
Rebase Helper Documentation

Release 0.11.0

Petr Hracek

Tomas Hozza

Dec 19, 2017

Contents

1	How to get it running?	3
2	How to use it?	5
3	How does it work?	7
3.1	User Guide	8
3.2	API	12
3.3	Change Log	32
	Python Module Index	39

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.

How does it work?

The following steps describe a rebase process:

- *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
- old and new source tarballs are downloaded and extracted to *rebase-helper-workspace* directory
- downstream patches are rebased on top of new sources using `git-rebase`, resulting modified patches are saved to *rebase-helper-results/rebased-sources* directory
- old and new source RPMs are created and built with selected build tool
- multiple checker tools are run against both sets of packages and their output is stored in *rebase-helper-results/checkers* directory
- *rebase-helper-workspace* directory is removed

Rebasing of downstream patches is done as follows:

- 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*

3.1 User Guide

3.1.1 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

3.1.2 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.

3.1.3 Usage

SYNOPSIS

```
rebase-helper [-h] [--version] [-v] [--color {always,never,auto}]
  [--results-dir RESULTS_DIR]
  [-p|-b|--comparepkgs-only COMPAREPKGS_DIR] [-c]
  [--buildtool {copr,rpmbuild,mock}] [--srpm-buildtool {rpmbuild,mock}]
  [--pkgcomparetool {rpmdiff,abipkgdiff,pkgdiff,csmock}]
  [--outputtool {text,json}]
  [--versioneer {cpan,pypi,rubygems,anitya,npmjs}]
  [--versioneer-blacklist {cpan,pypi,rubygems,anitya,npmjs}]
  [--spec-hook-blacklist {RubyHelperHook,PyPIURLFixHook,TypoFixHook}]
  [--non-interactive] [--not-download-sources] [-w]
  [--disable-inapplicable-patches] [--get-old-build-from-koji]
  [--skip-version-check] [--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]
```

[*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

--results-dir RESULTS_DIR

directory where rebase-helper output will be stored

-p, --patch-only

only apply patches

-b, --build-only

only build SRPMs and RPMs

--comparepkgs-only COMPAREPKGS_DIR

compare already built packages, COMPAREPKGS_DIR must be a directory with the following structure:
<dir_name>/{old,new}/RPM

-c, --continue

continue previously interrupted rebase

--buildtool {copr,rpmbuild,mock}

build tool to use, defaults to mock

--srpm-buildtool {rpmbuild,mock}

SRPM build tool to use, defaults to rpmbuild

--pkgcomparetool {rpmdiff,abipkgdiff,pkgdiff,csmock}

set of tools to use for package comparison, defaults to rpmdiff, abipkgdiff, pkgdiff

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

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

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

--spec-hook-blacklist {RubyHelperHook,PyPIURLFixHook,TypoFixHook}
prevent specified spec hooks from being run

--non-interactive
do not interact with user

--not-download-sources
do not download sources

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

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

--get-old-build-from-koji
do not build old sources, download latest build from Koji instead

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

--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

3.1.4 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 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.build_helper. BuildToolBase
checker	rebasehelper.checkers	rebasehelper.checker.BaseChecker
output tool	rebasehelper. output_tools	rebasehelper.output_tool. BaseOutputTool
SPEC hook	rebasehelper.spec_hooks	rebasehelper.specfile.BaseSpecHook
versioneer	rebasehelper.versioners	rebasehelper.versioner.BaseVersioner

Example

Listing 3.1: my_spec_hook/__init__.py

```

from rebasehelper.specfile import BaseSpecHook

class MySpecHook(BaseSpecHook):

    NAME = 'MySpecHook'

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

    @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.insert(0, '# processed by %s\n' % cls.NAME)
        rebase_spec_file.save()

```

Listing 3.2: 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']
    }
)
```

3.2 API

3.2.1 Application module

class rebasehelper.application.**Application** (*cli_conf*, *execution_dir*, *results_dir*, *debug_log_file*)

build_packages ()

Function calls build class for building packages

debug_log_file = None

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_all_log_files ()

Function returns all log_files created by rebase-helper First if debug log file and second is report summary log file

Returns

get_checker_outputs ()

get_new_build_logs ()

get_rebased_patches ()

Function returns a list of patches either ‘’: [list_of_deleted_patches] :return:

`get_rebasehelper_data()`

`get_rpm_packages` (*dirname*)

Function returns RPM packages stored in *dirname/old* and *dirname/new* directories

Parameters *dirname* – directory where are stored old and new RPMS

Returns

`kwargs = {}`

`new_rest_sources = []`

`new_sources = ''`

`old_rest_sources = []`

`old_sources = ''`

`patch_sources` (*sources*)

`prepare_sources()`

Function prepares a sources.

Returns

`print_summary` (*exception=None*)

Save rebase-helper result and print the summary using `output_tools_runner` :param *exception*: Error message from rebase-helper :return:

`print_task_info` (*builder*)

`rebase_spec_file = None`

`rebase_spec_file_path = None`

`rebased_patches = {}`

`rebased_repo = None`

`report_log_file = None`

`result_file = ''`

`run()`

`run_download_compare` (*tasks_dict, dir_name*)

`run_package_checkers` (*results_dir*)

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

Parameters *results_dir* (*str*) – Path to directory in which to store the results.

Returns None

`set_upstream_monitoring()`

`static setup` (*cli_conf*)

`spec_file = None`

`spec_file_path = None`

`temp_dir = ''`

`upstream_monitoring = False`

3.2.2 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 (filename=None, *args, **kwargs)
        Extracts the archive into the given path

        Parameters path – Path where to extract the archive to.

        Returns

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

    classmethod open (filename=None, *args, **kwargs)
        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, *args, **kwargs)

    classmethod open (filename=None)

class rebasehelper.archive.GemPseudoArchiveType
    .gem files are not archives - this is a pseudo type

    EXTENSION = '.gem'

    classmethod extract (archive=None, filename=None, path=None, *args, **kwargs)

    classmethod open (filename=None)

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, *args, **kwargs)

    classmethod open (filename=None)

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

```

EXTENSION = '.tar.xz'

classmethod extract (archive=None, filename=None, path=None, *args, **kwargs)

classmethod open (filename=None)

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, *args, **kwargs)

    classmethod match (filename=None)

    classmethod open (filename=None)

rebasehelper.archive.register_archive_type (archive)

```

3.2.3 Build helper module

```

exception rebasehelper.build_helper.BinaryPackageBuildError (*args, **kwargs)
    Error indicating failure to build Binary Package

class rebasehelper.build_helper.BuildTemporaryEnvironment (sources, patches, spec,
                                                         results_dir)
    Class representing temporary environment.

    TEMPDIR_RESULTS = 'TEMPDIR_RESULTS'
    TEMPDIR_SOURCES = 'TEMPDIR_SOURCES'
    TEMPDIR_SPEC = 'TEMPDIR_SPEC'
    TEMPDIR_SPECS = 'TEMPDIR_SPECS'

class rebasehelper.build_helper.BuildToolBase
    Base class for various build tools

    DEFAULT = False

    classmethod accepts_options ()
        Checks if the tool accepts additional command line options.

    classmethod build (*args, **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

    classmethod creates_tasks ()
        Checks if the tool creates build tasks.

    classmethod get_build_tool_name ()
        Returns the name of the build tool.

