# Package: zooExtra (via r-universe)

July 22, 2026

**Version** 0.1-1

**Date** 2026-07-22

**Title** Supplementary Functionality for zoo Time Series

**Description** Supplementary functionality that can be used with the zoo package for regular and irregular time series data. The package also serves as a development platform for ideas that might (or might not) eventually end up in the zoo package.

**Depends** R (>= 2.9.0), zoo

**Suggests** chron

**Imports** stats

**License** GPL-2 | GPL-3

**URL** https://zeileis.codeberg.page/zooExtra/

**BugReports** https://codeberg.org/zeileis/zooExtra/issues

**Repository** https://zeileis.r-universe.dev

**Date/Publication** 2026-07-22 12:16:39 UTC

**RemoteUrl** https://codeberg.org/zeileis/zooExtra

**RemoteRef** HEAD

**RemoteSha** f60115d2eb4016b95540768a3b2dd99ab6be78ad

## Contents

as.multitime *A time object with multiple time scales.*

## Description

Multitime objects have a primary time scale and a secondary time scale. The latter can be used to distinguish between duplicates in the primary time scale or provide an alternate representation such as specific dates that correspond to year/months.

## Usage

```
multitime(x, ...)

as.multitime(x, ...)
## Default S3 method:
as.multitime(x, index, ...)
```

## Arguments

| | |
|---|---|
| x | Primary time scale. |
| index | Optional secondary time scale. |
| ... | Other arguments passed to individual methods. |

## Details

Creates an object of class multitime which has a primary and possibly secondary time scale. Methods provided are: as.multitime, as.multitime.default, coredata.multitime, index.multitime, as.character.multitime, MATCH.multitime, ORDER.multitime, xtfrm.multitime, Ops.multitime, [.multitime c.multitime, as.numeric.multitime, as.Date.multitime, as.yearmon.multitime, as.yearqtr.multitime and make.unique.multitime. The primary and secondary time scales when considered in conjunction with each other must be unique if they are to be used as the times of a zoo object yet individually need not be.

as.Date, as.yearmon and as.yearqtr methods search through the time scales and return the first one found of the indicated class or if none found convert the first time scale to the indicated class.

## Value

An object of class "multitime".

## Examples

```
# use secondary time scale to uniquify times
if (require("zoo")) {
z <- zoo(1:3, as.multitime(c(1, 1, 2), c(1, 2, 1)))
z

# an object with both yearmon and Date scales
```

```
ym <- as.yearmon(2001:2003)
z2 <- zoo(1:3, as.multitime(ym, as.Date(ym, frac = 1)))
z2
as.yearmon(time(z2))
as.Date(time(z2))
}
```

---

`make.unique.approx` *Make a vector unique*

---

## Description

Make a vector unique

## Usage

```
make.unique.approx(x, ...)
## Default S3 method:
make.unique.approx(x, quantile = 0:1, ...)
```

## Arguments

| | |
|---|---|
| `x` | Vector. Vector of a class which has differences that can be coerced to numeric for which the difference, when added to the first value, returns the second value. |
| `quantile` | A number between 0 and 1 (or vector of numbers) such that if it is 0 then the first time value among duplicates is kept as is and the others calculated via linear interpolation or if 1 then the last time value is kept as is and the others calculated via linear interpolation. If an intermediate fraction is used then that quantile is kept as is. If a vector of numbers is specified then they are applied in succession. 0 will not remove duplicates at the end and 1 will not remove trailing duplicates at the beginning but `0:1` or `1:0` will remove both. |
| `...` | Further arguments. |

## Value

The function returns `x` except among any set of repeated elements the one at the position whose quantile is indicated is left as is and all others are linearly interpolated.

## Examples

```
xx <- c(10, rep(20, 10), 30)
make.unique.approx(xx)
```

# Index