Package ‘yaml12’

December 11, 2025

Title Fast 'YAML' 1.2 Parser and Formatter

Version 0.1.0

Description A fast, correct, safe, and ergonomic 'YAML' 1.2 parser and generator written in 'Rust'. Convert between 'YAML' and simple 'R' objects with full support for multi-document streams, tags, anchors, and aliases. Offers opt-in handlers for custom tag behavior and round-trips common 'R' data structures. Implements the 'YAML' 1.2.2 specification from the 'YAML' Language Development Team (2021) <`https://yaml.org/spec/1.2.2/`>. Proudly supported by Posit.

License MIT + file LICENSE

URL `https://posit-dev.github.io/r-yaml12/`,
`https://github.com/posit-dev/r-yaml12`

BugReports `https://github.com/posit-dev/r-yaml12/issues`

Depends R (>= 4.2)

Suggests jsonlite, knitr, rmarkdown, testthat (>= 3.0.0), waldo, withr

VignetteBuilder knitr

Config/Needs/website tidyverse/tidytemplate

Config/rextendr/version 0.4.2.9000

Config/testthat/edition 3

Config/testthat/parallel false

Encoding UTF-8

RoxygenNote 7.3.3

SystemRequirements Cargo (Rust's package manager), rustc >= 1.70.0, xz

NeedsCompilation yes

Author Tomasz Kalinowski [aut, cre],
Posit Software, PBC [cph, fnd] (ROR: <`https://ror.org/03wc8by49`>),
Authors of the dependency Rust crates [cph] (See inst/AUTHORS and LICENSE.note for vendored Rust dependency authors and licenses.)

Maintainer Tomasz Kalinowski <`tomasz@posit.co`>

Repository CRAN

Date/Publication 2025-12-11 15:50:02 UTC

Contents

---

format_yaml *Format or write R objects as YAML 1.2.*

Description

`format_yaml()` returns YAML as a character string. `write_yaml()` writes a YAML stream to a file or stdout and always emits document start (`---`) markers and a final end (`...`) marker. Both functions honor a `yaml_tag` attribute on values (see examples).

Usage

```
format_yaml(value, multi = FALSE)

write_yaml(value, path = NULL, multi = FALSE)
```

Arguments

`value`	Any R object composed of lists, atomic vectors, and scalars.
`multi`	When `TRUE`, treat `value` as a list of YAML documents and encode a stream.
`path`	Scalar string file path to write YAML to when using `write_yaml()`. When `NULL` (the default), write to R's standard output connection.

Value

`format_yaml()` returns a scalar character string containing YAML. `write_yaml()` invisibly returns `value`.

Examples

```
cat(format_yaml(list(foo = 1, bar = list(TRUE, NA))))

docs <- list("first", "second")
cat(format_yaml(docs, multi = TRUE))

tagged <- structure("1 + 1", yaml_tag = "!expr")
cat(tagged_yaml <- format_yaml(tagged), "\n")

dput(parse_yaml(tagged_yaml))


write_yaml(list(foo = 1, bar = list(2, "baz")))
write_yaml(list("foo", "bar"), multi = TRUE)

tagged <- structure("1 + 1", yaml_tag = "!expr")
write_yaml(tagged)
```

parse_yaml *Parse YAML 1.2 document(s) into base R structures.*

Description

parse_yaml() takes strings of YAML; read_yaml() reads from a file path.

Usage

```
parse_yaml(text, multi = FALSE, simplify = TRUE, handlers = NULL)

read_yaml(path, multi = FALSE, simplify = TRUE, handlers = NULL)
```

Arguments

text Character vector; elements are concatenated with "\n".

multi When TRUE, return a list containing all documents in the stream.

simplify When FALSE, keep YAML sequences as R lists instead of simplifying to atomic vectors.

handlers Named list of R functions with names corresponding to YAML tags; matching handlers transform tagged values.

path Scalar string path to a YAML file‘.

Details

YAML tags without a corresponding handler are preserved in a yaml_tag attribute. Mappings with keys that are not all simple scalar strings are returned as a named list with a yaml_keys attribute.

Value

When multi = FALSE, returns a parsed R object for the first document. When multi = TRUE, returns a list of parsed documents.

Examples

```
dput(parse_yaml("foo: [1, 2, 3]"))

# homogeneous sequences simplify by default.
# YAML null maps to NA in otherwise homogeneous sequences.
dput(parse_yaml("foo: [1, 2, 3, null]"))

# mixed type sequence never simplify
```

```
dput(parse_yaml(”[1, true, cat]”))

# use `simplify=FALSE` to always return sequences as lists.
str(parse_yaml(”foo: [1, 2, 3, null]”, simplify = FALSE))

# Parse multiple documents when requested.
stream <- ”
---
first: 1
---
second: 2
”
str(parse_yaml(stream, multi = TRUE))

# Read from a file; keep sequences as lists.
path <- tempfile(fileext = ”.yaml”)
writeLines(”alpha: [true, null]\nbeta: 3.5”, path)
str(read_yaml(path, simplify = FALSE))
```

Index