    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

static `get_srpm_builder_options` (**kwargs)

static `get_srpm_buildtool` (**kwargs)

classmethod `get_task_info` (*build_dict*)

Gets information about detached remote task

Parameters **build_dict** – build data

Returns task info

classmethod `is_default` ()

Checks if the tool is the default build tool.

classmethod `match` (*cmd=None, *args, **kwargs*)

Checks if tool name matches the desired one.

classmethod `prepare` (*spec, conf*)

Prepare for building.

Parameters **spec** – spec file object

classmethod `wait_for_task` (*build_dict, 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.build_helper.Builder` (*tool=None*)

Class representing a process of building binaries from sources.

accepts_options ()

build (**args, **kwargs*)

Build sources.

build_tools = {'copr': <class 'rebasehelper.build_tools.copr_tool.CoprBuildTool'>, 'rpm'

creates_tasks ()

classmethod `get_default_tool` ()

Returns default build tool

get_detached_task (*task_id, results_dir*)

Get detached task

get_logs ()

Get logs.

```

classmethod get_supported_tools ()
    Returns a list of supported build tools

    Returns list of supported build tools

get_task_info (build_dict)
    Get task info

classmethod load_build_tools ()

prepare (spec, conf)
    Prepare for build

wait_for_task (build_dict, results_dir)
    Wait for task

class rebasehelper.build_helper.RpmbuildTemporaryEnvironment (sources, patches,
                                                                spec, results_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'

class rebasehelper.build_helper.SRPMBuildToolBase
    Base class for SRPM builder tools

    DEFAULT = False

    classmethod build_srpm (spec, workdir, results_dir, srpm_builder_options)
    Build SRPM with chosen SRPM Build Tool

    Parameters

    • spec – abs path to SPEC file inside the rpmbuild/SPECS in workdir.

    • workdir – abs path to working directory with rpmbuild directory structure, which will
      be used as HOME dir.

    • results_dir – abs path to dir where the log should be placed.

    • srpm_builder_options – list of additional options to rpmbuild.

    Returns If build process ends successfully returns abs path to built SRPM, otherwise 'None'.

    classmethod get_build_tool_name ()
    Returns name of the SRPM Build Tool

    classmethod is_default ()
    Returns true if the SRPM Build Tool is set to be default

class rebasehelper.build_helper.SRPMBuilder
    Builder class for building SRPMs.

    classmethod get_default_tool ()
    Returns default build tool

    classmethod get_supported_tools ()
    Returns list of supported srpm build tools

    classmethod load_srpm_build_tools ()

```

```
srpm_build_tools = {'rpmbuild': <class 'rebasehelper.srpm_build_tools.rpmbuild_tool.RpmbuildTool'>
```

```
exception rebasehelper.build_helper.SourcePackageBuildError (*args, **kwargs)
    Error indicating failure to build Source Package.
```

3.2.4 CLI module

```
class rebasehelper.cli.CLI (args=None)
    Class for processing data from commandline
```

```
    static build_parser ()
```

```
class rebasehelper.cli.CliHelper
```

```
    static run ()
```

```
class rebasehelper.cli.CustomAction (option_strings, switch=False, actual_default=None,
    dest=None, default=None, nargs=None, required=False,
    type=None, metavar=None, help=None, choices=None)
```

```
class rebasehelper.cli.CustomArgumentParser (prog=None, usage=None, description=None,
    epilog=None, version=None, parents=[], formatter_class=<class 'argparse.HelpFormatter'>,
    prefix_chars='-', fromfile_prefix_chars=None, argument_default=None, conflict_handler='error',
    add_help=True)
```

```
    error (message)
```

```
class rebasehelper.cli.CustomHelpFormatter (prog, indent_increment=2,
    max_help_position=24, width=None)
```

3.2.5 Exception module

```
exception rebasehelper.exceptions.CheckerNotFoundError
    Error indicating failure unable to find checker binary.
```

```
exception rebasehelper.exceptions.RebaseHelperError (*args, **kwargs)
    Class representing Error raised inside rebase-helper after intentionally catching some expected and well known exception/error.
```

3.2.6 Logger module

```
class rebasehelper.logger.LoggerHelper
    Helper class for setting up a logger
```

```
    static add_file_handler (logger, path, formatter=None, level=None)
        Adds FileHandler to a given logger
```

Parameters

- **logger** – Logger object to which the file handler will be added
- **path** – Path to file where the debug log will be written

Returns None

static add_stream_handler (*logger, level=None*)

Adds console handler with given severity.

Parameters

- **logger** – logger object to add the handler to
- **level** – severity level

Returns created handler object

static get_basic_logger (*logger_name, level=10*)

Sets-up a basic logger without any handler

Parameters

- **logger_name** – Logger name
- **level** – severity level

Returns created logger

3.2.7 Ouput tool module

class rebasehelper.output_tool.**BaseOutputTool**

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

DEFAULT = False

EXTENSION = 'ext'

NAME = 'name'

static get_default_tool ()

Returns default output tool

classmethod get_extension ()

classmethod get_name ()

classmethod get_report_path (*app*)

static get_supported_tools ()

Returns list of supported output tools

classmethod match (*cmd=None*)

Checks if tool name matches the desired one.

classmethod print_cli_summary (*app*)

Print report of the rebase

Parameters **app** – Application instance

classmethod print_patches_cli ()

Print info about patches

classmethod print_patches_section_cli (*color, patch_type*)

Print info about one of the patches key section

Parameters

- **color** – color used for the message printing
- **patch_type** – string containing key for the patch_dict

classmethod `print_report_file_path()`

Print path to the report file

classmethod `run()`

class `rebasehelper.output_tool.OutputToolRunner`

Class representing the process of running various output tools.

run_output_tools (*logs=None, app=None*)

Runs all output tools.

Parameters

- **log** – Log that probably contains the important message concerning the rebase fail
- **app** – Application class instance

3.2.8 Patch helper module

class `rebasehelper.patch_helper.GitPatchTool`

Class for git command used for patching old and new sources

CMD = 'git'

classmethod `apply_old_patches()`

Function applies a patch to a old/new sources

static `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`

classmethod `call_prep_script(prepare_script_path)`

classmethod `create_prep_script(prepare)`

Function abstract special things from prep section and apply them to old sources

diff_cls = None

exec_prep_script = False

classmethod `init_git(directory)`

Function initialize old and new Git repository

classmethod `match(cmd=None)`

new_repo = None

new_sources = ''

non_interactive = False

old_repo = None

old_sources = ''

output_data = None

patch_sources_by_prep_script = False

patches = None

prep_section = False

classmethod `run_patch(old_dir, new_dir, rest_sources, patches, prep, **kwargs)`

The function can be used for patching one directory against another


```
source_dir = ''
```

```
class rebasehelper.patch_helper.PatchBase
```

Class used for using several patching command tools, ... Each method should overwrite method like `run_check`

```
classmethod match (cmd=None, *args, **kwargs)
```

Method checks whether it is usefull patch method

```
classmethod run_patch (old_dir, new_dir, rest_sources, git_helper, patches, *args, **kwargs)
```

Method will check all patches in relevant package

```
class rebasehelper.patch_helper.Patcher (tool=None)
```

