Package ‘semaphore’

November 13, 2024

Type Package

Title Shared Memory Atomic Operations

Version 1.0.2

Date 2024-11-12

Description Implements named semaphores from the 'boost' 'C++' library <https://www.boost.org/>. A semaphore object is shared amongst several processes. This integer value can be safely incremented or decremented by each processes. Processes can also wait (blocking) for the value to become non-zero.

URL https://cmmr.github.io/semaphore/,
https://github.com/cmmr/semaphore

BugReports https://github.com/cmmr/semaphore/issues

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Config/testthat/edition 3

LinkingTo Rcpp, BH

Imports Rcpp

Suggests testthat

NeedsCompilation yes

Author Daniel P. Smith [aut, cre] (<https://orcid.org/0000-0002-2479-2044>),
Alkek Center for Metagenomics and Microbiome Research [cph, fnd]

Maintainer Daniel P. Smith <dansmith01@gmail.com>

Repository CRAN

Date/Publication 2024-11-13 08:30:02 UTC

Contents

create_semaphore *Shared Memory Atomic Operations*

Description

A semaphore is an integer that the operating system keeps track of. Any process that knows the semaphore's identifier can increment or decrement its value, though it cannot be decremented below zero.

When the semaphore is zero, calling `decrement_semaphore(wait = FALSE)` will return `FALSE` whereas `decrement_semaphore(wait = TRUE)` will block until the semaphore is incremented by another process. If multiple processes are blocked, a single call to `increment_semaphore()` will only unblock one of the blocked processes.

Usage

```
create_semaphore(id = NULL, value = 0, cleanup = TRUE)

increment_semaphore(id)

decrement_semaphore(id, wait = TRUE)

remove_semaphore(id)
```

Arguments

`id`	A semaphore identifier (string). `create_semaphore()` defaults to generating a random identifier.
`value`	The initial value of the semaphore.
`cleanup`	Remove the semaphore when R session exits.
`wait`	If `TRUE`, blocks until semaphore is greater than zero.

Value

- `create_semaphore()` - The created semaphore's identifier (string), invisibly when `semaphore` is non-`NULL`.
- `increment_semaphore()` - `TRUE`, invisibly.
- `decrement_semaphore(wait = TRUE)` - `TRUE`, invisibly.
- `decrement_semaphore(wait = FALSE)` - `TRUE` if the decrement was successful; `FALSE` otherwise.
- `remove_semaphore()` - `TRUE` on success; `FALSE` on error.

Examples

```
library(semaphore)

s <- create_semaphore()
print(s)

increment_semaphore(s)
decrement_semaphore(s, wait = FALSE)
decrement_semaphore(s, wait = FALSE)

remove_semaphore(s)
```

Index