.3.21 [0.7.3] - 2016-04-08

Added

- Added `rpm.addMacro`

Fixed

- Handled exceptions raised during parsing of SPEC files
- Fixed unapplied patches mixing with deleted ones

5.3.22 [0.7.2] - 2016-03-15

Added

- Added information about scratch builds

Fixed

- Added check if file exists and is empty for **the-new-hotness**
- Patches are applied in case `--builds-nowait` option is used

5.3.23 [0.7.1] - 2016-02-22

Added

- Two new command line options used by upstream monitoring

Fixed

- **fedpkg** reimplementation

5.3.24 [0.7.0] - 2016-01-13

Changed

- Several improvements

Fixed

- **pkgdiff** is now smarter
- Included `tar.bz2` into list of supported formats
- Added support for noarch package in case of **fedpkg** build
- Checker should return `None` if there is no debug package

Removed

- Removed a bunch of debug stuff

5.3.25 [0.6.2] - 2015-11-09

Fixed

- Logs are being saved to their own directory
- Prep script is moved into workspace directory
- No more traceback in case `koji` module is not present
- Each checker creates its own log file
- **rebase-helper** informs if it failed or not
- Report on script is smarter

5.3.26 [0.6.1] - 2015-10-30

Added

- `upstream-monitoring.py` - used by upstream monitoring service
- `rebase-helper-fedmsg.py` - testing Python script

5.3.27 [0.6.0] - 2015-07-31

Added

- Parts of `%prep` section related to patching are executed
- Support for **abipkgdiff**

Fixed

- Several fixes
- Replaced `yum` with `dnf`

5.3.28 [0.5.0] - 2015-05-22

Added

- Added support for building packages via **fedpkg** (or **koji**)
- Added summary report for better overview
- `continue` option implemented for `git rebase`
- Added several tests
- Added class for operating with Git repositories

Changed

- `git rebase` is used instead of `patch` command

Fixed

- Fixed several decoding issues
- Several **PEP8** and **W1202** fixes

Removed

- `DiffHelper` class is not needed

5.3.29 [0.4.0] - 2014-12-05

Added

- Handling of extra versions like `b1`, `rc1`, etc.
- Added build log analyzer to detect unpackaged files or other issues
- Added Bash completion

Changed

- Improved version extraction from archive name
- **rebase-helper** output is logged to `rebase-helper-results` directory
- `SpecFile` class rewritten

5.3.30 [0.3.1] - 2014-07-25

Added

- New build class
- `--build-only` option
- Installation of build dependencies in case of **rpmbuild** tool
- More tests
- `RebaseHelperError` class for catching exceptions

Fixed

- Several fixes

5.3.31 [0.3.0]

Added

- `pkgdiff` tool for comparing RPM packages
- Tests for `Archive` class and SPEC file

5.3.32 [0.2.0]

Added

- `diff_helper` for comparing two tarballs
- Applying patches to tarballs
- `patch_helper`

5.3.33 [0.1.0]

Added

- Initial classes
- CLI interface

r

rebasehelper.application, 33
rebasehelper.archive, 34
rebasehelper.cli, 36
rebasehelper.exceptions, 36
rebasehelper.helpers.console_helper, 17
rebasehelper.helpers.copr_helper, 18
rebasehelper.helpers.download_helper,
19
rebasehelper.helpers.git_helper, 19
rebasehelper.helpers.input_helper, 20
rebasehelper.helpers.koji_helper, 20
rebasehelper.helpers.lookaside_cache_helper,
22
rebasehelper.helpers.macro_helper, 22
rebasehelper.helpers.path_helper, 23
rebasehelper.helpers.process_helper, 24
rebasehelper.helpers.rpm_helper, 25
rebasehelper.logger, 37
rebasehelper.patcher, 38
rebasehelper.plugins.build_log_hooks,
26
rebasehelper.plugins.build_tools, 27
rebasehelper.plugins.build_tools.rpm,
31
rebasehelper.plugins.build_tools.srpm,
32
rebasehelper.plugins.checkers, 28
rebasehelper.plugins.output_tools, 29
rebasehelper.plugins.plugin_collection,
30
rebasehelper.plugins.plugin_manager, 30
rebasehelper.plugins.spec_hooks, 32
rebasehelper.plugins.versioners, 33
rebasehelper.specfile, 39
rebasehelper.temporary_environment, 42

