Package 'bigmemoryExtras'

April 15, 2019

Type Package

Title An extension of the bigmemory package with added safety, convenience, and a factor class

Version 1.30.0

Author Peter M. Haverty

Maintainer Peter M. Haverty <phaverty@gene.com>

Description This package defines a ``BigMatrix'' ReferenceClass which adds safety and convenience features to the filebacked.big.matrix class from the bigmemory package. BigMatrix protects against segfaults by monitoring and gracefully restoring the connection to on-disk data and it also protects against accidental data modification with a filesystem-based permissions system. We provide utilities for using BigMatrix-derived classes as assayData matrices within the Biobase package's eSet family of classes. BigMatrix provides some optimizations related to attaching to, and indexing into, file-backed matrices with dimnames. Additionally, the package provides a ``BigMatrixFactor'' class, a file-backed matrix with factor properties.

License Artistic-2.0

LazyLoad yes

ByteCompile TRUE

Depends R (>= 2.12), bigmemory (>= 4.5.31)

Imports methods

Suggests testthat, BiocGenerics, BiocStyle, knitr

biocViews Infrastructure, DataRepresentation

OS_type unix

VignetteBuilder knitr

URL https://github.com/phaverty/bigmemoryExtras

RoxygenNote 6.0.1

git_url https://git.bioconductor.org/packages/bigmemoryExtras

git_branch RELEASE_3_8

git_last_commit 585a2c0

git_last_commit_date 2018-10-30

Date/Publication 2019-04-15

R topics documented:

bigmemoryExtras-package 2
annotatedDataFrameFrom 2
BigMatrix 3
BigMatrix-class 3
BigMatrixFactor 6
BigMatrixFactor-class 6
updateBackingfiles 8
updateBigMatrix 8

Index **10**

`bigmemoryExtras-package`

An extension of the bigmemory package with added safety, convenience, and a factor class.

Description

This package defines a "BigMatrix" ReferenceClass which adds safety and convenience features to the filebacked.big.matrix class from the bigmemory package. BigMatrix protects against segfaults by monitoring, and gracefully restoring, the connection to on-disk data. It also protects against accidental data modification with a filesystem-based permissions system. We provide functionality for using BigMatrix-derived classes as assayData matrices within the Biobase package's eSet family of classes. BigMatrix provides some optimizations related to attaching to, and indexing into, file-backed matrices with dimnames. Additionally, the package provides a "BigMatrixFactor" class, a file-backed matrix with factor properties.

Details

BigMatrix stores the filesystem path to its on-disk data. When a big.matrix object is saved, e.g. with "save" the external pointer to the on-disk data becomes "nil". Any access to this nil pointer causes a segfault. A BigMatrix object will gracefully re-attach itself to its on-disk components any time use of the "nil" pointer is attempted.

See Also

BigMatrix-class BigMatrixFactor-class BigMatrix BigMatrixFactor filebacked.big.matrix ReferenceClasses

---

`annotatedDataFrameFrom`

Use BigMatrix in eSet assayData

Description

The Biobase package's eSet objects hold one or more matrices within an environment stored in their assayData slot. Use of classes other than a base matrix require a method to create an annotatedDataFrame from that class. Users are unlikely to ever call this method.

BigMatrix *Create a new BigMatrix*

Description

Create a new BigMatrix

Usage

```
BigMatrix(x = NA_real_, backingfile, nrow, ncol,
  dimnames = NULL, type = "double")
```

Arguments

x	scalar numeric, NULL, matrix, or big.matrix. Optional data or big.matrix for new BigMatrix. A scalar numeric can be used to initalize the whole matrix. NULL gives the bigmemory feature of initializing to all zeros instantly.
backingfile	character, full path to the file that will contain the data matrix
nrow	integer, number of rows in the matrix we are about to create
ncol	integer, number of columns in the matrix we are about to create
dimnames	list, list(rownames,colnames), as for a typical matrix
type	character type of big.matrix (double, integer, char)