Class representing a process of applying and generating rebased patch using specific tool.

```
patch (old_dir, new_dir, rest_sources, patches, prep, **kwargs)
```

Apply patches and generate rebased patches if needed

Parameters

- **old_dir** – path to dir with old patches
- **new_dir** – path to dir with new patches
- **patches** – old patches
- **rebased_patches** – rebased patches
- **kwargs** –

Returns

```
rebasehelper.patch_helper.register_patch_tool (patch_tool)
```

3.2.9 Specfile module

```
class rebasehelper.specfile.BaseSpecHook
```

Base class for a spec hook

```
classmethod get_categories ()
```

Returns list of categories of a spec hook

```
classmethod get_name ()
```

Returns the name of a spec hook

```
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.specfile.PatchList
```

```
class rebasehelper.specfile.PatchObject (path, index, option)
```

Class represents set of information about patches

```
get_index ()
```

```
get_option ()
```

```
get_patch_name ()
```

`get_path()`

`git_generated = ''`

`index = ''`

`option = ''`

`path = ''`

`set_path(new_path)`

class `rebasehelper.specfile.SpecFile` (*path*, *changelog_entry*, *sources_location=""*, *download=True*)

Class representing a SPEC file

`category = None`

static `construct_string_with_comment` (*lines*)

Wraps the line in a rebase-helper specific comments

Parameters `lines` – line (or list of lines) to be wrapped

Returns list with lines

`copy` (*new_path=None*)

Create a copy of the current object and copy the SPEC file the new object represents to a new location.

Parameters `new_path` – new path to which to copy the SPEC file

Returns copy of the current object

`defined_sections = ['%package', '%description', '%prep', '%build', '%install', '%check`

`download = False`

`download_remote_sources()`

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

Returns None

`extra_version = 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_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_epoch_number ()

Method for getting epoch of the package

Returns

get_extra_version ()

Returns an extra version of the package - like b1, rc2, ...

Returns String

get_extra_version_separator ()

Returns the separator between version and extra version as used by upstream. If there is not separator or extra version, it returns an empty string.

Returns String with the separator between version as extra version as used by upstream.

Return type str

get_full_version ()

Returns the full version string, which is a combination of version, separator and extra version.

Returns String with full version, including the extra version part.

Return type str

get_new_log ()

get_not_used_patches ()

Method returns list of all unapplied patches.

Returns list of PatchObject

get_package_name ()

Function returns a package name

Returns

get_patch_option (*line*)

Function returns a patch options

Parameters *line* –

Returns patch options like -p1

get_patches ()

Method returns list of all applied and not applied patches

Returns list of PatchObject

get_path ()

Return only spec file path

Returns

static get_paths_with_rpm_macros (*files*)

Method modifies paths in passed list to use RPM macros

Parameters *files* – list of absolute paths

Returns modified list of paths with RPM macros

get_prep_section (*complete=False*)

Function returns whole prep section

get_release ()

Method for getting full release string of the package

Returns

get_release_number ()

Method for getting the release of the package

Returns

get_requires ()

Function returns a package requirements

Returns

get_setup_dirname ()

Get dirname from %setup or %autoseup 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

get_spec_section (*section_name*)

Returns the section of selected name

Parameters **section_name** – section name to get

Returns list of lines contained in the selected section

get_version ()

Method returns the version

Returns

hdr = None

insert_changelog (*new_log*)

is_test_suite_enabled ()

Returns whether test suite is enabled during the build time

Returns True if enabled or False if not

modify_spec_files_section (*files*)

Function repairs spec file according to new sources.

Parameters **files** –

Returns

patches = None

path = ''

prep_section = []

redefine_release_with_macro (*macro*)

Method redefines the Release: line to include passed macro and comments out the old line

Parameters `macro` –

Returns

`removed_patches = []`

`revert_redefine_release_with_macro` (*macro*)

Method removes the redefined the Release: line with given macro and uncomments the old Release line.

Parameters `macro` –

Returns

`rpm_sections = {}`

`save` ()

Save changes made to the `spec_content` to the disc and update internal variables

`set_extra_version` (*extra_version*)

Method to update the extra version in the SPEC file. Redefined `Source0` if needed and also changes `Release` accordingly.

Parameters `extra_version` – the extra version string, if any (e.g. 'b1', 'rc2', ...)

Returns None

`set_extra_version_separator` (*separator*)

Set the string that separates the version and extra version

Parameters `separator` –

Returns

`set_release_number` (*release*)

Method to set release number

Parameters `release` –

Returns

`set_spec_section` (*section_name*, *new_section*)

Returns the section of selected name

Parameters `section_name` – section name to get

Returns list of lines contained in the selected section

`set_tag` (*tag*, *value*, *preserve_macros=False*)

Sets value of a tag while trying to preserve macros if requested

`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

`sources = None`

`spc = None`

`spec_content = []`

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 (`new_log`)

Function updates changelog with new version

update_paths_to_patches ()

update_setup_dirname (`dirname`)

Update %setup or %autosetup dirname argument if needed

Parameters `dirname` – new dirname to be used

write_updated_patches (`patches, disable_inapplicable`)

Function writes the patches to -rebase.spec file

class `rebasehelper.specfile.SpecHooksRunner`

Class representing the process of running various spec file hooks.

get_available_spec_hooks ()

Returns a list of all available spec hooks

run_spec_hooks (`spec_file, rebase_spec_file, **kwargs`)

Runs all spec hooks.

Parameters

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

`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

3.2.10 Utils module

class `rebasehelper.utils.ConsoleHelper`

Class for command line interaction with the user.

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

ContextManager for capturing stdout/stderr

classmethod `cprint (message, color=None)`

Print colored output if possible

Parameters

- `color` – color to be used in the output

- **message** – string to be printed out

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

Function for command line messages

Parameters

- **message** – prompt string
- **default_yes** – If the default value is YES
- **any_input** – if True, return input without checking it first

Returns True or False, based on user's input

classmethod should_use_colors (*conf*)

Determine whether ansi colors should be used for CLI output

use_colors = False

class rebasehelper.utils.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*)

exception rebasehelper.utils.DownloadError

Exception indicating that download of a file failed.

class rebasehelper.utils.DownloadHelper

Class for downloading sources defined in SPEC file

static download_file (*url, destination_path, timeout=10, blocksize=8192*)

Method for downloading file from HTTP, HTTPS and FTP URL.

Parameters

- **url** – URL from which to download the file
- **destination_path** – path where to store downloaded file
- **timeout** – timeout in seconds for blocking actions like connecting, etc.
- **blocksize** – size in Bytes of blocks used for downloading the file and reporting progress

Returns None

static progress (*download_total, downloaded, start_time*)

The function prints download progress and remaining time of the download directly to the standard output.

Parameters

- **download_total** (*int or float*) – size of the file which is being downloaded
- **downloaded** (*int or float*) – already downloaded size of the file
- **start_time** (*float*) – time in seconds since the epoch from the point when the download started. This is used to calculate the remaining time of the download.