Symbols

- apply-changes
 - rebase-helper command line option,
15
- background {dark,light,auto}
 - rebase-helper command line option,
14
- build-log-hook-blacklist {files}
 - rebase-helper command line option,
15
- build-tasks OLD_TASK,NEW_TASK
 - rebase-helper command line option,
15
- builder-options BUILDER_OPTIONS
 - rebase-helper command line option,
16
- builds-nowait
 - rebase-helper command line option,
15
- buildtool {copr,koji,mock,rpmbuild}
 - rebase-helper command line option,
15
- changelog-entry CHANGELOG_ENTRY
 - rebase-helper command line option,
16
- color {always,never,auto}
 - rebase-helper command line option,
14
- config-file CONFIG_FILE
 - rebase-helper command line option,
16
- disable-inapplicable-patches
 - rebase-helper command line option,
15
- favor-on-conflict
 - {downstream,upstream,off}
 - rebase-helper command line option,
15
- force-build-log-hooks
 - rebase-helper command line option,
15
- non-interactive
 - rebase-helper command line option,
15
- not-download-sources
 - rebase-helper command line option,
15
- outputtool {json,text}
 - rebase-helper command line option,
15
- pkgcomparetool {abipkgdiff,csmock,licensecheck,pk...}
 - rebase-helper command line option,
15
- replace-old-version-with-macro
 - rebase-helper command line option,
16
- results-dir RESULTS_DIR
 - rebase-helper command line option,
14
- skip-upload
 - rebase-helper command line option,
15
- skip-version-check
 - rebase-helper command line option,
15
- spec-hook-blacklist
 - {commit-hash-updater,escape-macros,paths-to-...}
 - rebase-helper command line option,
15
- srpm-builder-options
 - SRPM_BUILDER_OPTIONS
 - rebase-helper command line option,
16
- srpm-buildtool {mock,rpmbuild}
 - rebase-helper command line option,
15
- update-sources
 - rebase-helper command line option,
15

-version
 rebase-helper command line option,
 14

-versioneer {anitya, cpan, hackage, npmjs, pypi, rubygems, rpm} (rebase-helper.plugins.spec_hooks), 32
 rebase-helper command line option,
 15

-versioneer-blacklist
 {anitya, cpan, hackage, npmjs, pypi, rubygems, rpm} (rebasehelper.helpers.copr_helper.CoprHelper class method), 18
 rebase-helper command line option,
 15

-workspace-dir WORKSPACE_DIR
 rebase-helper command line option,
 15

-h, -help
 rebase-helper command line option,
 14

-v, -verbose
 rebase-helper command line option,
 14

-w, -keep-workspace
 rebase-helper command line option,
 15

A

ACCEPTS_OPTIONS (rebase-helper.plugins.build_tools.rpm.BuildToolBase attribute), 31

add_file_handler() (rebase-helper.logger.LoggerHelper static method), 37

add_stream_handler() (rebase-helper.logger.LoggerHelper static method), 37

all_packages_installed() (rebase-helper.helpers.rpm_helper.RpmHelper static method), 25

Application (class in rebasehelper.application), 33

apply_changes() (rebase-helper.application.Application method), 33

apply_old_patches() (rebase-helper.patcher.Patcher class method), 38

apply_patch() (rebasehelper.patcher.Patcher class method), 38

ARCHES (rebasehelper.helpers.rpm_helper.RpmHelper attribute), 25

Archive (class in rebasehelper.archive), 34

ArchiveTypeBase (class in rebasehelper.archive), 34

B

BaseBuildLogHook (class in rebase-helper.plugins.build_log_hooks), 26

BaseChecker (class in rebasehelper.plugins.checkers), 28

BaseOutputTool (class in rebase-helper.plugins.output_tools), 29

BaseSpecHook (class in rebase-helper.plugins.spec_hooks), 32

BaseVersioneer (class in rebase-helper.plugins.versioners), 33

BinaryPackageBuildError, 36

build() (rebasehelper.helpers.copr_helper.CoprHelper class method), 18

build() (rebasehelper.plugins.build_tools.rpm.BuildToolBase class method), 31

build() (rebasehelper.plugins.build_tools.srpm.SRPMBuildToolBase class method), 32

build_binary_packages() (rebase-helper.application.Application method), 33

build_parser() (rebasehelper.cli.CLI static method), 36

build_source_packages() (rebase-helper.application.Application method), 34

BuildLogHookCollection (class in rebase-helper.plugins.build_log_hooks), 27

BuildTemporaryEnvironment (class in rebase-helper.plugins.build_tools), 27

BuildToolBase (class in rebase-helper.plugins.build_tools.rpm), 31

BuildToolCollection (class in rebase-helper.plugins.build_tools.rpm), 32

Bz2ArchiveType (class in rebasehelper.archive), 35

C

CATEGORIES (rebase-helper.plugins.build_log_hooks.BaseBuildLogHook attribute), 26

