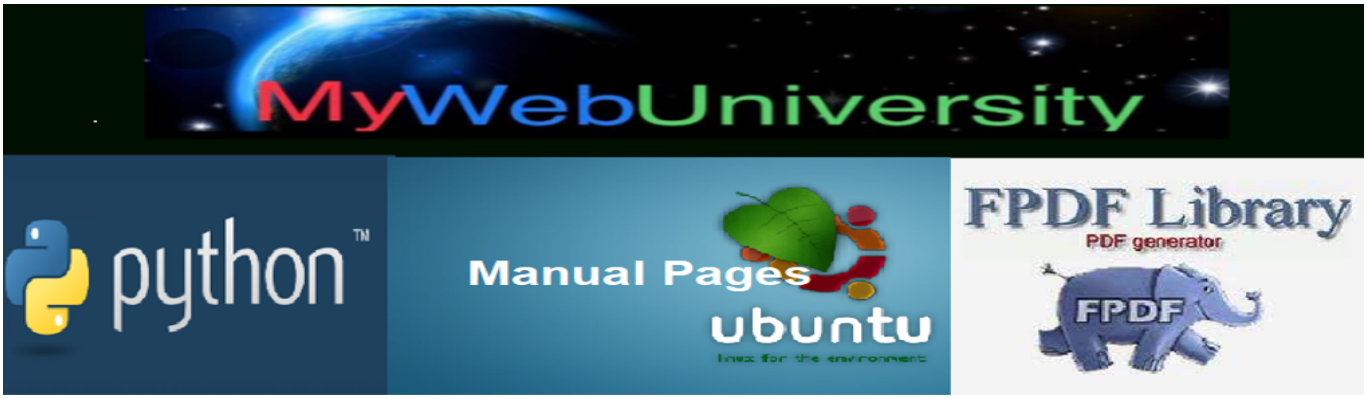

Rocky Enterprise Linux 9.2 Manual Pages on command 'dsc.5'

$ man dsc.5

dsc(5) dpkg suite dsc(5)

NAME

dsc - Debian source packages' control file format

SYNOPSIS

filename.dsc

DESCRIPTION

Each Debian source package is composed of a .dsc control file, which contains a number of fields, in deb822(5) format.

Each field begins with a tag, such as Source or Binary (case insensitive), followed by a colon, and the body of the field (case sensitive unless stated otherwise). Fields are delimited only by field tags. In other words, field text may be multiple lines in length, but the installation tools will generally join lines when processing the body of the field (except in case of the multiline fields Package-List, Files, Checksums-Sha1 and Checksums-Sha256, see below).

The control data might be enclosed in an OpenPGP ASCII Armored signature, as specified in RFC4880.

FIELDS

Format: format-version (required)

The value of this field declares the format version of the source package. The field value is used by programs acting on a source package to interpret the list of files in the source package and determine how to unpack it. The syntax of the field value is a numeric major revision (?0-9?), a period (?.?), a numeric minor revision (?0-9?), and then an optional subtype after whitespace (? \t?), which if specified is a lowercase

alphanumeric (?a-z0-9?) word in parentheses (?()?). The subtype is optional in the syntax but may be mandatory for particular source format revisions.

The source formats currently supported by dpkg are 1.0, 2.0, 3.0 (native), 3.0 (quilt), 3.0 (git), 3.0 (bzr) and 3.0 (custom). See dpkg-source(1) for their description.

Source: source-name (required)

The value of this field determines the package name, and is used to generate file names by most installation tools.

Binary: binary-package-list

This folded field lists binary packages which this source package can produce, separated by commas.

This field has now been superseded by the Package-List field, which gives enough information about what binary packages are produced on which architecture, build-profile and other involved restrictions.

Architecture: arch-list (recommended)

A list of architectures and architecture wildcards separated by spaces which specify the type of hardware this package can be compiled for. Common architecture names and architecture wildcards are amd64, armel, i386, linux-any, any-amd64, etc.

Note that the all value is meant for packages that are architecture independent, and any for packages that are architecture dependent. The list may include (or consist solely of) the special value all. When the list contains the architecture wildcard any, the only other value allowed in the list is all.