Returns None

class rebasehelper.utils.**FileHelper**

static **file_available** (*filename*)

class rebasehelper.utils.**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*)

exception rebasehelper.utils.**GitRebaseError**

 Error indicating problems with Git

exception rebasehelper.utils.**GitRuntimeError**

 Error indicating problems with Git

class rebasehelper.utils.**KojiHelper**

baseurl = 'http://kojipkgs.fedoraproject.org/work/'

baseurl_pkg = 'https://kojipkgs.fedoraproject.org/packages/'

ca_cert = '/home/docs/.fedora-server-ca.cert'

cert = '/home/docs/.fedora.cert'

classmethod **display_task_results** (*tasks*)

 Taken from from koji_cli.lib

classmethod **download_build** (*build_id, destination*)

 Downloads all x86_64 RPMs and logs of a Koji build

Parameters

- **build_id** – Koji build ID
- **destination** – target path

Returns (list of paths to RPMs, list of paths to logs)

classmethod **download_scratch_build** (*session, task_list, dir_name*)

functional = **False**

classmethod **get_koji_tasks** (*task_id, dir_name*)

classmethod **get_latest_build** (*package*)

 Searches for the latest Koji build of a package

Parameters **package** – package name

Returns (latest version, Koji build ID)

koji_web = 'koji.fedoraproject.org'

scratch_url = 'http://koji.fedoraproject.org/work/'

server = 'https://koji.fedoraproject.org/kojihub'


```

classmethod session_maker (baseurl=None)

classmethod upload_srpm (session, source)

classmethod watch_koji_tasks (session, tasklist)
    Taken from from koji_cli.lib

exception rebasehelper.utils.LookasideCacheError
    Exception indicating a problem accessing lookaside cache

class rebasehelper.utils.LookasideCacheHelper
    Class for downloading files from Fedora/RHEL lookaside cache

classmethod download (tool, basepath, package)

rpkg_config_dir = '/etc/rpkg'

class rebasehelper.utils.MacroHelper
    Helper class for working with RPM macros

static dump ()
    Returns list of all defined macros

    Returns list of macros

static expand (s, default=None)

static filter (macros, **kwargs)
    Returns all macros satisfying specified filters

    Parameters

    • macros – list of macros to be filtered

    • kwargs – filters

    Returns filtered list of macros

class rebasehelper.utils.PathHelper
    Class which finds a file or files in specific directory

static find_all_files (top_path, pattern)
    Finds a file that matches the given 'pattern' recursively starting in the 'top_path' directory. If found, returns full path to the first occurrence of the file, otherwise returns None.

static find_all_files_current_dir (top_path, pattern)
    Finds all files that matches the given 'pattern' in the 'top_path' directory. If found, returns fields of all files, otherwise returns None.

static find_first_dir_with_file (top_path, pattern)
    Finds a file that matches the given 'pattern' recursively starting in the 'top_path' directory. If found, returns full path to the directory with first occurrence of the file, otherwise returns None.

static find_first_file (top_path, pattern, recursion_level=None)
    Finds a file that matches the given 'pattern' recursively starting in the 'top_path' directory. If found, returns full path to the first occurrence of the file, otherwise returns None.

static get_temp_dir ()
    Returns a path to new temporary directory.

class rebasehelper.utils.ProcessHelper
    Class for execution subprocess

DEV_NULL = '/dev/null'

```

static run_subprocess (*cmd, input=None, output=None*)

Runs the passed command as a subprocess.

Parameters

- **cmd** – command with arguments to be run
- **input** – file to read the input from. If None, read from STDIN
- **output** – file to write the output of the command. If None, write to STDOUT

Returns exit code of the process

static run_subprocess_cwd (*cmd, cwd=None, input=None, output=None, shell=False*)

Runs the passed command as a subprocess in different working directory.

Parameters

- **cmd** – command with arguments to be run
- **cwd** – the directory to change the working dir to
- **input** – file to read the input from. If None, read from STDIN
- **output** – file to write the output of the command. If None, write to STDOUT
- **shell** – if to run the command as shell command (default: False)

Returns exit code of the process

static run_subprocess_cwd_env (*cmd, cwd=None, env=None, input=None, output=None, shell=False*)

Runs the passed command as a subprocess in different working directory with possibly changed ENVIRONMENT VARIABLES.

Parameters

- **cmd** – command with arguments to be run
- **cwd** – the directory to change the working dir to
- **env** – dictionary with ENVIRONMENT VARIABLES to define
- **input** – file to read the input from. If None, read from STDIN
- **output** – file to write the output of the command. If None, write to STDOUT
- **shell** – if to run the command as shell command (default: False)

Returns exit code of the process

static run_subprocess_env (*cmd, env=None, input=None, output=None, shell=False*)

Runs the passed command as a subprocess with possibly changed ENVIRONMENT VARIABLES.

Parameters

- **cmd** – command with arguments to be run
- **env** – dictionary with ENVIRONMENT VARIABLES to define
- **input** – file to read the input from. If None, read from STDIN
- **output** – file to write the output of the command. If None, write to STDOUT
- **shell** – if to run the command as shell command (default: False)

Returns exit code of the process

class rebasehelper.utils.RpmHelper

Helper class for doing various tasks with RPM database, packages, ...

ARCHES = None

static all_packages_installed (*pkg_names=None*)

Check if all packages in passed list are installed.

Parameters *pkg_names* – iterable with package named to check for

Returns True if all packages are installed, False if at least one package is not installed.

static get_arches ()

Get list of all known architectures

static get_header_from_rpm (*rpm_name*)

Function returns a rpm header from given rpm package for later on analysis

Parameters *rpm_name* –

Returns

static get_info_from_rpm (*rpm_name, info*)

Method returns a name of the package from RPM file format

Parameters *pkg_name* –

Returns

static install_build_dependencies (*spec_path=None, assume_yes=False*)

Install all build requires for a package using PolicyKits

Parameters *spec_path* – absolute path to SPEC file

Returns

static is_package_installed (*pkg_name=None*)

Checks whether package with passed name is installed.

Parameters *package_name* – package name we want to check for

Returns True if installed, False if not installed

classmethod parse_spec (*path*)

classmethod split_nevra (*string*)

Splits string into name, epoch, version, release and arch components

class rebasehelper.utils.**TemporaryEnvironment** (*exit_callback=None, **kwargs*)

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

TEMPDIR = 'TEMPDIR'

env ()

Returns copy of *_env* dictionary.

Returns copy of *_env* dictionary

path ()

Returns path to the temporary environment.

Returns abs path to the environment

rebasehelper.utils.**get_value_from_kwargs** (*kwargs, field, source='old'*)

Function returns a part of self.kwargs dictionary

Parameters

- *kwargs* –

- **source** – ‘old’ or ‘new’
- **field** – like ‘patches’, ‘source’

Returns value from dictionary

3.3 Change Log

3.3.1 [Unreleased]

3.3.2 [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

3.3.3 [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

3.3.4 [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

3.3.5 [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

3.3.6 [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

3.3.7 [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

3.3.8 [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

3.3.9 [0.7.1] - 2016-02-22

Added

- Two new command line options used by upstream monitoring

Fixed

- **fedpkg** reimplementation

3.3.10 [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

3.3.11 [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

3.3.12 [0.6.1] - 2015-10-30

Added

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

3.3.13 [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`

3.3.14 [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

3.3.15 [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

3.3.16 [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

3.3.17 [0.3.0]

Added

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

3.3.18 [0.2.0]

Added

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

3.3.19 [0.1.0]

