Package 'NoSleepR'

November 24, 2025

Title Prevent System Sleep During Long R Tasks

Version 0.2.0

Description
Provides a cross-platform interface to prevent the operating system from going to sleep while long-running R tasks are executing.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

BugReports https://github.com/hetalang/NoSleepR/issues

URL https://github.com/hetalang/NoSleepR

Depends R (>= 3.5.0)

NeedsCompilation yes

Author Evgeny Metelkin [aut, cre]

Maintainer Evgeny Metelkin <evgeny.metelkin@gmail.com>

Repository CRAN

Date/Publication 2025-11-24 09:40:12 UTC

Contents

NoSleepR   *NoSleepR: Prevent system sleep during long R tasks*

Description

NoSleepR exposes a tiny, cross-platform API that temporarily disables system sleep while your R script performs a long-running operation. The package delegates to the native inhibition mechanisms shipped with each platform (Win32 power requests, `caffeinate`, or `systemd-inhibit`) and automatically tears them down once you are done.

Details

Core helpers

- `nosleep_on()` — establish a sleep-prevention request and keep the handle alive for as long as the work runs.
- `nosleep_off()` — release a specific handle or clear all active ones when called without arguments.
- `with_nosleep()` — wrap a code block so that NoSleepR turns itself on before the block executes and reliably shuts down after it completes or errors.

All helpers accept the optional `keep_display` flag, allowing you to request that the monitor stays on (when supported by the OS) in addition to the system-wide sleep prevention.

Typical workflow

1. Call `nosleep_on()` (optionally with `keep_display = TRUE`) right before a long computation or data transfer.
2. Run the expensive task.
3. Explicitly stop the request with `nosleep_off()` as soon as the work finishes, or rely on `with_nosleep()` to bracket the code block.

NoSleepR automatically cleans up pending requests when the R session ends, but it is still best practice to explicitly call `nosleep_off()` so that the operating system can resume managing power immediately after the protected job completes.

Author(s)

Maintainer: Evgeny Metelkin <evgeny.metelkin@gmail.com>

See Also

https://github.com/hetalang/NoSleepR

---

nosleep_off *Turn nosleep off*

Description

Turn off a specific nosleep request or, if no handle is supplied, every active request opened by the current R session.

Usage

```
nosleep_off(handle)
```

Arguments

handle   Optional "NoSleepR_handle" object returned by nosleep_on(). If omitted,
                all active nosleep handles created in this session are turned off. Passing NULL is
                treated as a no-op.

Value

Invisibly returns NULL.

Examples

```
## Not run:
h <- nosleep_on()
# ... do work ...
nosleep_off(h)

# Equivalent shortcut to clear everything
nosleep_on()
nosleep_on()
nosleep_off()

## End(Not run)
```

---

nosleep_on *Turn nosleep on*

Description

Prevent the operating system from suspending or putting the display to sleep while long-running R work is executing.

Usage

```
nosleep_on(keep_display = FALSE)
```

Arguments

keep_display   logical. If TRUE, also prevent the display from powering off (when supported by the underlying OS). Default is FALSE.

Details

The returned handle must stay alive for as long as you want the nosleep request to remain in effect. Call nosleep_off() with the handle to release the underlying system resource as soon as the protected work is complete.

If no backend is available for the current platform (e.g., missing dependencies), a warning is issued and invisible NULL is returned.

Value

An object of class "NoSleepR_handle" that stores the active nosleep request for the current platform. Invisible NULL is returned when the request could not be established.

Examples

```
# Simple usage
## Not run:
nosleep_on()
long_running_job()
nosleep_off()

# Handle-based usage
h <- nosleep_on()
long_running_job()
nosleep_off(h)

# Keep the display awake as well (when supported)
h <- nosleep_on(keep_display = TRUE)
Sys.sleep(100)  # simulate long job
nosleep_off(h)

## End(Not run)
```

---

  with_nosleep   *Execute an expression while preventing the system from sleeping*

Description

Helper that automatically brackets an expression with nosleep_on() and nosleep_off().

Usage

```
with_nosleep(expr, keep_display = FALSE)
```

Arguments

expr   Expression to execute while nosleep is on.

keep_display   logical. If TRUE, also prevent the display from sleeping.

Value

The result of evaluating expr.

Examples

```
## Not run:
with_nosleep({
  message("Downloading a large file...")
  download_large_file()
})

## End(Not run)
```

Index