

Package: sakura (via r-universe)

May 13, 2026

Type Package

Title Extension to R Serialization

Version 0.1.0.9000

Description Extends the functionality of R serialization by augmenting the built-in reference hook system. This enhanced implementation allows an integrated single-pass operation that combines R serialization with third-party serialization methods. Facilitates the serialization of even complex R objects, which contain non-system reference objects, such as those accessed via external pointers, to enable their use in parallel and distributed computing.

License MIT + file LICENSE

URL <https://shikokuchuo.net/sakura/>,
<https://github.com/shikokuchuo/sakura/>

BugReports <https://github.com/shikokuchuo/sakura/issues>

Depends R (>= 3.5)

Suggests arrow, testthat (>= 3.0.0), torch

Config/build/compilation-database true

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2.9000

Repository <https://rconsortium.r-universe.dev>

Date/Publication 2025-04-25 11:10:37 UTC

RemoteUrl <https://github.com/rconsortium/sakura>

RemoteRef HEAD

RemoteSha 70860f75747c3201019e9060f38c3d7f22efc4fc

Contents

sakura-package	2
serial_config	2
serialize	3

Index	5
--------------	----------

sakura-package	<i>sakura: Extension to R Serialization</i>
----------------	---

Description

Extends the functionality of R serialization by augmenting the built-in reference hook system. This enhanced implementation allows an integrated single-pass operation that combines R serialization with third-party serialization methods. Facilitates the serialization of even complex R objects, which contain non-system reference objects, such as those accessed via external pointers, to enable their use in parallel and distributed computing.

Author(s)

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

Authors:

- Travers Ching <traversc@gmail.com>

See Also

Useful links:

- <https://shikokuchuo.net/sakura/>
- <https://github.com/shikokuchuo/sakura/>
- Report bugs at <https://github.com/shikokuchuo/sakura/issues>

serial_config	<i>Create Serialization Configuration</i>
---------------	---

Description

Returns a serialization configuration for custom serialization and unserialization of non-system reference objects, using the 'refhook' system of R native serialization. This allows their use across different R sessions.

Usage

```
serial_config(class, sfunc, ufunc)
```

Arguments

class	a character string (or vector) of the class of object custom serialization functions are applied to, e.g. 'ArrowTabular' or c('torch_tensor', 'ArrowTabular').
sfunc	a function (or list of functions) that accepts a reference object inheriting from class and returns a raw vector.
ufunc	a function (or list of functions) that accepts a raw vector and returns a reference object.

Value

A list comprising the configuration. This may be provided to the hook argument of [serialize\(\)](#) and [unserialize\(\)](#).

Examples

```
serial_config("test_class", base::serialize, base::unserialize)

serial_config(
  c("class_one", "class_two"),
  list(base::serialize, base::serialize),
  list(base::unserialize, base::unserialize)
)
```

serialize	<i>Serialize</i>
-----------	------------------

Description

An extension of R native serialization using the 'refhook' system for custom serialization and un-serialization of non-system reference objects.

Usage

```
serialize(x, hook = NULL)

unserialize(x, hook = NULL)
```

Arguments

x	an object.
hook	[default NULL] optionally, a configuration returned by serial_config() .

Value

For `serialize`: a raw vector. For `unserialize`: the unserialized object.

Examples

```
vec <- serialize(data.frame())
vec
unserialize(vec)
```

```
obj <- list(arrow::as_arrow_table(iris), arrow::as_arrow_table(mtcars))
cfg <- serial_config(
  "ArrowTabular",
  arrow::write_to_raw,
  function(x) arrow::read_ipc_stream(x, as_data_frame = FALSE)
)
raw <- serialize(obj, cfg)
unserialize(raw, cfg)
```

```
x <- list(torch::torch_rand(5L), runif(5L))
cfg <- serial_config("torch_tensor", torch::torch_serialize, torch::torch_load)
unserialize(serialize(x, cfg), cfg)
```

```
cfg <- serial_config(
  c("torch_tensor", "ArrowTabular"),
  list(torch::torch_serialize, arrow::write_to_raw),
  list(torch::torch_load, function(x) arrow::read_ipc_stream(x, as_data_frame = FALSE))
)
y <- list(torch::torch_rand(5L), runif(5L), arrow::as_arrow_table(iris))
unserialize(serialize(y, cfg), cfg)
```

Index

sakura (sakura-package), [2](#)
sakura-package, [2](#)
serial_config, [2](#)
serial_config(), [3](#)
serialize, [3](#)
serialize(), [3](#)

unserialize (serialize), [3](#)
unserialize(), [3](#)