zplugin-install.zsh(1)

NAME

zplugin-install.zsh - a shell script

SYNOPSIS

Documentation automatically generated with `zshelldoc'

FUNCTIONS

```
-zplg-at-eval
-zplg-compile-plugin
-zplg-download-file-stdout
-zplg-download-snippet
-zplg-forget-completion
-zplg-get-latest-gh-r-version
-zplg-handle-binary-file
-zplg-install-completions
-zplg-mirror-using-svn
-zplg-setup-plugin-dir
```

DETAILS

Script Body

Has 3 line(s). No functions are called (may attach to Zle or do hooks, etc.).

-zplg-at-eval

Has 1 line(s). Doesn't call other functions.

Uses feature(s): *eval*

Called by:

```
-zplg-download-snippet
```

-zplg-compile-plugin

```
# Compiles given plugin (its main source file, and also an
# additional "....zsh" file if it exists).
#
# $1 - plugin spec (4 formats: user---plugin, user/plugin, user, plugin)
# $2 - plugin (only when $1 - i.e. user - given)
```

Has 30 line(s). Calls functions:

```
-zplg-compile-plugin
|-- zplugin-side.zsh/-zplg-first
`-- zplugin.zsh/-zplg-any-to-user-plugin
```

Uses feature(s): *zcompile*

Called by:

```
-zplg-setup-plugin-dir
zplugin-autoload.zsh/-zplg-compile-uncompile-all
zplugin.zsh/zplugin
```

-zplg-download-file-stdout

```
# Downloads file to stdout. Supports following backend commands:
# curl, wget, lftp, lynx. Used by snippet loading.
```

Has 32 line(s). Calls functions:

```
-zplg-download-file-stdout
```

Uses feature(s): *type*

Called by:

```
-zplg-download-snippet
-zplg-setup-plugin-dir
```

-zplg-download-snippet

```
# Downloads snippet — either a file — with curl, wget, lftp or lynx,
# or a directory, with Subversion — when svn-ICE is active. Github
# supports Subversion protocol and allows to clone subdirectories.
# This is used to provide a layer of support for Oh-My-Zsh and Prezto.
```

Has 111 line(s). Calls functions:

```
-zplg-download-snippet
|-- -zplg-at-eval
|-- -zplg-download-file-stdout
|-- -zplg-install-completions
|   |-- -zplg-forget-completion
|   |-- zplugin-side.zsh/-zplg-any-colorify-as-uspl2
|   |-- zplugin-side.zsh/-zplg-exists-physically-message
|   `-- zplugin.zsh/-zplg-any-to-user-plugin
|-- -zplg-mirror-using-svn
`-- zplugin-side.zsh/-zplg-store-ices
```

Uses feature(s): *eval, zcompile*

Called by:

```
zplugin.zsh/-zplg-load-snippet
```

-zplg-forget-completion

```
# Implements alternation of Zsh state so that already initialized
# completion stops being visible to Zsh.
#
# $1 - completion function name, e.g. "_cp"; can also be "cp"
```

Has 15 line(s). Doesn't call other functions.

Uses feature(s): *unfunction*

Called by:

```
-zplg-install-completions
zplugin-autoload.zsh/-zplg-compinit
zplugin-autoload.zsh/-zplg-uninstall-completions
zplugin.zsh/zplugin
```

-zplg-get-latest-gh-r-version

```
# Gets version string of latest release of given Github
# package. Connects to Github releases page.
```

Has 14 line(s). Calls functions:

```
-zplg-get-latest-gh-r-version
`-- zplugin.zsh/-zplg-any-to-user-plugin
```

Called by:

```
zplugin-autoload.zsh/-zplg-update-or-status
```

-zplg-handle-binary-file

```
# If the file is an archive, it is extracted by this function.
# Next stage is scanning of files with the common utility `file',
# to detect executables. They are given +x mode. There are also
# messages to the user on performed actions.
#
# $1 - url
# $2 - file
```

Has 53 line(s). Doesn't call other functions.

Uses feature(s): *unfunction*

Called by:

```
-zplg-setup-plugin-dir
```

-zplg-install-completions

```
# Installs all completions of given plugin. After that they are
# visible to `compinit'. Visible completions can be selectively
# disabled and enabled. User can access completion data with
# `clist' or `completions' subcommand.
#
# $1 - plugin spec (4 formats: user---plugin, user/plugin, user, plugin)
# $2 - plugin (only when $1 - i.e. user - given)
# $3 - if 1, then reinstall, otherwise only install completions that aren't there
```

Has 39 line(s). Calls functions:

```
-zplg-install-completions
|-- -zplg-forget-completion
|-- zplugin-side.zsh/-zplg-any-colorify-as-uspl2
|-- zplugin-side.zsh/-zplg-exists-physically-message
`-- zplugin.zsh/-zplg-any-to-user-plugin
```

Called by:

```
-zplg-download-snippet
-zplg-setup-plugin-dir
zplugin.zsh/zplugin
```

-zplg-mirror-using-svn

```
# Used to clone subdirectories from Github. If in update mode
# (see $2), then invokes `svn update', in normal mode invokes
# `svn checkout --non-interactive -q <URL>'. In test mode only
# compares remote and local revision and outputs true if update
# is needed.
#
# $1 - URL
# $2 - mode, "" - normal, "-u" - update, "-t" - test
# $3 - subdirectory (not path) with working copy, needed for -t and -u
```

Has 27 line(s). Doesn't call other functions.

Called by:

```
-zplg-download-snippet
```

-zplg-setup-plugin-dir

```
# Clones given plugin into PLUGIN_DIR. Supports multiple
# sites (respecting `from' and `proto' ice modifiers).
# Invokes compilation of plugin's main file.
#
# $1 - user
# $2 - plugin
```

Has 124 line(s). Calls functions:

```
-zplg-setup-plugin-dir
|-- -zplg-compile-plugin
|   |-- zplugin-side.zsh/-zplg-first
|   `-- zplugin.zsh/-zplg-any-to-user-plugin
|-- -zplg-download-file-stdout
|-- -zplg-handle-binary-file
|-- -zplg-install-completions
|   |-- -zplg-forget-completion
|   |-- zplugin-side.zsh/-zplg-any-colorify-as-uspl2
|   |-- zplugin-side.zsh/-zplg-exists-physically-message
|   `-- zplugin.zsh/-zplg-any-to-user-plugin
|-- zplugin-side.zsh/-zplg-any-colorify-as-uspl2
`-- zplugin-side.zsh/-zplg-store-ices
```

Uses feature(s): *eval*

Called by:

```
zplugin-autoload.zsh/-zplg-update-or-status
zplugin.zsh/-zplg-load
```