CATEGORIES (rebase-helper.plugins.spec_hooks.BaseSpecHook attribute), 32

CATEGORIES (rebase-helper.plugins.versioners.BaseVersioneer attribute), 33

CATEGORY (rebasehelper.plugins.checkers.BaseChecker attribute), 28

check_mock_privileges() (in module rebase-helper.plugins.build_tools), 28

CheckerCategory (class in rebase-helper.plugins.checkers), 29

CheckerCollection (class in rebase-helper.plugins.checkers), 29

CheckerNotFoundError, 36

CLI (class in rebasehelper.cli), 36

CliHelper (class in rebasehelper.cli), 36

COLLECTIONS (rebase-helper.plugins.plugin_manager.PluginManager attribute), 36

- attribute), 30
- color_is_light() (rebase-helper.helpers.console_helper.ConsoleHelper static method), 17
- ColorizingStreamHandler (class in rebase-helper.logger), 37
- colors (rebasehelper.logger.ColorizingStreamHandler attribute), 37
- ConsoleHelper (class in rebase-helper.helpers.console_helper), 17
- ConsoleHelper.Capturer (class in rebase-helper.helpers.console_helper), 17
- CoprHelper (class in rebase-helper.helpers.copr_helper), 18
- copy() (rebasehelper.specfile.SpecFile method), 40
- cprint() (rebasehelper.helpers.console_helper.ConsoleHelper class method), 17
- create_file_handlers() (rebase-helper.logger.LoggerHelper class method), 38
- create_project() (rebase-helper.helpers.copr_helper.CoprHelper class method), 18
- create_session() (rebase-helper.helpers.koji_helper.KojiHelper class method), 20
- create_stream_handlers() (rebase-helper.logger.LoggerHelper class method), 38
- CREATES_TASKS (rebase-helper.plugins.build_tools.rpm.BuildToolBase attribute), 31
- CustomLogger (class in rebasehelper.logger), 37
- ## D
- decorate_patch_name() (rebase-helper.patcher.Patcher static method), 38
- DEFAULT (rebasehelper.plugins.build_tools.rpm.BuildToolBase attribute), 31
- DEFAULT (rebasehelper.plugins.build_tools.srpm.SRPMBuildToolBase attribute), 32
- DEFAULT (rebasehelper.plugins.checkers.BaseChecker attribute), 28
- DEFAULT (rebasehelper.plugins.output_tools.BaseOutputTool attribute), 29
- detect_background() (rebase-helper.helpers.console_helper.ConsoleHelper static method), 18
- DEV_NULL (rebasehelper.helpers.process_helper.ProcessHelper attribute), 24
- DISALLOW_INLINE_COMMENTS (rebase-helper.specfile.SpecContent attribute), 39
- display_task_results() (rebase-helper.helpers.koji_helper.KojiHelper class method), 20
- download() (rebase-helper.helpers.lookaside_cache_helper.LookasideCacheHelper class method), 22
- download_build() (rebase-helper.helpers.copr_helper.CoprHelper class method), 18
- download_build() (rebase-helper.helpers.koji_helper.KojiHelper class method), 20
- download_file() (rebase-helper.helpers.download_helper.DownloadHelper static method), 19
- download_remote_sources() (rebase-helper.specfile.SpecFile method), 40
- download_task_results() (rebase-helper.helpers.koji_helper.KojiHelper class method), 20
- DownloadError, 36
- DownloadHelper (class in rebase-helper.helpers.download_helper), 19
- dump() (rebasehelper.helpers.macro_helper.MacroHelper static method), 22
- ## E
- emit() (rebasehelper.logger.ColorizingStreamHandler method), 37
- env() (rebasehelper.temporary_environment.TemporaryEnvironment method), 42
- environment variable
- PACKAGE, 13
 - REPOSITORY, 13
- exchange_control_sequence() (rebase-helper.helpers.console_helper.ConsoleHelper static method), 18
- expand() (rebasehelper.helpers.macro_helper.MacroHelper static method), 22
- expand_macros() (rebase-helper.helpers.macro_helper.MacroHelper class method), 22
- EXTENSION (rebasehelper.archive.ArchiveTypeBase attribute), 35
- EXTENSION (rebasehelper.archive.Bz2ArchiveType attribute), 35
- EXTENSION (rebasehelper.archive.TarArchiveType attribute), 35
- EXTENSION (rebasehelper.archive.TarBz2ArchiveType attribute), 35
- EXTENSION (rebasehelper.archive.TarGzArchiveType attribute), 35
- EXTENSION (rebasehelper.archive.TarXzArchiveType attribute), 35
- EXTENSION (rebasehelper.archive.TgzArchiveType attribute), 36