Added

- Initial classes
- CLI interface

r

rebasehelper.application, 12
rebasehelper.archive, 14
rebasehelper.build_helper, 15
rebasehelper.cli, 18
rebasehelper.exceptions, 18
rebasehelper.logger, 18
rebasehelper.output_tool, 19
rebasehelper.patch_helper, 20
rebasehelper.specfile, 21
rebasehelper.utils, 26

Symbols

-build-tasks OLD_TASK,NEW_TASK
 rebase-helper command line option, 10
 -builder-options BUILDER_OPTIONS
 rebase-helper command line option, 10
 -builds-nowait
 rebase-helper command line option, 10
 -buildtool {copr,rpmbuild,mock}
 rebase-helper command line option, 9
 -changelog-entry CHANGELOG_ENTRY
 rebase-helper command line option, 10
 -color {always,never,auto}
 rebase-helper command line option, 9
 -comparepkgs-only COMPAREPKGS_DIR
 rebase-helper command line option, 9
 -config-file CONFIG_FILE
 rebase-helper command line option, 10
 -disable-inapplicable-patches
 rebase-helper command line option, 10
 -get-old-build-from-koji
 rebase-helper command line option, 10
 -non-interactive
 rebase-helper command line option, 10
 -not-download-sources
 rebase-helper command line option, 10
 -outputtool {text,json}
 rebase-helper command line option, 10
 -pkgcomparetool {rpmdiff,abipkgdiff,pkgdiff,csmock}
 rebase-helper command line option, 9
 -results-dir RESULTS_DIR
 rebase-helper command line option, 9
 -skip-version-check
 rebase-helper command line option, 10
 -spec-hook-blacklist {RubyHelper-
 Hook,PyPIURLFixHook,TypoFixHook}
 rebase-helper command line option, 10
 -srpm-builder-options SRPM_BUILDER_OPTIONS
 rebase-helper command line option, 10
 -srpm-buildtool {rpmbuild,mock}

 rebase-helper command line option, 9
 -version
 rebase-helper command line option, 9
 -versioneer {cpan,pypi,rubygems,anitya,npmjs}
 rebase-helper command line option, 10
 -versioneer-blacklist {cpan,pypi,rubygems,anitya,npmjs}
 rebase-helper command line option, 10
 -b, -build-only
 rebase-helper command line option, 9
 -c, -continue
 rebase-helper command line option, 9
 -h, -help
 rebase-helper command line option, 9
 -p, -patch-only
 rebase-helper command line option, 9
 -v, -verbose
 rebase-helper command line option, 9
 -w, -keep-workspace
 rebase-helper command line option, 10

A

accepts_options() (rebasehelper.build_helper.Builder
 method), 16
 accepts_options() (rebase-
 helper.build_helper.BuildToolBase
 class
 method), 15
 add_file_handler() (rebasehelper.logger.LoggerHelper
 static method), 18
 add_stream_handler() (rebasehelper.logger.LoggerHelper
 static method), 18
 all_packages_installed() (rebasehelper.utils.RpmHelper
 static method), 31
 Application (class in rebasehelper.application), 12
 apply_old_patches() (rebase-
 helper.patch_helper.GitPatchTool
 class
 method), 20
 apply_patch() (rebasehelper.patch_helper.GitPatchTool
 static method), 20
 ARCHES (rebasehelper.utils.RpmHelper attribute), 30
 Archive (class in rebasehelper.archive), 14

ArchiveTypeBase (class in rebasehelper.archive), 14

B

BaseOutputTool (class in rebasehelper.output_tool), 19

BaseSpecHook (class in rebasehelper.specfile), 21

baseurl (rebasehelper.utils.KojiHelper attribute), 28

baseurl_pkg (rebasehelper.utils.KojiHelper attribute), 28

BinaryPackageBuildError, 15

build() (rebasehelper.build_helper.Builder method), 16

build() (rebasehelper.build_helper.BuildToolBase class method), 15

build() (rebasehelper.utils.CoprHelper class method), 27

build_packages() (rebasehelper.application.Application method), 12

build_parser() (rebasehelper.cli.CLI static method), 18

build_srpm() (rebasehelper.build_helper.SRPMBuildToolBase class method), 17

build_tools (rebasehelper.build_helper.Builder attribute), 16

Builder (class in rebasehelper.build_helper), 16

BuildTemporaryEnvironment (class in rebasehelper.build_helper), 15

BuildToolBase (class in rebasehelper.build_helper), 15

Bz2ArchiveType (class in rebasehelper.archive), 14

C

ca_cert (rebasehelper.utils.KojiHelper attribute), 28

call_prep_script() (rebasehelper.patch_helper.GitPatchTool class method), 20

category (rebasehelper.specfile.SpecFile attribute), 22

cert (rebasehelper.utils.KojiHelper attribute), 28

CheckerNotFoundError, 18

CLI (class in rebasehelper.cli), 18

CliHelper (class in rebasehelper.cli), 18

CMD (rebasehelper.patch_helper.GitPatchTool attribute), 20

ConsoleHelper (class in rebasehelper.utils), 26

ConsoleHelper.Capturer (class in rebasehelper.utils), 26

construct_string_with_comment() (rebasehelper.specfile.SpecFile static method), 22

