zinit-install.zsh(1)

NAME

zinit-install.zsh - a shell script

SYNOPSIS

Documentation automatically generated with `zshelldoc'

FUNCTIONS

```
.zinit-compile-plugin
.zinit-compinit
.zinit-download-file-stdout
.zinit-download-snippet
.zinit-forget-completion
.zinit-get-latest-gh-r-url-part
.zinit-get-package
.zinit-get-url-mtime
.zinit-install-completions
.zinit-jq-check
.zinit-json-get-value
.zinit-json-to-array
.zinit-mirror-using-svn
.zinit-setup-plugin-dir
.zinit-update-snippet
ziextract
AUTOLOAD compinit
```

DETAILS

Script Body

Has 632 line(s). No functions are called (may set up e.g. a hook, a Zle widget bound to a key, etc.).

Uses feature(s): *source*

zinit-compile-plugin
~~~~~~

```
FUNCTION: .zinit-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 92 line(s). Calls functions:

```
.zinit-compile-plugin
|-- zinit-side.zsh/.zinit-compute-ice
|-- zinit-side.zsh/.zinit-first
`-- zinit.zsh/+zinit-message
```

Uses feature(s): *eval, setopt, zcompile*

Called by:

```
zinit-compile-plugin-hook
zinit-autoload.zsh/.zinit-compile-uncompile-all
zinit.zsh/zinit
```

zinit-compinit

~~~~

```
FUNCTION: .zinit-compinit [[[
User-exposed `compinit' frontend which first ensures that all
completions managed by Zinit are forgotten by Zshell. After
that it runs normal `compinit', which should more easily detect
Zinit's completions.
```

```
No arguments.
```

Has 21 line(s). Calls functions:

```
.zinit-compinit
|-- compinit
`-- zinit.zsh/+zinit-message
```

Uses feature(s): *autoload, compinit, setopt, unfunction*

Called by:

```
.zinit-install-completions
zinit-autoload.zsh/.zinit-uninstall-completions
zinit-autoload.zsh/.zinit-update-or-status-all
zinit.zsh/.zinit-prepare-home
zinit.zsh/zinit
```

zinit-download-file-stdout

~~~~~~~~

```
FUNCTION: .zinit-download-file-stdout [[[
Downloads file to stdout. Supports following backend commands:
curl, wget, lftp, lynx. Used by snippet loading.
```

Has 57 line(s). Calls functions:

```
.zinit-download-file-stdout
`-- zinit.zsh/+zinit-message
```

Uses feature(s): *setopt, trap, type*

Called by:

```
.zinit-download-snippet
.zinit-get-cygwin-package
.zinit-get-package
.zinit-setup-plugin-dir
```

zinit-download-snippet

~~~~~~~~~~

```
FUNCTION: .zinit-download-snippet [[[
```

Has 383 line(s). Calls functions:

```
.zinit-download-snippet
|-- zinit-side.zsh/.zinit-store-ices
|-- zinit.zsh/+zinit-message
`-- zinit.zsh/is-at-least
```

Uses feature(s): *is-at-least, setopt, trap, zcompile*

Called by:

```
.zinit-update-snippet
zinit.zsh/.zinit-load-snippet
```

zinit-forget-completion

~~~~~~~

```
FUNCTION: .zinit-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 16 line(s). Doesn't call other functions.

Uses feature(s): *setopt, unfunction*

Called by:

```
.zinit-compinit
.zinit-install-completions
zinit-autoload.zsh/.zinit-uninstall-completions
zinit.zsh/zinit
```

zinit-get-latest-gh-r-url-part

~~~~~~~~~~~

```
FUNCTION: .zinit-get-latest-gh-r-url-part [[[
```

Has 109 line(s). Calls functions:

```
.zinit-get-latest-gh-r-url-part
`-- zinit.zsh/+zinit-message
```

Uses feature(s): *setopt*

Called by:

```
.zinit-setup-plugin-dir
zinit-autoload.zsh/.zinit-update-or-status
```

zinit-get-package

~~~~~~

```
FUNCTION: .zinit-get-package [[[
```

Has 160 line(s). Calls functions:

```
.zinit-get-package
|-- ziextract
|   `-- zinit.zsh/+zinit-message
|-- zinit.zsh/+zinit-message
`-- zinit.zsh/@zinit-substitute
```

Uses feature(s): *eval, setopt, trap*

Called by:

```
zinit.zsh/.zinit-load
```

Environment variables used: zinit.zsh → ZPFX

zinit-get-url-mtime

~~~~~~~~~

```
FUNCTION: .zinit-get-url-mtime [[[
For the given URL returns the date in the Last-Modified
header as a time stamp
```

Has 31 line(s). Doesn't call other functions.