- EXTENSION (*rebasehelper.archive.ZipArchiveType attribute*), 36
- EXTENSION (*rebasehelper.plugins.output_tools.BaseOutputTool attribute*), 29
- extract() (*rebasehelper.archive.ArchiveTypeBase class method*), 35
- extract() (*rebasehelper.archive.Bz2ArchiveType class method*), 35
- extract() (*rebasehelper.archive.TarGzArchiveType class method*), 35
- extract() (*rebasehelper.archive.TarXzArchiveType class method*), 35
- extract() (*rebasehelper.archive.ZipArchiveType class method*), 36
- extract_archive() (*rebasehelper.application.Application static method*), 34
- extract_archive() (*rebasehelper.archive.Archive method*), 34
- extract_patch_name() (*rebasehelper.patcher.Patcher class method*), 38
- extract_sources() (*rebasehelper.application.Application static method*), 34
- extract_version_from_archive_name() (*rebasehelper.specfile.SpecFile static method*), 40
- ## F
- file_available() (*rebasehelper.helpers.path_helper.PathHelper static method*), 23
- filter() (*rebasehelper.helpers.macro_helper.MacroHelper static method*), 22
- find_all_files() (*rebasehelper.helpers.path_helper.PathHelper static method*), 23
- find_all_files_current_dir() (*rebasehelper.helpers.path_helper.PathHelper static method*), 23
- find_archive_target_in_prep() (*rebasehelper.specfile.SpecFile method*), 40
- find_first_dir_with_file() (*rebasehelper.helpers.path_helper.PathHelper static method*), 23
- find_first_file() (*rebasehelper.helpers.path_helper.PathHelper static method*), 24
- format() (*rebasehelper.plugins.build_log_hooks.BaseBuildLogHook class method*), 26
- format() (*rebasehelper.plugins.checkers.BaseChecker class method*), 28
- functional (*rebasehelper.helpers.koji_helper.KojiHelper attribute*), 21
- ## G
- generate_patch() (*rebasehelper.application.Application method*), 34
- get_all_plugins() (*rebasehelper.plugins.plugin_collection.PluginCollection method*), 30
- get_applied_patches() (*rebasehelper.specfile.SpecFile method*), 40
- get_arches() (*rebasehelper.helpers.rpm_helper.RpmHelper static method*), 26
- get_archive() (*rebasehelper.specfile.SpecFile method*), 40
- get_build() (*rebasehelper.helpers.koji_helper.KojiHelper class method*), 21
- get_build_status() (*rebasehelper.helpers.copr_helper.CoprHelper class method*), 18
- get_build_url() (*rebasehelper.helpers.copr_helper.CoprHelper class method*), 19
- get_builder_options() (*rebasehelper.plugins.build_tools.rpm.BuildToolBase static method*), 31
- get_category() (*rebasehelper.plugins.checkers.BaseChecker class method*), 28
- get_checker_output_dir_short() (*rebasehelper.plugins.checkers.BaseChecker class method*), 28
- get_client() (*rebasehelper.helpers.copr_helper.CoprHelper class method*), 19
- get_comment_span() (*rebasehelper.specfile.SpecContent class method*), 39
- get_default_plugins() (*rebasehelper.plugins.checkers.CheckerCollection method*), 29
- get_default_plugins() (*rebasehelper.plugins.plugin_collection.PluginCollection method*), 30
- get_detached_task() (*rebasehelper.plugins.build_tools.rpm.BuildToolBase class method*), 31
- get_log_hook() (*rebasehelper.helpers.git_helper.GitHelper class method*), 19
- get_extra_version() (*rebasehelper.specfile.SpecFile method*), 40
- get_header_from_rpm() (*rebasehelper.helpers.rpm_helper.RpmHelper static method*), 26