CoprHelper (class in rebasehelper.utils), 27

copy() (rebasehelper.specfile.SpecFile method), 22

cprint() (rebasehelper.utils.ConsoleHelper class method), 26

create_prep_script() (rebasehelper.patch_helper.GitPatchTool class method), 20

create_project() (rebasehelper.utils.CoprHelper class method), 27

creates_tasks() (rebasehelper.build_helper.Builder method), 16

creates_tasks() (rebasehelper.build_helper.BuildToolBase class method), 15

CustomAction (class in rebasehelper.cli), 18

CustomArgumentParser (class in rebasehelper.cli), 18

CustomHelpFormatter (class in rebasehelper.cli), 18

D

debug_log_file (rebasehelper.application.Application attribute), 12

DEFAULT (rebasehelper.build_helper.BuildToolBase attribute), 15

DEFAULT (rebasehelper.build_helper.SRPMBuildToolBase attribute), 17

DEFAULT (rebasehelper.output_tool.BaseOutputTool attribute), 19

defined_sections (rebasehelper.specfile.SpecFile attribute), 22

DEV_NULL (rebasehelper.utils.ProcessHelper attribute), 29

diff_cls (rebasehelper.patch_helper.GitPatchTool attribute), 20

display_task_results() (rebasehelper.utils.KojiHelper class method), 28

download (rebasehelper.specfile.SpecFile attribute), 22

download() (rebasehelper.utils.LookasideCacheHelper class method), 29

download_build() (rebasehelper.utils.CoprHelper class method), 27

download_build() (rebasehelper.utils.KojiHelper class method), 28

download_file() (rebasehelper.utils.DownloadHelper static method), 27

download_remote_sources() (rebasehelper.specfile.SpecFile method), 22

download_scratch_build() (rebasehelper.utils.KojiHelper class method), 28

DownloadError, 27

DownloadHelper (class in rebasehelper.utils), 27

dump() (rebasehelper.utils.MacroHelper static method), 29

E

env() (rebasehelper.utils.TemporaryEnvironment method), 31

environment variable

PACKAGE, 8

REPOSITORY, 8

error() (rebasehelper.cli.CustomArgumentParser method), 18

exec_prep_script (rebasehelper.patch_helper.GitPatchTool attribute), 20

expand() (rebasehelper.utils.MacroHelper static method), 29

- EXTENSION (rebasehelper.archive.ArchiveTypeBase attribute), 14
- EXTENSION (rebasehelper.archive.Bz2ArchiveType attribute), 14
- EXTENSION (rebasehelper.archive.GemPseudoArchiveType attribute), 14
- EXTENSION (rebasehelper.archive.TarBz2ArchiveType attribute), 14
- EXTENSION (rebasehelper.archive.TarGzArchiveType attribute), 14
- EXTENSION (rebasehelper.archive.TarXzArchiveType attribute), 14
- EXTENSION (rebasehelper.archive.TgzArchiveType attribute), 15
- EXTENSION (rebasehelper.archive.ZipArchiveType attribute), 15
- EXTENSION (rebasehelper.output_tool.BaseOutputTool attribute), 19
- extra_version (rebasehelper.specfile.SpecFile attribute), 22
- extract() (rebasehelper.archive.ArchiveTypeBase class method), 14
- extract() (rebasehelper.archive.Bz2ArchiveType class method), 14
- extract() (rebasehelper.archive.GemPseudoArchiveType class method), 14
- extract() (rebasehelper.archive.TarGzArchiveType class method), 14
- extract() (rebasehelper.archive.TarXzArchiveType class method), 15
- extract() (rebasehelper.archive.ZipArchiveType class method), 15
- extract_archive() (rebasehelper.application.Application static method), 12
- extract_archive() (rebasehelper.archive.Archive method), 14
- extract_sources() (rebasehelper.application.Application static method), 12
- extract_version_from_archive_name() (rebasehelper.specfile.SpecFile static method), 22
- ## F
- file_available() (rebasehelper.utils.FileHelper static method), 28
- FileHelper (class in rebasehelper.utils), 28
- filter() (rebasehelper.utils.MacroHelper static method), 29
- find_all_files() (rebasehelper.utils.PathHelper static method), 29
- find_all_files_current_dir() (rebasehelper.utils.PathHelper static method), 29
- find_archive_target_in_prep() (rebasehelper.specfile.SpecFile method), 22
- find_first_dir_with_file() (rebasehelper.utils.PathHelper static method), 29
- find_first_file() (rebasehelper.utils.PathHelper static method), 29
- functional (rebasehelper.utils.KojiHelper attribute), 28
- ## G
- GemPseudoArchiveType (class in rebasehelper.archive), 14
- generate_patch() (rebasehelper.application.Application method), 12
- get_all_log_files() (rebasehelper.application.Application method), 12
- get_applied_patches() (rebasehelper.specfile.SpecFile method), 22
- get_arches() (rebasehelper.utils.RpmHelper static method), 31
- get_archive() (rebasehelper.specfile.SpecFile method), 23
- get_available_spec_hooks() (rebasehelper.specfile.SpecHooksRunner method), 26
- get_basic_logger() (rebasehelper.logger.LoggerHelper static method), 19
- get_build_status() (rebasehelper.utils.CoprHelper class method), 27
- get_build_tool_name() (rebasehelper.build_helper.BuildToolBase class method), 15
- get_build_tool_name() (rebasehelper.build_helper.SRPMBuildToolBase class method), 17
- get_build_url() (rebasehelper.utils.CoprHelper class method), 27
- get_builder_options() (rebasehelper.build_helper.BuildToolBase static method), 15
- get_categories() (rebasehelper.specfile.BaseSpecHook class method), 21
- get_checker_outputs() (rebasehelper.application.Application method), 12
- get_client() (rebasehelper.utils.CoprHelper class method), 27
- get_default_tool() (rebasehelper.build_helper.Builder class method), 16
- get_default_tool() (rebasehelper.build_helper.SRPMBuilder class method), 17
- get_default_tool() (rebasehelper.output_tool.BaseOutputTool static method), 19
- get_detached_task() (rebasehelper.build_helper.Builder method), 16

get_detached_task() (rebasehelper.build_helper.BuildToolBase class method), 15
 get_email() (rebasehelper.utils.GitHelper class method), 28
 get_epoch_number() (rebasehelper.specfile.SpecFile method), 23
 get_extension() (rebasehelper.output_tool.BaseOutputTool class method), 19
 get_extra_version() (rebasehelper.specfile.SpecFile method), 23
 get_extra_version_separator() (rebasehelper.specfile.SpecFile method), 23
 get_full_version() (rebasehelper.specfile.SpecFile method), 23
 get_header_from_rpm() (rebasehelper.utils.RpmHelper static method), 31
 get_index() (rebasehelper.specfile.PatchObject method), 21
 get_info_from_rpm() (rebasehelper.utils.RpmHelper static method), 31
 get_koji_tasks() (rebasehelper.utils.KojiHelper class method), 28
 get_latest_build() (rebasehelper.utils.KojiHelper class method), 28
 get_logs() (rebasehelper.build_helper.Builder method), 16
 get_logs() (rebasehelper.build_helper.BuildToolBase class method), 16
 get_message() (rebasehelper.utils.ConsoleHelper static method), 27
 get_name() (rebasehelper.output_tool.BaseOutputTool class method), 19
 get_name() (rebasehelper.specfile.BaseSpecHook class method), 21
 get_new_build_logs() (rebasehelper.application.Application method), 12
 get_new_log() (rebasehelper.specfile.SpecFile method), 23
 get_not_used_patches() (rebasehelper.specfile.SpecFile method), 23
 get_option() (rebasehelper.specfile.PatchObject method), 21
 get_package_name() (rebasehelper.specfile.SpecFile method), 23
 get_patch_name() (rebasehelper.specfile.PatchObject method), 21
 get_patch_option() (rebasehelper.specfile.SpecFile method), 23
 get_patches() (rebasehelper.specfile.SpecFile method), 23
 get_path() (rebasehelper.specfile.PatchObject method), 22
 get_path() (rebasehelper.specfile.SpecFile method), 23
 get_paths_with_rpm_macros() (rebasehelper.specfile.SpecFile static method), 23
 get_prep_section() (rebasehelper.specfile.SpecFile method), 24
 get_rebase_name() (in module rebasehelper.specfile), 26
 get_rebased_patches() (rebasehelper.application.Application method), 12
 get_rebasehelper_data() (rebasehelper.application.Application method), 12
 get_release() (rebasehelper.specfile.SpecFile method), 24
 get_release_number() (rebasehelper.specfile.SpecFile method), 24
 get_report_path() (rebasehelper.output_tool.BaseOutputTool class method), 19
 get_requires() (rebasehelper.specfile.SpecFile method), 24
 get_rpm_packages() (rebasehelper.application.Application method), 13
 get_setup_dirname() (rebasehelper.specfile.SpecFile method), 24
 get_sources() (rebasehelper.specfile.SpecFile method), 24
 get_spec_section() (rebasehelper.specfile.SpecFile method), 24
 get_srpm_builder_options() (rebasehelper.build_helper.BuildToolBase static method), 16
 get_srpm_buildtool() (rebasehelper.build_helper.BuildToolBase static method), 16
 get_supported_archives() (rebasehelper.archive.Archive class method), 14
 get_supported_tools() (rebasehelper.build_helper.Builder class method), 16
 get_supported_tools() (rebasehelper.build_helper.SRPMBuilder class method), 17
 get_supported_tools() (rebasehelper.output_tool.BaseOutputTool static method), 19
 get_task_info() (rebasehelper.build_helper.Builder method), 17
 get_task_info() (rebasehelper.build_helper.BuildToolBase class method), 16
 get_temp_dir() (rebasehelper.utils.PathHelper static method), 29
 get_user() (rebasehelper.utils.GitHelper class method), 28
 get_value_from_kwargs() (in module rebasehelper.utils),

