Package: zap (via r-universe)

May 29, 2026

Type Package

Title Fast Object Serialization with High Compression

Version 0.1.1.9002

Maintainer Mike Cheng <mikefc@coolbutuseless.com>

URL https://github.com/coolbutuseless/zap

BugReports https://github.com/coolbutuseless/zap/issues

Description Quickly serialize R objects with high comopression using a custom serialization framework. Lossless transformation is performed on atomic types making them easier to compress; this means that compression can be better and faster than built-in methods. This package includes an implementation of the floating-point compression algorithm described in <doi:10.1145/3626717>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.5.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://coolbutuseless.r-universe.dev

Date/Publication 2026-03-17 07:30:07 UTC

RemoteUrl https://github.com/coolbutuseless/zap

RemoteRef HEAD

RemoteSha 0d4101d629c97049a32fc460e551ac03a88f089c

Contents

`zap_count` *Count the uncompressed bytes to serialize this object*

Description

Count the uncompressed bytes to serialize this object

Usage

```
zap_count(x, opts = list(), ...)
```

Arguments

`x`	R object
`opts`	Named list of options. See `zap_opts()`
`...`	other named options to be included in `opts`. See `zap_opts()` for list of valid options.

Value

Integer

Examples

```
zap_count(mtcars)
length(zap_write(mtcars))
```

`zap_opts` *Create options list for writing data*

Description

Create options list for writing data

Usage

```
zap_opts(
  transform,
  verbosity,
  lgl,
  int,
  fct,
  dbl,
  str,
  list,
  lgl_threshold,
  int_threshold,
  fct_threshold,
  dbl_threshold,
  str_threshold,
  dbl_fallback,
  ...
)
```

Arguments

`transform`	Enable transformations? Default: TRUE. Setting to FALSE will disable all transformations.
`verbosity`	Verbosity level. Default: 0 (no text output).
	64 Return a data.frame with information on each SEXP within the object. `start` and `end` values are the position of the object within the *uncompressed* stream
`lgl`	transformation method for logical vectors. Default: 'packed'
	`raw` Raw. No transformation
	`packed` Packed 2 bits per logical value
`int`	transformation method for integer vectors. Default: 'deltaframe'
	`raw` Raw. No transformation
	`zzshuf` Zig-zag encoding, delta and shuffle
	`deltaframe` Delta frame-of-reference coding
`fct`	transformation method for factors vectors. Default: 'packed'
	`raw` Raw. No transformation
	`packed` Packed minimal bits per level
`dbl`	transformation method for doubles (and complex) vectors. Default: 'alp'
	`raw` Raw. No transformation
	`shuffle` Byte shuffle
	`delta_shuffle` Byte shuffle with delta
	`alp` ALP, Adaptive Lossless Floating Point compression
`str`	transformation method for character vectors. Default: 'mega'

`raw` Raw. No transformation

`mega` Concatenate all strings. Length implicitly encoded by null bytes in strings

`list` transformation method for lists (and data.frames). Default: 'raw'

`raw` Raw. All lists written out in-full

`reference` Cache lists and data.frames as they are seen, and if seen again, write out a reference to the prior object rather than writing out in-full.

`int_threshold`, `lgl_threshold`, `fct_threshold`, `dbl_threshold`, `str_threshold`

Below this threshold, no transformation will be done. All default to 0, meaning transformation is always attempted.

`dbl_fallback` if `dbl = 'alp'`, the data is not always conducive to this compression scheme and after probing the data the code can exit early and try a different method. The `dbl_fallback` variable nominates the fallback method if ALP transformation is being attempted, but fails. The options are the same as for the `dbl` argument (excluding option `'alp'`)

`...` expert level options

Value

named list

Examples

```
myopts <- zap_opts(dbl = 'shuffle')
zap_write(seq(1:1000) * 1.5, opts = myopts)
```

---

zap_read *Unserialize R object from raw vector or file*

Description

Unserialize R object from raw vector or file

Usage

```
zap_read(src, opts = list(), ...)
```

Arguments

`src` Serialization source - either a raw vector of filename.

`opts` Named list of options. See `zap_opts()`

`...` other named options to be included in `opts`. See `zap_opts()` for list of valid options.

Value

Unserialized R object

Examples

```
raw_vec <- zap_write(head(mtcars))
head(raw_vec, 50)
length(raw_vec)
zap_read(raw_vec)
```

---

zap_version *Get version of internal transformation code.*

Description

This version number is the same as the version number present in the header of the serialized data.

Usage

```
zap_version()
```

Value

Integer

Examples

```
zap_version()
```

---

zap_write *Serialize R object to raw vector or file*

Description

Serialize R object to raw vector or file

Usage

```
zap_write(
  x,
  dst = NULL,
  compress = Sys.getenv("zap_compress_default"),
  opts = list(),
  ...
)
```

Arguments

x	R object
dst	Serialization destination. Default: NULL means to return the raw vector. If a character string is given it is assumed to be the path to the output file.
compress	compression type. Default: 'zstd' if available, otherwise 'gzip'. This is set in the 'zap_compress_default' environment variable after being detected during package start. Other valid values 'none', 'xz', 'bzip2'. Compression is done using memCompress()
opts	Named list of options. See zap_opts()
...	other named options to be included in opts. See zap_opts() for list of valid options.

Value

IF dst is NULL, then return a raw vector, otherwise data is written to file and nothing is returned.

Examples

```
raw_vec <- zap_write(head(mtcars))
head(raw_vec, 50)
length(raw_vec)
zap_read(raw_vec)
```

Index