- method), 26
- get_important_changes() (rebase-helper.plugins.checkers.BaseChecker class method), 28
- get_latest_build() (rebase-helper.helpers.koji_helper.KojiHelper class method), 21
- get_logs() (rebase-helper.plugins.build_tools.rpm.BuildToolBase class method), 31
- get_logs() (rebase-helper.plugins.build_tools.srpm.SRPMBuildToolBase class method), 33
- get_main_files_section() (rebase-helper.specfile.SpecFile method), 40
- get_message() (rebase-helper.helpers.input_helper.InputHelper static method), 20
- get_new_build_logs() (rebase-helper.application.Application method), 34
- get_new_log() (rebasehelper.specfile.SpecFile method), 40
- get_not_used_patches() (rebase-helper.specfile.SpecFile method), 41
- get_NVR() (rebasehelper.specfile.SpecFile method), 40
- get_old_build_info() (rebase-helper.helpers.koji_helper.KojiHelper class method), 21
- get_options() (rebase-helper.plugins.plugin_collection.PluginCollection method), 30
- get_options() (rebase-helper.plugins.plugin_manager.PluginManager method), 30
- get_patch_name() (rebase-helper.specfile.PatchObject method), 39
- get_patches() (rebasehelper.specfile.SpecFile method), 41
- get_plugin() (rebase-helper.plugins.plugin_collection.PluginCollection method), 30
- get_prep_section() (rebase-helper.specfile.SpecFile method), 41
- get_rebase_name() (in module rebase-helper.specfile), 42
- get_release_number() (rebase-helper.specfile.SpecFile method), 41
- get_report_path() (rebase-helper.plugins.output_tools.BaseOutputTool class method), 29
- get_rpm_spec() (rebase-helper.helpers.rpm_helper.RpmHelper class method), 26
- get_setup_dirname() (rebase-helper.specfile.SpecFile method), 41
- get_sources() (rebasehelper.specfile.SpecFile method), 41
- get_srpm_builder_options() (rebase-helper.plugins.build_tools.srpm.SRPMBuildToolBase static method), 33
- get_subpackage_name() (rebase-helper.specfile.SpecFile static method), 41
- get_supported_archives() (rebase-helper.archive.Archive class method), 34
- get_supported_plugins() (rebase-helper.plugins.checkers.CheckerCollection method), 29
- get_supported_plugins() (rebase-helper.plugins.plugin_collection.PluginCollection method), 30
- get_task_info() (rebase-helper.plugins.build_tools.rpm.BuildToolBase class method), 31
- get_task_url() (rebase-helper.helpers.koji_helper.KojiHelper class method), 21
- get_temp_dir() (rebase-helper.helpers.path_helper.PathHelper static method), 24
- get_underlined_title() (rebase-helper.plugins.checkers.BaseChecker class method), 28
- get_user() (rebase-helper.helpers.git_helper.GitHelper class method), 19
- get_version() (rebasehelper.specfile.SpecFile method), 41
- GIT_USER_EMAIL (rebase-helper.helpers.git_helper.GitHelper attribute), 19
- GIT_USER_NAME (rebase-helper.helpers.git_helper.GitHelper attribute), 19
- GitHelper (class in rebasehelper.helpers.git_helper), 19
- ## H
- haskell (rebasehelper.specfile.PackageCategory attribute), 39
- HEADING (rebasehelper.logger.CustomLogger attribute), 37
- ## I
- IMPORTANT (rebasehelper.logger.CustomLogger attribute), 37
- init_git() (rebasehelper.patcher.Patcher class method), 38

InputHelper (class in rebase-helper.helpers.input_helper), 20
 insert_patch_name() (rebase-helper.patcher.Patcher class method), 38
 install_build_dependencies() (rebase-helper.helpers.rpm_helper.RpmHelper static method), 26
 is_available() (rebase-helper.plugins.checkers.BaseChecker class method), 28
 is_package_installed() (rebase-helper.helpers.rpm_helper.RpmHelper static method), 26
 is_test_suite_enabled() (rebase-helper.specfile.SpecFile method), 41

K

KojiHelper (class in rebase-helper.helpers.koji_helper), 20

L

LoggerHelper (class in rebasehelper.logger), 37
 LookasideCacheError, 36
 LookasideCacheHelper (class in rebase-helper.helpers.lookaside_cache_helper), 22

M

MacroHelper (class in rebase-helper.helpers.macro_helper), 22
 MACROS_WHITELIST (rebase-helper.helpers.macro_helper.MacroHelper attribute), 22
 match() (rebasehelper.archive.ArchiveTypeBase class method), 35
 match() (rebasehelper.archive.ZipArchiveType class method), 36
 merge_two_results() (rebase-helper.plugins.build_log_hooks.BaseBuildLogHook class method), 26
 MockTemporaryEnvironment (class in rebase-helper.plugins.build_tools), 27

N

new_repo (rebasehelper.patcher.Patcher attribute), 38
 new_sources (rebasehelper.patcher.Patcher attribute), 38
 nodejs (rebasehelper.specfile.PackageCategory attribute), 39
 non_interactive (rebasehelper.patcher.Patcher attribute), 38

O

old_repo (rebasehelper.patcher.Patcher attribute), 38

old_sources (rebasehelper.patcher.Patcher attribute), 38
 open() (rebasehelper.archive.ArchiveTypeBase class method), 35
 open() (rebasehelper.archive.Bz2ArchiveType class method), 35
 open() (rebasehelper.archive.TarGzArchiveType class method), 35
 open() (rebasehelper.archive.TarXzArchiveType class method), 36
 open() (rebasehelper.archive.ZipArchiveType class method), 36
 output_data (rebasehelper.patcher.Patcher attribute), 38
 OutputToolCollection (class in rebase-helper.plugins.output_tools), 30

