

# Package: RSBJson (via r-universe)

September 18, 2024

**Type** Package

**Title** Handle R Requests from R Service Bus Applications with JSON Payloads

**Version** 1.1.2

**Date** 2020-06-02

**Author** Tobias Verbeke

**Maintainer** Tobias Verbeke <tobias.verbeke@openanalytics.eu>

**URL** <https://www.rservicebus.io>

**Description** Package to Handle R Requests from R Service Bus Applications with JSON Payloads in a generic way. The incoming request is encoded as a string (character vector of length one) containing the JSON file passed through by the client.

**Imports** jsonlite

**License** GPL-3

**Encoding** UTF-8

**Collate** 'allPossibleCombinations.R' 'RSBJsonService.R'

**RoxygenNote** 7.1.0

**NeedsCompilation** no

**Date/Publication** 2020-06-16 11:00:12 UTC

**Repository** <https://tverbeke.r-universe.dev>

**RemoteUrl** <https://github.com/cran/RSBJson>

**RemoteRef** HEAD

**RemoteSha** 85c7e9b5cec187d4d586c678a0ec486c0cdacefb

## Contents

allPossibleCombinations . . . . .	2
RSBJsonService . . . . .	2

<b>Index</b>	<b>4</b>
--------------	----------

allPossibleCombinations

*Generate All Possible Combinations of m Out of n*

---

### Description

Generate All Possible Combinations of m Out of n

### Usage

```
allPossibleCombinations(n, m)
```

### Arguments

n	n
m	m

### Value

a JSON string with all possible combinations; object of class `json` as returned by [toJSON](#)

### Examples

```
allPossibleCombinations(4, 2)
```

---

RSBJsonService

*Generic Function to Handle R Requests from RSB JSON Applications*

---

### Description

The function extracts relevant information from the JSON string to dispatch to the appropriate R wrapper for that particular application

### Usage

```
RSBJsonService(jsonString)
```

### Arguments

jsonString	character vector of length one containing the JSON string
------------	---

### Value

object of class `json` as returned by [toJSON](#)

**Examples**

```
examplePath <- system.file("examples", "example.json", package = "RSBJson")
exampleCon <- file(description = examplePath, open = "rt")
exampleLines <- readLines(exampleCon)
close(exampleCon)
exampleString <- paste(exampleLines, collapse = "\n")
(string <- RSBJsonService(jsonString = exampleString))
```

# Index

`allPossibleCombinations`, [2](#)

`RSBJsonService`, [2](#)

`toJSON`, [2](#)