Package 'AsioHeaders'

April 15, 2025

Type Package

Title 'Asio' C++ Header Files

Version 1.30.2-1

Date 2025-04-15

Description 'Asio' is a cross-platform C++ library for network and low-level I/O programming that provides developers with a consistent asynchronous model using a modern C++ approach. It is also included in Boost but requires linking when used with Boost. Standalone it can be used header-only (provided a recent compiler). 'Asio' is written and maintained by Christopher M. Kohlhoff, and released under the 'Boost Software License', Version 1.0.

Copyright file inst/COPYRIGHTS

License BSL-1.0

URL https://github.com/eddelbuettel/asioheaders, https://dirk.eddelbuettel.com/code/asioheaders.html

BugReports https://github.com/eddelbuettel/asioheaders/issues

NeedsCompilation no

Author Dirk Eddelbuettel [aut, cre] (<https://orcid.org/0000-0001-6419-907X>), Christopher M. Kohlhoff [aut] (Author of Asio)

Maintainer Dirk Eddelbuettel <edd@debian.org>

Repository CRAN

Date/Publication 2025-04-15 10:50:01 UTC

Contents

AsioHeaders-package 2

Index **3**

AsioHeaders-package *The Asio C++ Library for Network and Low-Level I/O Programming*

Description

Asio is a cross-platform C++ library for network and low-level I/O programming that provides developers with a consistent asynchronous model using a modern C++ approach.

Details

Asio is a cross-platform C++ library for network and low-level I/O programming that provides developers with a consistent asynchronous model using a modern C++ approach.

Asio is also included in Boost but requires linking when used with Boost. Standalone it can be used header-only provided a recent-enough compiler.

Bug reports can also be registered at the GitHub issue tracker at https://github.com/eddelbuettel/asioheaders/issues.

Author(s)

Dirk Eddelbuettel

Maintainer: Dirk Eddelbuettel <edd@debian.org>

References

https://think-async.com/Asio/

See Also

The https://github.com/eddelbuettel/rcppasioexample package provides a simple illustration and example of using this package. It can be used to both assert compiler and setup are working correctly, and form a basis to extend work from. Generally speaking, only a 'LinkingTo: AsioHeaders' should is needed, plus on Windows only a very simply link instruction in `src/Makevars.win` adding `PKG_LIBS = -lwsock32 -lws2_32`.

Examples

```
# None
```

Index

∗ **package**
AsioHeaders-package, 2

AsioHeaders *(*AsioHeaders-package*)*, 2
AsioHeaders-package, 2