31
 get_version() (rebasehelper.specfile.SpecFile method), 24
 git_generated (rebasehelper.specfile.PatchObject attribute), 22
 GIT_USER_EMAIL (rebasehelper.utils.GitHelper attribute), 28
 GIT_USER_NAME (rebasehelper.utils.GitHelper attribute), 28
 GitHelper (class in rebasehelper.utils), 28
 GitPatchTool (class in rebasehelper.patch_helper), 20
 GitRebaseError, 28
 GitRuntimeError, 28

H

hdr (rebasehelper.specfile.SpecFile attribute), 24

I

index (rebasehelper.specfile.PatchObject attribute), 22
 init_git() (rebasehelper.patch_helper.GitPatchTool class method), 20
 insert_changelog() (rebasehelper.specfile.SpecFile method), 24
 install_build_dependencies() (rebasehelper.utils.RpmHelper static method), 31
 is_default() (rebasehelper.build_helper.BuildToolBase class method), 16
 is_default() (rebasehelper.build_helper.SRPMBuildToolBase class method), 17
 is_package_installed() (rebasehelper.utils.RpmHelper static method), 31
 is_test_suite_enabled() (rebasehelper.specfile.SpecFile method), 24

K

koji_web (rebasehelper.utils.KojiHelper attribute), 28
 KojiHelper (class in rebasehelper.utils), 28
 kwargs (rebasehelper.application.Application attribute), 13

L

load_build_tools() (rebasehelper.build_helper.Builder class method), 17
 load_srpm_build_tools() (rebasehelper.build_helper.SRPMBuildToolBase class method), 17
 LoggerHelper (class in rebasehelper.logger), 18
 LookasideCacheError, 29
 LookasideCacheHelper (class in rebasehelper.utils), 29

M

MacroHelper (class in rebasehelper.utils), 29
 match() (rebasehelper.archive.ArchiveTypeBase class method), 14

match() (rebasehelper.archive.ZipArchiveType class method), 15
 match() (rebasehelper.build_helper.BuildToolBase class method), 16
 match() (rebasehelper.output_tool.BaseOutputTool class method), 19
 match() (rebasehelper.patch_helper.GitPatchTool class method), 20
 match() (rebasehelper.patch_helper.PatchBase class method), 21
 modify_spec_files_section() (rebasehelper.specfile.SpecFile method), 24

N

NAME (rebasehelper.output_tool.BaseOutputTool attribute), 19
 new_repo (rebasehelper.patch_helper.GitPatchTool attribute), 20
 new_rest_sources (rebasehelper.application.Application attribute), 13
 new_sources (rebasehelper.application.Application attribute), 13
 new_sources (rebasehelper.patch_helper.GitPatchTool attribute), 20
 non_interactive (rebasehelper.patch_helper.GitPatchTool attribute), 20

O

old_repo (rebasehelper.patch_helper.GitPatchTool attribute), 20
 old_rest_sources (rebasehelper.application.Application attribute), 13
 old_sources (rebasehelper.application.Application attribute), 13
 old_sources (rebasehelper.patch_helper.GitPatchTool attribute), 20
 open() (rebasehelper.archive.ArchiveTypeBase class method), 14
 open() (rebasehelper.archive.Bz2ArchiveType class method), 14
 open() (rebasehelper.archive.GemPseudoArchiveType class method), 14
 open() (rebasehelper.archive.TarGzArchiveType class method), 14
 open() (rebasehelper.archive.TarXzArchiveType class method), 15
 open() (rebasehelper.archive.ZipArchiveType class method), 15
 option (rebasehelper.specfile.PatchObject attribute), 22
 output_data (rebasehelper.patch_helper.GitPatchTool attribute), 20
 OutputToolRunner (class in rebasehelper.output_tool), 20

P

PACKAGE, 8

parse_spec() (rebasehelper.utils.RpmHelper class method), 31

patch() (rebasehelper.patch_helper.Patcher method), 21

patch_sources() (rebasehelper.application.Application method), 13

patch_sources_by_prep_script (rebasehelper.patch_helper.GitPatchTool attribute), 20

PatchBase (class in rebasehelper.patch_helper), 21

Patcher (class in rebasehelper.patch_helper), 21

patches (rebasehelper.patch_helper.GitPatchTool attribute), 20

patches (rebasehelper.specfile.SpecFile attribute), 24

PatchList (class in rebasehelper.specfile), 21

PatchObject (class in rebasehelper.specfile), 21

path (rebasehelper.specfile.PatchObject attribute), 22

path (rebasehelper.specfile.SpecFile attribute), 24

path() (rebasehelper.utils.TemporaryEnvironment method), 31

PathHelper (class in rebasehelper.utils), 29

prep_section (rebasehelper.patch_helper.GitPatchTool attribute), 20

prep_section (rebasehelper.specfile.SpecFile attribute), 24

prepare() (rebasehelper.build_helper.Builder method), 17

prepare() (rebasehelper.build_helper.BuildToolBase class method), 16

prepare_sources() (rebasehelper.application.Application method), 13

print_cli_summary() (rebasehelper.output_tool.BaseOutputTool class method), 19

print_patches_cli() (rebasehelper.output_tool.BaseOutputTool class method), 19

print_patches_section_cli() (rebasehelper.output_tool.BaseOutputTool class method), 19

print_report_file_path() (rebasehelper.output_tool.BaseOutputTool class method), 19

