

# Ruuid

October 5, 2010

---

Ruuid-class

*Class "Ruuid": A class to manage UUID values*

---

## Description

Class `Ruuid` wraps and handles the UUID values for the user within R.

## Objects from the Class

Objects are not intended to be directly created. A user can retrieve a new instantiation of this class with the `getuuid` function.

## Slots

`stringRep`: Object of class "character" A human readable string representation of the ID  
`uuid`: Object of class "uuidt" Contains the actual UUID value.

## Methods

**stringRep** Retrieves the string representation of the UUID

**uuid** Retrieves the "uuidt" object

**show** Outputs the string representation of the UUID

**as.character** Returns the string representation of the UUID

**==** Returns true if both arguments are `Ruuid` objects and have the same string representation.

**!=** Returns true if both arguments are `Ruuid` objects and have different string representations.

## Author(s)

R. Gentleman and Jeff Gentry

## References

Theodore Ts'o's Universally Unique ID library  
<http://web.mit.edu/tytso/www/home.html>

## See Also

[getuuid](#), [uuidt](#)

## Examples

```
z <- getuuid()
z
class(z)
```

---

getuuid

*Functions to handle unique IDs*

---

## Description

These functions allow a user to retrieve and transform universally unique ID values. A user can get a new UUID, as well as transform a string representation into a byte representation and vice versa.

## Usage

```
getuuid(cur)
```

## Arguments

`cur` If provided, either a UUID string or a `uuidt` object

## Details

This package uses Theodore Ts'o's `libuuid` to generate universally unique ID values. A unique ID value is represented in R by the class `uuidt`, which wraps the byte values. A wrapper object is the most common method of handling these values, `Ruuid` - which also provides for a string representation of the UUID.

If no parameter `cur` is passed into `getuuid`, a new `Ruuid` is generated. If either a string or a `uuidt` object is passed in, then the complementary piece is generated and the resultant `Ruuid` is generated.

## Value

`getuuid`: Returns an object of class `Ruuid`.

## Author(s)

R. Gentleman and Jeff Gentry

## References

Theodore Ts'o's Universally Unique ID library  
<http://web.mit.edu/tytso/www/home.html>

## See Also

[Ruuid](#), [uuidt](#)

**Examples**

```
z <- getuuid()
z
a <- getuuid(as.character(z))
a
q <- getuuid(uuid(z))
q
w <- getuuid()
w
```

---

`uuidt-class`*Class "uuidt": A class to contain UUID values*

---

**Description**

Class `uuidt` wraps the actual byte values for a UUID

**Objects from the Class**

Objects of class `uuidt` are not intended to be created directly by the user. A call to `getuuid` or `str2uuid` will generate such a value.

**Slots**

`uuid`: Object of class "character": Byte values of the UUID

**Methods**

**uuid** Retrieves the byte string of the UUID

**show** Prints a human readable display of the byte values

**Author(s)**

R. Gentleman and Jeff Gentry

**References**

Theodore Ts'o's Universally Unique ID library  
<http://web.mit.edu/tytso/www/home.html>

**See Also**

[getuuid](#), [Ruuid](#)

**Examples**

```
z <- getuuid()
a <- uuid(z)
a
class(a)
```

# Index

!=, Ruuid, Ruuid-method  
    (*Ruuid-class*), 1

**\*Topic classes**

    Ruuid-class, 1  
    uuidt-class, 3

**\*Topic utilities**

    getuuid, 2

==, Ruuid, Ruuid-method  
    (*Ruuid-class*), 1

as.character, Ruuid-method  
    (*Ruuid-class*), 1

getuuid, 1, 2, 3

Ruuid, 2, 3  
Ruuid (*Ruuid-class*), 1  
Ruuid-class, 1

show, Ruuid-method (*Ruuid-class*), 1  
show, uuidt-method (*uuidt-class*), 3  
stringRep (*Ruuid-class*), 1  
stringRep, Ruuid-method  
    (*Ruuid-class*), 1

uuid (*Ruuid-class*), 1  
uuid, Ruuid-method (*Ruuid-class*), 1  
uuid, uuidt-method (*uuidt-class*), 3  
uuidt, 1, 2  
uuidt (*uuidt-class*), 3  
uuidt-class, 3