Value

BigMatrix

Examples

```
dnames = dimnames=list(letters[1:3],LETTERS[1:3])
x = matrix(1:9,ncol=3,dimnames=dnames)
ds = BigMatrix(x,tempfile())
ds = BigMatrix(backingfile=tempfile(),nrow=3,ncol=3,dimnames=dnames)
```

BigMatrix-class *Class "BigMatrix"*

Description

This package defines a "BigMatrix" ReferenceClass which adds safety and convenience features to the filebacked.big.matrix class from the bigmemory package. BigMatrix protects against segfaults by monitoring and gracefully restoring the connection to on-disk data. We provide utilities for using BigMatrix-derived classes as assayData matrices within the Biobase package's eSet family of classes. BigMatrix provides some optimizations related to attaching to, and indexing into, file-backed matrices with dimnames.

A big.matrix object is backed by two files, a data file (backingfile) and a text metadata file (descriptor file). BigMatrix replaces the text metadata file with a binary "rds" file, for faster access. The location of this file is stored in the descpath field. The data in the metadata file is used to "attach" a

big.matrix object its on-disk data. BigMatrix stores the big.matrix object and checks for a valid connection to the on-disk data each time it is necessary to access the big.matrix object. Re-attachment, using the `attach` method and the `descpath` field is performed as necessary. This avoids a segfault caused by accessing what could otherwise be a nil pointer. For example, A nil pointer occurs when a big.matrix data is saved and loaded using `save()` and `load()`.

Reference classes are a relatively new and unused feature of R. They represent a new object model that is quite similar to OOP you might have seen in java or python. The ?ReferenceClasses help page is an excellent resource on this topic. As ReferenceClasses are an extension of R's S4 class system, S4-style methods can also be defined. In order to give the familiar matrix API, BigMatrix's ReferenceClass methods have a corresponding S4-style method. Please see the examples below.

Extends

All reference classes extend and inherit methods from `"envRefClass"`.

Methods

S4-style methods supplement ReferenceClass methods to provide a standard matrix API.

[`signature(x = "BigMatrix", i = "ANY", j = "ANY", drop = "ANY")`: Access matrix data.

[<- `signature(x = "BigMatrix", i = "ANY", j = "ANY")`: Set matrix data.

[<- `signature(x = "BigMatrix", i = "ANY", j = "ANY", NA = "ANY")`: Set matrix data

annotatedDataFrameFrom `signature(object = "BigMatrix")`: Used in creation of a Biobase eSet with a BigMatrix as an assayDataElement.

coerce `signature(from = "BigMatrix", to = "matrix")`: Coerce BigMatrix to a base matrix.

as.matrix `signature(x = "BigMatrix")`: Coerce BigMatrix to a base matrix.

dim `signature(x = "BigMatrix")`: Get dimensions of the matrix.

dimnames `signature(x = "BigMatrix")`: Get dimension names of the matrix.

dimnames<- `signature(x = "BigMatrix", value = "ANY")`: Set dimension names

length `signature(x = "BigMatrix")`: Get length of the vector used as a matrix (nrow * ncol).

ncol `signature(x = "BigMatrix")`: Get the number of columns in the matrix.

nrow `signature(x = "BigMatrix")`: Get the number of rows in the matrix.

apply `signature(X = "BigMatrix")`: Apply a function over one margin of the data in a BigMatrix, borrowing from biganalytics.

Fields

`bigmat`**:** Object of class `activeBindingFunction` Public access to the big.matrix object.

`datapath`**:** Object of class `activeBindingFunction` Full file path the big.matrix backing file.

`descpath`**:** Object of class `character` Full file path the big.matrix metadata file.

`.bm`**:** Object of class `big.matrix` Private storage of the big.matrix object.