P

PACKAGE, 13
 PackageCategory (class in rebasehelper.specfile), 39
 parse_rgb_device_specification() (rebase-helper.helpers.console_helper.ConsoleHelper static method), 18
 parse_spec() (rebase-helper.helpers.rpm_helper.RpmHelper class method), 26
 ParseError, 36
 patch() (rebasehelper.patcher.Patcher class method), 38
 patch_sources() (rebase-helper.application.Application method), 34
 Patcher (class in rebasehelper.patcher), 38
 patches (rebasehelper.patcher.Patcher attribute), 38
 PatchList (class in rebasehelper.specfile), 39
 PatchObject (class in rebasehelper.specfile), 39
 path() (rebasehelper.temporary_environment.TemporaryEnvironment method), 42
 PathHelper (class in rebase-helper.helpers.path_helper), 23
 perl (rebasehelper.specfile.PackageCategory attribute), 39
 php (rebasehelper.specfile.PackageCategory attribute), 39
 PluginCollection (class in rebase-helper.plugins.plugin_collection), 30
 PluginManager (class in rebase-helper.plugins.plugin_manager), 30
 prepare() (rebasehelper.plugins.build_tools.rpm.BuildToolBase class method), 31
 prepare_next_run() (rebase-helper.application.Application method), 34

prepare_sources() (*rebase-helper.application.Application* method), 34
 prepend_results_dir_name() (*rebase-helper.plugins.output_tools.BaseOutputTool* class method), 29
 print_cli_summary() (*rebase-helper.plugins.output_tools.BaseOutputTool* class method), 29
 print_important_checkers_output() (*rebase-helper.plugins.output_tools.BaseOutputTool* class method), 29
 print_patches_cli() (*rebase-helper.plugins.output_tools.BaseOutputTool* class method), 29
 print_patches_section_cli() (*rebase-helper.plugins.output_tools.BaseOutputTool* class method), 29
 print_report_file_path() (*rebase-helper.plugins.output_tools.BaseOutputTool* class method), 29
 print_summary() (*rebase-helper.application.Application* method), 34
 print_task_info() (*rebase-helper.application.Application* method), 34
 ProcessHelper (class in *rebase-helper.helpers.process_helper*), 24
 progress() (*rebase-helper.helpers.download_helper.DownloadHelper* static method), 19
 purge_macro() (*rebase-helper.helpers.macro_helper.MacroHelper* class method), 23
 python (*rebasehelper.specfile.PackageCategory* attribute), 39

R

R (*rebasehelper.specfile.PackageCategory* attribute), 39
 rebase-helper command line option
 -apply-changes, 15
 -background {dark, light, auto}, 14
 -build-log-hook-blacklist {files}, 15
 -build-tasks OLD_TASK, NEW_TASK, 15
 -builder-options BUILDER_OPTIONS, 16
 -builds-nowait, 15
 -buildtool {copr, koji, mock, rpmbuild}, 15
 -changelog-entry CHANGELOG_ENTRY, 16
 -color {always, never, auto}, 14
 -config-file CONFIG_FILE, 16
 -disable-inapplicable-patches, 15
 -favor-on-conflict {downstream, upstream, off}, 15
 -force-build-log-hooks, 15
 -non-interactive, 15
 -not-download-sources, 15
 -outputtool {json, text}, 15
 -pkgcomparetool {abipkgdiff, csmock, licensecheck, pkgdiff, rpm}, 15
 -replace-old-version-with-macro, 16
 -results-dir RESULTS_DIR, 14
 -skip-upload, 15
 -skip-version-check, 15
 -spec-hook-blacklist {commit-hash-updater, escape-macros, paths-to}, 15
 -srpm-builder-options SRPM_BUILDER_OPTIONS, 16
 -srpm-buildtool {mock, rpmbuild}, 15
 -update-sources, 15
 -version, 14
 -versioneer {anitya, cpan, hackage, npmjs, pypi, rubygems}, 15
 -versioneer-blacklist {anitya, cpan, hackage, npmjs, pypi, rubygems}, 15
 -workspace-dir WORKSPACE_DIR, 15
 -h, -help, 14
 -v, -verbose, 14
 -w, -keep-workspace, 15

rebasehelper.application (module), 33
 rebasehelper.archive (module), 34
 rebasehelper.cli (module), 36
 rebasehelper.exceptions (module), 36
 rebasehelper.helpers.console_helper (module), 17
 rebasehelper.helpers.copr_helper (module), 18
 rebasehelper.helpers.download_helper (module), 19
 rebasehelper.helpers.git_helper (module), 19
 rebasehelper.helpers.input_helper (module), 20
 rebasehelper.helpers.koji_helper (module), 20
 rebasehelper.helpers.lookaside_cache_helper (module), 22
 rebasehelper.helpers.macro_helper (module), 22
 rebasehelper.helpers.path_helper (module), 23
 rebasehelper.helpers.process_helper