Uses feature(s): *read, setopt, trap, type*

Called by:

```
.zinit-download-snippet
```

zinit-install-completions

~~~~~~~~~~~

```
FUNCTION: .zinit-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 53 line(s). Calls functions:

```
.zinit-install-completions
|-- zinit-side.zsh/.zinit-any-colorify-as-uspl2
|-- zinit-side.zsh/.zinit-exists-physically-message
|-- zinit.zsh/+zinit-message
`-- zinit.zsh/.zinit-any-to-user-plugin
```

Uses feature(s): *setopt*

Called by:

```
.zinit-download-snippet
.zinit-setup-plugin-dir
zinit.zsh/zinit
```

zinit-jq-check

~~~~

```
FUNCTION: .zinit-jq-check [[[
Check if jq is available and outputs an error message with instructions if
that's not the case
```

Has 3 line(s). Calls functions:

```
.zinit-jq-check
`-- zinit.zsh/+zinit-message
```

Called by:

```
.zinit-get-package
.zinit-json-get-value
.zinit-json-to-array
```

zinit-json-get-value

~~~~~~

```
FUNCTION: .zinit-json-get-value [[[
Wrapper around jq that return the value of a property
$1: JSON structure
$2: jq path
```

Has 3 line(s). Calls functions:

```
.zinit-json-get-value
```

Not called by script or any function (may be e.g. a hook, a Zle widget, etc.).

zinit-json-to-array

~~~~~~~~

```
FUNCTION: .zinit-json-to-array [[[
Wrapper around jq that sets key/values of an associative array, replicating
the structure of a given JSON object
$1: JSON structure
$2: jq path
$3: name of the associative array to store the key/value pairs in
```

Has 7 line(s). Calls functions:

```
.zinit-json-to-array
```

Uses feature(s): *eval, setopt*

Called by:

```
.zinit-get-package
```

zinit-mirror-using-svn

~~~~~~~~~

```
FUNCTION: .zinit-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 34 line(s). Doesn't call other functions.

Uses feature(s): *setopt*

Called by:

```
.zinit-download-snippet
```

zinit-setup-plugin-dir

~~~~~~~~~~

```
FUNCTION: .zinit-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 194 line(s). Calls functions:

```
.zinit-setup-plugin-dir
|-- ziextract
|   `-- zinit.zsh/+zinit-message
|-- zinit-side.zsh/.zinit-any-colorify-as-uspl2
|-- zinit-side.zsh/.zinit-store-ices
|-- zinit.zsh/+zinit-message
`-- zinit.zsh/.zinit-get-object-path
```

Uses feature(s): *setopt*, *trap*

Called by:

```
zinit-autoload.zsh/.zinit-update-or-status
zinit.zsh/.zinit-load
```

zinit-update-snippet

~~~~~~

```
   FUNCTION: .zinit-update-snippet [[[
```

Has 71 line(s). Calls functions:

```
.zinit-update-snippet
|-- zinit.zsh/+zinit-message
|-- zinit.zsh/.zinit-get-object-path
`-- zinit.zsh/.zinit-pack-ice
```

Uses feature(s): *eval, setopt*

Called by:

```
zinit-autoload.zsh/.zinit-update-or-status-snippet
```

ziextract

```
   FUNCTION: ziextract [[[
   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 107 line(s). Calls functions:

```
ziextract
`-- zinit.zsh/+zinit-message
```

Uses feature(s): *setopt, zparseopts*

Called by:

```
.zinit-extract
.zinit-get-package
.zinit-setup-plugin-dir
zpextract
```

compinit

Initialisation for new style completion. This mainly contains some helper
functions and setup. Everything else is split into different files that
will automatically be made autoloaded (see the end of this file). The
names of the files that will be considered for autoloading are those that
begin with an underscores (like `_condition).

The first line of each of these files is read and must indicate what
should be done with its contents:

`#compdef <names ...>'

Has 573 line(s). Doesn't call other functions.

Uses feature(s): *setopt*

Called by:

```
.zinit-compinit
```