

# Package: msgpackR (via r-universe)

August 31, 2024

**Type** Package

**Title** A library to serialize or unserialize data in MessagePack format

**Version** 1.1

**Date** 2013-11-21

**Author** Mikiya TANIZAWA

**Maintainer** Mikiya TANIZAWA <1970mix@gmail.com>

**Description** This is the library that can serialize or unserialize MessagePack format data.

**License** BSD\_2\_clause + file LICENSE

**NeedsCompilation** no

**Date/Publication** 2013-11-22 07:33:57

**Repository** <https://mikiya0417.r-universe.dev>

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

**RemoteRef** HEAD

**RemoteSha** 83a90eed65e33a773128fc2e2f4e93a774bdc465

## Contents

msgpackR-package . . . . .	2
msgpack.matrix . . . . .	3
msgpack.writeResult . . . . .	5
pack . . . . .	5
unpack . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

---

msgpackR-package

*A library to serialize or deserialize data in MessagePack format*

---

## Description

This is the library that can serialize or deserialize MessagePack format data.

## Details

Package: MessagePack for R  
Type: Package  
Version: 1.1  
Date: 2013-11-21  
License: BSD\_2\_clause + file LICENSE

## Author(s)

Mikiya TANIZAWA <1970mix@gmail.com>

## References

<http://msgpack.org/>

## See Also

[pack](#), [unpack](#), [msgpack.writeResult](#), [msgpack.matrix](#)

## Examples

```
(data <- c(1,2,3))
#[1] 1 2 3
(d <- pack(data))
#[1] 93 01 02 03
msgpack.writeResult("test.txt", d)
unpack("test.txt") # <= unserialize from binary file
#[1] 1 2 3
unpack(d) # <= unserialize from binary bits
#[1] 1 2 3

# example to serialize {"compact":true}, which appears at http://msgpack.org/.
sample <- TRUE
names(sample) <- c("compact")
sample
#compact
# TRUE
```

```
pack(sample)
#[1] 81 a7 63 6f 6d 70 61 63 74 c3

# Positive FixNum
a <- 100
pack(a)
#[1] 64
unpack(pack(a))
#[1] 100

# uint16
a <- 2^10
pack(a)
#[1] cd 04 00
unpack(pack(a))
#[1] 1024

# int16
a <- -2^10
pack(a)
#[1] d1 fc 00
unpack(pack(a))
#[1] -1024

# double
a <- 10.1
pack(a)
#[1] cb 40 24 33 33 33 33 33
unpack(pack(a))
#[1] 10.1

# FixRaw
a <- "sample character"
pack(a)
#[1] b0 73 61 6d 70 6c 65 20 63 68 61 72 61 63 74 65 72
unpack(pack(a))
#[1] "sample character"
```

---

msgpack.matrix

*transfer list format to matrix format*

---

## Description

Transfer list format to matrix format

## Usage

```
msgpack.matrix(data)
```

**Arguments**

data                    a list format data that is made from unpack function

**Value**

matrix format data

**Author(s)**

Mikiya TANIZAWA <1970mix@gmail.com>

**See Also**

[unpack](#)

**Examples**

```
(mat <- matrix(1:6, 2))
#      [,1] [,2] [,3]
#[1,]  1   3   5
#[2,]  2   4   6
(m <- pack(mat))
# [1] 92 93 01 03 05 93 02 04 06
unpack(m)
#[[1]]
#[1] 1 3 5
#
#[[2]]
#[1] 2 4 6
msgpack.matrix(unpack(m))
#      [,1] [,2] [,3]
#[1,]  1   3   5
#[2,]  2   4   6

colnames(mat) <- c("A","B","C")
mat
#      A B C
#[1,] 1 3 5
#[2,] 2 4 6
(m <- pack(mat)) # <= if data has colname, serialize to "map"
#[1] 92 83 a1 41 01 a1 42 03 a1 43 05 83 a1 41 02 a1 42 04 a1 43 06
unpack(m)
#[[1]]
#A B C
#1 3 5
#
#[[2]]
#A B C
#2 4 6
msgpack.matrix(unpack(m))
#      A B C
#[1,] 1 3 5
```

```
#[2,] 2 4 6
```

---

`msgpack.writeResult` *the function to save serialized data to file*

---

### **Description**

Save serialized data to file.

### **Usage**

```
msgpack.writeResult(filename, result)
```

### **Arguments**

filename	filename that you want to save as
result	serialized data that you want to save

### **Author(s)**

Mikiya TANIZAWA <1970mix@gmail.com>

### **See Also**

[pack](#)

### **Examples**

```
(data <- c(1,2,3))
#[1] 1 2 3
(d <- pack(data))
#[1] 93 01 02 03
msgpack.writeResult("test.txt", d)
```

---

`pack` *serialize data to MessagePack format*

---

### **Description**

Serialize data to MessagePack format

### **Usage**

```
pack(data)
```

**Arguments**

data                    data that you want to serialize, e.g. c(1,2,3)

**Value**

serialized data to MessagePack format

**Author(s)**

Mikiya TANIZAWA <1970mix@gmail.com>

**See Also**

[unpack](#), [msgpack.writeResult](#)

**Examples**

```
pack(c(1,2,3))
#[1] 93 01 02 03
mat <- matrix(c(1:6), nrow=2); colnames(mat) <- c("A","B","C"); pack(mat);
#[1] 92 83 a1 41 01 a1 42 03 a1 43 05 83 a1 41 02 a1 42 04 a1 43 06
```

---

unpack

*unserialize MessagePack format data*

---

**Description**

Unserialize MessagePack format data

**Usage**

```
unpack(str)
```

**Arguments**

str                    str is filename or array of raw data

**Value**

unserialized data from MessagePack format

**Author(s)**

Mikiya TANIZAWA <1970mix@gmail.com>

**See Also**

[pack](#), [msgpack.writeResult](#)

**Examples**

```
unpack(pack(c(1,2,3)))
#[1] 1 2 3
(data <- c(1,2,3))
#[1] 1 2 3
(d <- pack(data))
#[1] 93 01 02 03
msgpack.writeResult("test.txt", d)
unpack("test.txt") # <= unserialize from binary file
#[1] 1 2 3
unpack(d) # <= unserialize from binary bits
#[1] 1 2 3
```

# Index

## \* **package**

msgpackR-package, 2

msgpack.matrix, 2, 3

msgpack.writeResult, 2, 5, 6

msgpackR (msgpackR-package), 2

msgpackR-package, 2

pack, 2, 5, 5, 6

unpack, 2, 4, 6, 6