(*module*), 24

rebasehelper.helpers.rpm_helper (*module*), 25

rebasehelper.logger (*module*), 37

rebasehelper.patcher (*module*), 38

rebasehelper.plugins.build_log_hooks (*module*), 26

rebasehelper.plugins.build_tools (*module*), 27

rebasehelper.plugins.build_tools.rpm (*module*), 31

rebasehelper.plugins.build_tools.srpm (*module*), 32

rebasehelper.plugins.checkers (*module*), 28

rebasehelper.plugins.output_tools (*module*), 29

rebasehelper.plugins.plugin_collection (*module*), 30

rebasehelper.plugins.plugin_manager (*module*), 30

rebasehelper.plugins.spec_hooks (*module*), 32

rebasehelper.plugins.versioners (*module*), 33

rebasehelper.specfile (*module*), 39

rebasehelper.temporary_environment (*module*), 42

RebaseHelperError, 36

redefine_release_with_macro() (*rebase-helper.specfile.SpecFile method*), 41

register_archive_type() (*in module rebase-helper.archive*), 36

reload() (*rebasehelper.specfile.SpecFile method*), 41

remove_file_handlers() (*rebase-helper.logger.LoggerHelper class method*), 38

replace_section() (*rebase-helper.specfile.SpecContent method*), 39

REPOSITORY, 13

request() (*rebasehelper.helpers.download_helper.DownloadHelper class static method*), 19

results_dir (*rebase-helper.plugins.checkers.BaseChecker attribute*), 28

revert_redefine_release_with_macro() (*rebasehelper.specfile.SpecFile method*), 41

rpkg_config_dir (*rebase-helper.helpers.lookaside_cache_helper.LookasideCacheHelper class attribute*), 22

RPM (*rebasehelper.plugins.checkers.CheckerCategory attribute*), 29

RpmbuildTemporaryEnvironment (*class in rebasehelper.plugins.build_tools*), 27

RpmHeader (*class in rebasehelper.helpers.rpm_helper*), 25

RpmHelper (*class in rebasehelper.helpers.rpm_helper*), 25

ruby (*rebasehelper.specfile.PackageCategory attribute*), 39

run() (*rebasehelper.application.Application method*), 34

run() (*rebasehelper.cli.CliHelper static method*), 36

run() (*rebasehelper.plugins.build_log_hooks.BaseBuildLogHook class method*), 26

run() (*rebasehelper.plugins.build_log_hooks.BuildLogHookCollection method*), 27

run() (*rebasehelper.plugins.checkers.CheckerCollection method*), 29

run() (*rebasehelper.plugins.output_tools.BaseOutputTool class method*), 30

run() (*rebasehelper.plugins.output_tools.OutputToolCollection method*), 30

run() (*rebasehelper.plugins.spec_hooks.BaseSpecHook class method*), 32

run() (*rebasehelper.plugins.spec_hooks.SpecHookCollection method*), 32

run() (*rebasehelper.plugins.versioners.BaseVersioneer class method*), 33

run() (*rebasehelper.plugins.versioners.VersioneerCollection method*), 33

run_check() (*rebase-helper.plugins.checkers.BaseChecker class method*), 28

run_mergetool() (*rebase-helper.helpers.git_helper.GitHelper class method*), 19

run_package_checkers() (*rebase-helper.application.Application method*), 34

run_subprocess() (*rebase-helper.helpers.process_helper.ProcessHelper static method*), 24

run_subprocess_cwd() (*rebase-helper.helpers.process_helper.ProcessHelper static method*), 24

run_subprocess_cwd_env() (*rebase-helper.helpers.process_helper.ProcessHelper static method*), 25

run_subprocess_env() (*rebase-helper.helpers.process_helper.ProcessHelper static method*), 25