print_summary() (rebasehelper.application.Application method), 13

print_task_info() (rebasehelper.application.Application method), 13

ProcessHelper (class in rebasehelper.utils), 29

progress() (rebasehelper.utils.DownloadHelper static method), 27

R

rebase-helper command line option

–build-tasks OLD_TASK,NEW_TASK, 10

–builder-options BUILDER_OPTIONS, 10

–builds-nowait, 10

–buildtool {copr,rpmbuild,mock}, 9

–changelog-entry CHANGELOG_ENTRY, 10

–color {always,never,auto}, 9

–comparepkgs-only COMPAREPKGS_DIR, 9

–config-file CONFIG_FILE, 10

–disable-inapplicable-patches, 10

–get-old-build-from-koji, 10

–non-interactive, 10

–not-download-sources, 10

–outputtool {text,json}, 10

–pkgcomparetool {rpmdiff,abipkgdiff,pkgdiff,csmock}, 9

–results-dir RESULTS_DIR, 9

–skip-version-check, 10

–spec-hook-blacklist {RubyHelperHook,PyPIURLFixHook,TypoFixHook}, 10

–srpm-builder-options SRPM_BUILDER_OPTIONS, 10

–srpm-buildtool {rpmbuild,mock}, 9

–version, 9

–versioneer {cpan,pypi,rubygems,anitya,npmjs}, 10

–versioneer-blacklist {cpan,pypi,rubygems,anitya,npmjs}, 10

–b, –build-only, 9

–c, –continue, 9

–h, –help, 9

–p, –patch-only, 9

–v, –verbose, 9

–w, –keep-workspace, 10
SOURCES, 9

rebase_spec_file (rebasehelper.application.Application attribute), 13

rebase_spec_file_path (rebasehelper.application.Application attribute), 13

rebased_patches (rebasehelper.application.Application attribute), 13

rebased_repo (rebasehelper.application.Application attribute), 13

rebasehelper.application (module), 12

rebasehelper.archive (module), 14

rebasehelper.build_helper (module), 15

rebasehelper.cli (module), 18

rebasehelper.exceptions (module), 18

rebasehelper.logger (module), 18

rebasehelper.output_tool (module), 19

rebasehelper.patch_helper (module), 20

rebasehelper.specfile (module), 21

rebasehelper.utils (module), 26

RebaseHelperError, 18

redefine_release_with_macro() (rebasehelper.specfile.SpecFile method), 24

register_archive_type() (in module rebasehelper.archive), 15

register_patch_tool() (in module rebasehelper.patch_helper), 21

removed_patches (rebasehelper.specfile.SpecFile attribute), 25

report_log_file (rebasehelper.application.Application attribute), 13

REPOSITORY, 8

result_file (rebasehelper.application.Application attribute), 13

revert_redefine_release_with_macro() (rebasehelper.specfile.SpecFile method), 25

rpkg_config_dir (rebasehelper.utils.LookasideCacheHelper attribute), 29

rpm_sections (rebasehelper.specfile.SpecFile attribute), 25

RpmbuildTemporaryEnvironment (class in rebasehelper.build_helper), 17

RpmHelper (class in rebasehelper.utils), 30

run() (rebasehelper.application.Application method), 13

run() (rebasehelper.cli.CliHelper static method), 18

run() (rebasehelper.output_tool.BaseOutputTool class method), 20

run() (rebasehelper.specfile.BaseSpecHook class method), 21

run_download_compare() (rebasehelper.application.Application method), 13

run_mergetool() (rebasehelper.utils.GitHelper class method), 28

run_output_tools() (rebasehelper.output_tool.OutputToolRunner method), 20

run_package_checkers() (rebasehelper.application.Application method), 13

run_patch() (rebasehelper.patch_helper.GitPatchTool class method), 20

run_patch() (rebasehelper.patch_helper.PatchBase class method), 21

run_spec_hooks() (rebasehelper.specfile.SpecHooksRunner method), 26

run_subprocess() (rebasehelper.utils.ProcessHelper static method), 29

run_subprocess_cwd() (rebasehelper.utils.ProcessHelper static method), 30

run_subprocess_cwd_env() (rebasehelper.utils.ProcessHelper static method), 30

run_subprocess_env() (rebasehelper.utils.ProcessHelper static method), 30

S

save() (rebasehelper.specfile.SpecFile method), 25

scratch_url (rebasehelper.utils.KojiHelper attribute), 28

server (rebasehelper.utils.KojiHelper attribute), 28

session_maker() (rebasehelper.utils.KojiHelper class method), 28

set_extra_version() (rebasehelper.specfile.SpecFile method), 25

set_extra_version_separator() (rebasehelper.specfile.SpecFile method), 25

set_path() (rebasehelper.specfile.PatchObject method), 22

set_release_number() (rebasehelper.specfile.SpecFile method), 25

set_spec_section() (rebasehelper.specfile.SpecFile method), 25

set_tag() (rebasehelper.specfile.SpecFile method), 25

set_upstream_monitoring() (rebasehelper.application.Application method), 13

set_version() (rebasehelper.specfile.SpecFile method), 25

set_version_using_archive() (rebasehelper.specfile.SpecFile method), 25

setup() (rebasehelper.application.Application static method), 13

should_use_colors() (rebasehelper.utils.ConsoleHelper class method), 27

source_dir (rebasehelper.patch_helper.GitPatchTool attribute), 20

SourcePackageBuildError, 18

SOURCES

- rebase-helper command line option, 9

sources (rebasehelper.specfile.SpecFile attribute), 25

spc (rebasehelper.specfile.SpecFile attribute), 25

spec_content (rebasehelper.specfile.SpecFile attribute), 25

spec_file (rebasehelper.application.Application attribute), 13

spec_file_path (rebasehelper.application.Application attribute), 13

SpecFile (class in rebasehelper.specfile), 22

SpecHooksRunner (class in rebasehelper.specfile), 26

split_nevra() (rebasehelper.utils.RpmHelper class method), 31

split_version_string() (rebasehelper.specfile.SpecFile static method), 26

srpm_build_tools (rebasehelper.build_helper.SRPMBuild attribute), 17

SRPMBuild (class in rebasehelper.build_helper), 17

SRPMBuildToolBase (class in rebasehelper.build_helper), 17

T

TarBz2ArchiveType (class in rebasehelper.archive), 14

TarGzArchiveType (class in rebasehelper.archive), 14
TarXzArchiveType (class in rebasehelper.archive), 14
temp_dir (rebasehelper.application.Application attribute), 13
TEMPDIR (rebasehelper.utils.TemporaryEnvironment attribute), 31
TEMPDIR_BUILD (rebasehelper.build_helper.RpmbuildTemporaryEnvironment attribute), 17
TEMPDIR_BUILDROOT (rebasehelper.build_helper.RpmbuildTemporaryEnvironment attribute), 17
TEMPDIR_RESULTS (rebasehelper.build_helper.BuildTemporaryEnvironment attribute), 15
TEMPDIR_RPMBUILD (rebasehelper.build_helper.RpmbuildTemporaryEnvironment attribute), 17
TEMPDIR_RPMS (rebasehelper.build_helper.RpmbuildTemporaryEnvironment attribute), 17
TEMPDIR_SOURCES (rebasehelper.build_helper.BuildTemporaryEnvironment attribute), 15
TEMPDIR_SPEC (rebasehelper.build_helper.BuildTemporaryEnvironment attribute), 15
TEMPDIR_SPECS (rebasehelper.build_helper.BuildTemporaryEnvironment attribute), 15
TEMPDIR_SRPMS (rebasehelper.build_helper.RpmbuildTemporaryEnvironment attribute), 17
TemporaryEnvironment (class in rebasehelper.utils), 31
TgzArchiveType (class in rebasehelper.archive), 15
wait_for_task() (rebasehelper.build_helper.BuildToolBase class method), 16
watch_build() (rebasehelper.utils.CoprHelper class method), 27
watch_koji_tasks() (rebasehelper.utils.KojiHelper class method), 29
write_updated_patches() (rebasehelper.specfile.SpecFile method), 26

Z

ZipArchiveType (class in rebasehelper.archive), 15

U

update_changelog() (rebasehelper.specfile.SpecFile method), 26
update_paths_to_patches() (rebasehelper.specfile.SpecFile method), 26
update_setup_dirname() (rebasehelper.specfile.SpecFile method), 26
upload_srpm() (rebasehelper.utils.KojiHelper class method), 29
upstream_monitoring (rebasehelper.application.Application attribute), 13
use_colors (rebasehelper.utils.ConsoleHelper attribute), 27

W

wait_for_task() (rebasehelper.build_helper.Builder method), 17