The field value is generally generated from Architecture fields from in the debian/control in the source package.

Version: version-string (required)

Typically, this is the original package's version number in whatever form the program's author uses. It may also include a Debian revision number (for non-native packages). The exact format and sorting algorithm are described in deb-version(7).

Origin: name

The name of the distribution this package is originating from.

Maintainer: fullname-email (recommended)

Should be in the format ?Joe Bloggs <jbloggs@foo.com>?, and is typically the person who created the package, as opposed to the author of the software that was packaged.

Uploaders: fullname-email-list

Lists all the names and email addresses of co-maintainers of the package, in the same format as the Maintainer field. Multiple co-maintainers should be separated by a comma.

Description short-description

long-description

The format for the source package description is a short brief summary on the first line (after the Description field). The following lines should be used as a longer, more detailed description. Each line of the long description must be preceded by a space, and blank lines in the long description must contain a single ?.? following the preceding space.

Homepage: url

The upstream project home page url.

Standards-Version: version-string (recommended)

This documents the most recent version of the distribution policy standards this package complies with.

Vcs-Browser: url

The url of a web interface to browse the Version Control System repository.

Vcs-Arch: url

Vcs-Bzr: url

Vcs-Cvs: url

Vcs-Darcs: url

Vcs-Git: url

Vcs-Hg: url

Vcs-Mtn: url

Vcs-Svn: url

These fields declare the url of the Version Control System repository used to maintain this package. See deb-src-control(5) for more details.

Testsuite: name-list

This field declares that the source package contains the specified test suites. The value is a comma-separated list of test suites. If the autopkgtest value is present, a debian/tests/control is expected to be present, if the file is present but not the value, then dpkg-source will automatically add it, preserving previous values.

Testsuite-Triggers: package-list

This field declares the comma-separated union of all test dependencies (Depends fields in debian/tests/control file), with all restrictions removed, and OR dependencies flattened (that is, converted to separate AND relationships), except for binaries generated by this source package and its meta-dependency equivalent @.

Rationale: this field is needed because otherwise to be able to get the test dependencies, each source package would need to be unpacked.

Build-Depends: package-list

Build-Depends-Arch: package-list

Build-Depends-Indep: package-list

Build-Conflicts: package-list

Build-Conflicts-Arch: package-list

Build-Conflicts-Indep: package-list

These fields declare relationships between the source package and packages used to build it. They are discussed in the deb-src-control(5) manpage.

Package-List:

?package package-type section priority key-value-list

This multiline field contains a list of binary packages generated by this source package.

The package is the binary package name.

The package-type is the binary package type, usually deb, another common value is udeb.

The section and priority match the binary package fields of the same name.

The key-value-list is a space separated key=value list, and the currently known optional keys are:

arch

The architecture restriction from the binary package Architecture field, with spaces converted to ?,?.

profile

The normalized build-profile restriction formula from the binary package Build-Profile field, with ORs converted to ?+? and ANDs to ?,?.

protected

If the binary package is protected, this key will contain the value of the

Protected field, that is a yes value.

Supported since dpkg 1.20.1.

essential

If the binary package is essential, this key will contain the value of the

Essential field, that is a yes value.

Files: (required)

Checksums-Sha1: (required)

Checksums-Sha256: (required)

?checksum size filename

These multiline fields contain a list of files with a checksum and size for each one.

These fields have the same syntax and differ only in the checksum algorithm used: MD5 for Files, SHA-1 for Checksums-Sha1 and SHA-256 for Checksums-Sha256.

The first line of the field value (the part on the same line as the field name followed by a colon) is always empty. The content of the field is expressed as continuation lines, one line per file. Each line consists of the checksum, a space, the file size, a space, and the file name.

These fields list all files that make up the source package. The list of files in these fields must match the list of files in the other related fields.

BUGS

The Format field conflates the format for the .dsc file itself and the format of the extracted source package.

SEE ALSO

deb822(5), deb-src-control(5), deb-version(7), dpkg-source(1).