CacheHelper (*rebasehelper.specfile.PackageCategory attribute*), 39

S

save() (*rebasehelper.specfile.SpecFile method*), 41

saves() (*in module rebasehelper.specfile*), 42

- section() (*rebasehelper.specfile.SpecContent* method), 39
- SECTION_HEADERS (*rebase-helper.specfile.SpecContent* attribute), 39
- set_extra_version() (*rebase-helper.specfile.SpecFile* method), 41
- set_release_number() (*rebase-helper.specfile.SpecFile* method), 41
- set_tag() (*rebasehelper.specfile.SpecFile* method), 41
- set_terminal_background() (*rebase-helper.logger.ColorizingStreamHandler* method), 37
- set_version() (*rebasehelper.specfile.SpecFile* method), 41
- set_version_using_archive() (*rebase-helper.specfile.SpecFile* method), 42
- setup() (*rebasehelper.application.Application* static method), 34
- should_use_colors() (*rebase-helper.helpers.console_helper.ConsoleHelper* class method), 18
- SOURCE (*rebasehelper.plugins.checkers.CheckerCategory* attribute), 29
- source_dir (*rebasehelper.patcher.Patcher* attribute), 38
- SourcePackageBuildError, 36
- SOURCES
rebase-helper command line option, 14
- SpecContent (*class in rebasehelper.specfile*), 39
- SpecFile (*class in rebasehelper.specfile*), 40
- SpecHookCollection (*class in rebase-helper.plugins.spec_hooks*), 32
- split_nevra() (*rebase-helper.helpers.rpm_helper.RpmHelper* class method), 26
- split_version_string() (*rebase-helper.specfile.SpecFile* static method), 42
- SRPM (*rebasehelper.plugins.checkers.CheckerCategory* attribute), 29
- SRPMBuildToolBase (*class in rebase-helper.plugins.build_tools.srpm*), 32
- SRPMBuildToolCollection (*class in rebase-helper.plugins.build_tools.srpm*), 33
- strip_patch_name() (*rebasehelper.patcher.Patcher* class method), 38
- substitute_path_with_macros() (*rebase-helper.helpers.macro_helper.MacroHelper* static method), 23
- SUCCESS (*rebasehelper.logger.CustomLogger* attribute), 37
- T**
- TarArchiveType (*class in rebasehelper.archive*), 35
- TarBz2ArchiveType (*class in rebasehelper.archive*), 35
- TarGzArchiveType (*class in rebasehelper.archive*), 35
- TarXzArchiveType (*class in rebasehelper.archive*), 35
- TEMPDIR (*rebasehelper.temporary_environment.TemporaryEnvironment* attribute), 42
- TEMPDIR_BUILD (*rebase-helper.plugins.build_tools.RpmbuildTemporaryEnvironment* attribute), 28
- TEMPDIR_BUILDROOT (*rebase-helper.plugins.build_tools.RpmbuildTemporaryEnvironment* attribute), 28
- TEMPDIR_RESULTS (*rebase-helper.plugins.build_tools.BuildTemporaryEnvironment* attribute), 27
- TEMPDIR_RPMBUILD (*rebase-helper.plugins.build_tools.RpmbuildTemporaryEnvironment* attribute), 28
- TEMPDIR_RPMS (*rebase-helper.plugins.build_tools.RpmbuildTemporaryEnvironment* attribute), 28
- TEMPDIR_SOURCES (*rebase-helper.plugins.build_tools.BuildTemporaryEnvironment* attribute), 27
- TEMPDIR_SPEC (*rebase-helper.plugins.build_tools.BuildTemporaryEnvironment* attribute), 27
- TEMPDIR_SPECS (*rebase-helper.plugins.build_tools.BuildTemporaryEnvironment* attribute), 27
- TEMPDIR_SRPMS (*rebase-helper.plugins.build_tools.RpmbuildTemporaryEnvironment* attribute), 28
- TemporaryEnvironment (*class in rebase-helper.temporary_environment*), 42
- terminal_background (*rebase-helper.logger.ColorizingStreamHandler* attribute), 37
- TgzArchiveType (*class in rebasehelper.archive*), 36
- TRACE (*rebasehelper.logger.CustomLogger* attribute), 37
- U**
- update_changelog() (*rebase-helper.specfile.SpecFile* method), 42
- update_paths_to_patches() (*rebase-helper.specfile.SpecFile* method), 42
- update_setup_dirname() (*rebase-helper.specfile.SpecFile* method), 42
- update_sources() (*rebase-helper.helpers.lookaside_cache_helper.LookasideCacheHelper* class method), 22

`upload_srpm()` (*rebase-helper.helpers.koji_helper.KojiHelper class method*), 21

`use_colors` (*rebase-helper.helpers.console_helper.ConsoleHelper attribute*), 18

V

`VERBOSE` (*rebasehelper.logger.CustomLogger attribute*), 37

`VersioneerCollection` (*class in rebase-helper.plugins.versioners*), 33

W

`wait_for_task()` (*rebase-helper.plugins.build_tools.rpm.BuildToolBase class method*), 31

`watch_build()` (*rebase-helper.helpers.copr_helper.CoprHelper class method*), 19

`watch_koji_tasks()` (*rebase-helper.helpers.koji_helper.KojiHelper class method*), 22

`write_updated_patches()` (*rebase-helper.specfile.SpecFile method*), 42

Z

`ZipArchiveType` (*class in rebasehelper.archive*), 36