

# Package: rconf (via r-universe)

March 3, 2025

**Title** Minimal and Lightweight Configuration Tool

**Version** 0.1.2

**Description** Minimal and lightweight configuration tool that provides basic support for 'YAML' configuration files without requiring additional package dependencies. It offers a simple method for loading and parsing configuration settings, making it ideal for quick prototypes and lightweight projects.

**License** GPL-3

**Encoding** UTF-8

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**RoxygenNote** 7.3.2

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Yaoxiang Li [cre, aut]

**Maintainer** Yaoxiang Li <liyaoxiang@outlook.com>

**Repository** CRAN

**Date/Publication** 2025-03-03 12:10:03 UTC

## Contents

get_config . . . . .	2
merge_configs . . . . .	3
<b>Index</b>	<b>4</b>

---

`get_config`*Get Configuration*

---

**Description**

Reads a YAML configuration file (default "config.yml") and returns the configuration stored under the specified key (default "default"). This function merges the functionality of loading the file and parsing its contents into a list. It uses UTF-8 encoding to read the file and then processes it with a minimal YAML parser.

**Usage**

```
get_config(file = "config.yml", config_name = "default")
```

**Arguments**

<code>file</code>	Path to the YAML configuration file. For example, to load the sample configuration provided in the package <code>extdata</code> directory, use: <code>'system.file("extdata", "config.yml", package = "rconf")'</code> .
<code>config_name</code>	The key in the configuration to extract (e.g., "default"). If the key exists in the parsed configuration list, only that sub-list is returned.

**Details**

If the parsed configuration list contains a component with the name specified by `'config_name'`, only that component is returned. Otherwise, the entire configuration list is returned.

**Value**

A list of configuration settings. When the configuration file contains multiple sections, and the section matching `'config_name'` is found, only that section is returned. Otherwise, the full configuration list is returned.

**Author(s)**

Yaoxiang Li

**Examples**

```
# Example: Load sample configuration from the extdata directory.
cfg <- get_config(system.file("extdata", "config.yml", package = "rconf"),
                 config_name = "default")
# Access specific configuration settings:
print(cfg$raw_data_dir)
print(cfg$processed_data_dir)
```

---

merge_configs	<i>Merge Configurations</i>
---------------	-----------------------------

---

**Description**

Recursively merges two configuration lists, with values in the override replacing those in the base configuration.

**Usage**

```
merge_configs(base_cfg, override_cfg)
```

**Arguments**

`base_cfg`        A list containing the base configuration.  
`override_cfg`    A list containing override values.

**Value**

A merged configuration list.

**Examples**

```
base_cfg <- get_config(system.file("extdata",  
  "config.yml", package = "rconf"), config_name = "default")  
dev_cfg <- get_config(system.file("extdata",  
  "config.yml", package = "rconf"), config_name = "development")  
combined_cfg <- merge_configs(base_cfg, dev_cfg)  
print(combined_cfg)
```

# Index

`get_config`, 2

`merge_configs`, 3