

# Package: wither (via r-universe)

October 14, 2024

**Title** Temporarily Change Project Root

**Version** 0.1.0.9000

**Description** Lets you temporarily execute an expression or a local block with a different here() root in the 'here' package. This is useful for sourcing code in other projects which expect the root directory of here() to be the project directory of those projects. This may be the case with git submodules for example.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Suggests** devtools, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Imports** fs, here, withr

**URL** <https://github.com/torbjorn/wither>,  
<https://torbjorn.github.io/wither/>

**BugReports** <https://github.com/torbjorn/wither/issues>

**Repository** <https://torbjorn.r-universe.dev>

**RemoteUrl** <https://github.com/torbjorn/wither>

**RemoteRef** HEAD

**RemoteSha** ba6669633b3f0c061baa126c9e3457521f213e91

## Contents

with_here . . . . .	2
<b>Index</b>	<b>4</b>

---

`with_here`*Temporarily Change Project Root*

---

### Description

Temporarily change the `here()` root

### Usage

```
with_here(new_here, expr, chdir = FALSE, verbose = FALSE)
```

```
local_here(  
  new_here,  
  chdir = FALSE,  
  verbose = FALSE,  
  .local_envir = parent.frame()  
)
```

### Arguments

<code>new_here</code>	new temporary here root directory
<code>expr</code>	expression to evaluate
<code>chdir</code>	also temporarily change working directory
<code>verbose</code>	show here's messages on setting new root
<code>.local_envir</code>	the environment to use for scoping, see <a href="#">withr::local_dir()</a>

### Details

Changes [here::here\(\)](#) to temporarily point to a new directory. Automatically changes back to the original value when finished.

The `with_*` and `local_*` flavours of this functionality mimics that which is typically used in the `withr` package.

### Value

[with\\_here\(\)](#) returns the result of the expression. [local\\_here\(\)](#) returns the original value of `here()`, before the change.

### Author(s)

Torbjørn Lindahl

**Examples**

```
library(here)
library(withr)

d <- local_tempdir()

cat("here() is initially:", here(), "\n")

# temporarily do things under a different here() root:
with_here(d, cat("but here() is now:", here(), "\n"))

# check that everything is shifted back
cat("here() is now again:", here(), "\n")

local({

  d <- local_tempdir()

  cat("here was initially: ", here(), "\n")

  local_here(d)

  cat("after local_here(), here() is: ", here(), "\n")
  stopifnot(normalizePath(d) == normalizePath(here()))

  # do something that requires here() be elsewhere

})

cat("outside the block, here() is again:", here(), "\n")
```

# Index

`here::here()`, 2

`local_here (with_here)`, 2

`local_here()`, 2

`with_here`, 2

`with_here()`, 2

`withr::local_dir()`, 2