

# Package ‘mirai.promises’

December 5, 2023

**Type** Package

**Title** Make 'Mirai' 'Promises'

**Version** 0.4.1

**Description** Allows 'mirai' objects encapsulating asynchronous computations, from the 'mirai' package by Gao (2023) <[doi:10.5281/zenodo.7912722](https://doi.org/10.5281/zenodo.7912722)>, to be used interchangeably with 'promise' objects from the 'promises' package by Cheng (2021) <<https://CRAN.R-project.org/package=promises>>. This facilitates their use with packages 'plumber' by Schloerke and Allen (2022) <<https://CRAN.R-project.org/package=plumber>> and 'shiny' by Cheng, Allaire, Sievert, Schloerke, Xie, Allen, McPherson, Dipert and Borges (2022) <<https://CRAN.R-project.org/package=shiny>>.

**License** GPL (>= 3)

**BugReports** <https://github.com/shikokuchuo/mirai.promises/issues>

**URL** <https://shikokuchuo.net/mirai.promises/>,  
<https://github.com/shikokuchuo/mirai.promises/>

**Encoding** UTF-8

**Depends** R (>= 3.5)

**Imports** nanonext (>= 0.10.1), promises (>= 1.1.0)

**Suggests** mirai (>= 0.10.0)

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Charlie Gao [aut, cre] (<<https://orcid.org/0000-0002-0750-061X>>),  
Hibiki AI Limited [cph]

**Maintainer** Charlie Gao <[charlie.gao@shikokuchuo.net](mailto:charlie.gao@shikokuchuo.net)>

**Repository** CRAN

**Date/Publication** 2023-12-05 20:40:02 UTC

## R topics documented:

mirai.promises-package . . . . .	2
as.promise.mirai . . . . .	3
polling . . . . .	4
<b>Index</b>	<b>5</b>

---

mirai.promises-package

*mirai.promises: Make 'Mirai' 'Promises'*

---

## Description

Allows 'mirai' objects encapsulating asynchronous computations, from the **mirai** package by Gao (2023), to be used interchangeably with 'promise' objects from the **promises** package by Cheng (2021). This facilitates their use with packages **plumber** by Schloerke and Allen (2022) and **shiny** by Cheng, Allaire, Sievert, Schloerke, Xie, Allen, McPherson, Dipert and Borges (2022).

## Notes

This package provides the methods `as.promise.mirai` and `as.promise.recvAio` for the S3 generic `as.promise` exported by the 'promises' package.

An auxiliary function `polling` provides the additional option to tune the frequency at which 'mirai' are checked for resolution.

Package authors wishing to use the S3 methods may simply import the function `polling` to make them available.

## Links

**mirai** website: <https://shikokuchuo.net/mirai/>

**nanonext** website: <https://shikokuchuo.net/nanonext/>

NNG website: <https://nng.nanomsg.org/>

## Author(s)

Charlie Gao <[charlie.gao@shikokuchuo.net](mailto:charlie.gao@shikokuchuo.net)> ([ORCID](#))

---

as.promise.mirai      *Make 'Mirai' 'Promise'*

---

## Description

Creates a 'promise' from a 'mirai' or 'recvAio'.

## Usage

```
## S3 method for class 'mirai'  
as.promise(x)  
  
## S3 method for class 'recvAio'  
as.promise(x)
```

## Arguments

x                      an object of class 'mirai' or 'recvAio'.

## Details

This function is an S3 method for the generic [as.promise](#) for class 'mirai' or 'recvAio'.

[polling](#) may be used to customise the frequency with which to poll for promise resolution (defaults to every 100 milliseconds).

## Value

A 'promise' object.

## Examples

```
if (interactive()) {  
  # Only run examples in interactive R sessions  
  
  p <- promises::as.promise(mirai::mirai("example"))  
  print(p)  
  promises::is.promise(p)  
  
}
```

---

`polling`*Set Polling Frequency*

---

**Description**

Set the frequency with which to poll for promise resolution (the default being 100 milliseconds).

**Usage**

```
polling(freq = 100L)
```

**Arguments**

`freq` [default 100L] integer number of milliseconds.

**Value**

Invisible NULL.

**Examples**

```
# set polling frequency to 1s
polling(freq = 1000L)

# reset polling frequency to default 100 ms
polling()
```

# Index

[as.promise](#), [2](#), [3](#)  
[as.promise.mirai](#), [2](#), [3](#)  
[as.promise.recvAio](#), [2](#)  
[as.promise.recvAio \(as.promise.mirai\)](#), [3](#)  
  
[mirai.promises-package](#), [2](#)  
  
[polling](#), [2](#), [3](#), [4](#)