Class-Based Methods

`save()`: Saves metadata about big.matrix in .bm field to file at location given by descpath field.

colnames `signature(x = "BigMatrix")`: Get column names of the matrix.

rownames `signature(x = "BigMatrix")`: Get row names of the matrix.

dimnames `signature(x = "BigMatrix")`: Get dimension names of the matrix.

dim `signature(x = "BigMatrix")`: Get the dimensions of the matrix.

nrow `signature(x = "BigMatrix")`: Get the number of rows in the matrix.

ncol `signature(x = "BigMatrix")`: Get the number of columns in the matrix.

length `signature(x = "BigMatrix")`: Get the number of cells in the matrix.

`setValues(i, j, value)`: Setter for on-disk data.

`getValues(i, j, drop)`: Getter for on-disk data.

`attach()`: Uses path from descpath field to connect object to on-disk data. 'force' argument will force re-attachment of an object that is already attached.

Author(s)

Peter M. Haverty

See Also

See Also `ReferenceClasses`, `BigMatrix`, `BigMatrixFactor`, `makeActiveBinding`,`filebacked.big.matrix`

Examples

```
showClass("BigMatrix")
back.file = tempfile()
x = matrix(1:9,ncol=3,dimnames=list(letters[1:3],LETTERS[1:3]))
ds = BigMatrix(x,back.file)
ds[1,3]
ds[3,3] = 5
as(ds,"matrix")
ds[,]
rownames(ds)
ds$rownames()
colnames(ds)
ds$colnames()
dimnames(ds)
ds$dimnames()
dim(ds)
ds$dim()
nrow(ds)
ds$nrow()
ncol(ds)
ds$ncol()
```

BigMatrixFactor *Create a new BigMatrixFactor*

Description

Create a new BigMatrixFactor

Usage

```
BigMatrixFactor(x = NA_character_, backingfile, nrow,
  ncol, dimnames = NULL, levels)
```

Arguments

x	scalar or matrix to be treated as character.
backingfile	character, full path to the file that will contain the data matrix
nrow	integer, number of rows in the matrix we are about to create
ncol	integer, number of columns in the matrix we are about to create
dimnames	list, list(rownames,colnames), as for a typical matrix
levels	character, as for a typical factor

Value

BigMatrixFactor

Examples

```
dnames = dimnames=list(letters[1:3],LETTERS[1:3])
levels=c("AA","AB","BB")
x = matrix( sample( levels, 9, replace=TRUE), ncol=3, dimnames=dnames)
ds = BigMatrixFactor(x,tempfile(),levels=levels)
ds = BigMatrixFactor(backingfile=tempfile(),nrow=3,ncol=3,dimnames=dnames,levels=levels)
```

BigMatrixFactor-class *Class* "BigMatrixFactor"

Description

The BigMatrixFactor class extends the BigMatrix class to add factor functionality. Data are stored as 32-bit or 8-bit integers on-disk depending on the number of levels. Reference Class methods are also provided in S4 style to give the familiar matrix and factor API (please see the examples below). Subsetting returns a integer matrix with the S3 class "factor" and an appropriate levels attribute.

Extends

Class "BigMatrix", directly.

All reference classes extend and inherit methods from "envRefClass".

Methods

S4-style methods supplement ReferenceClass methods to provide a standard factor API.

`signature(x = "BigMatrixFactor")`: Query the number of levels, as in a typical factor.

nlevels `signature(x = "BigMatrixFactor")`: Access the levels, as in a typical factor.

levels<- `signature(x = "BigMatrixFactor")`: Set the levels, as in a typical factor. Disallowed for now.

Fields

`bigmat`: Object of class `activeBindingFunction` Public access to the big.matrix object.

`datapath`: Object of class `activeBindingFunction` Full file path to the big.matrix backing file.

`descpath`: Object of class `character` Full file path to the big.matrix metadata file.

`.bm`: Object of class `big.matrix` Private storage of the big.matrix object.

`levels`: Object of class `activeBindingFunction` Public access to the levels.

`.levels`: Object of class `character` Levels as in a typical factor.

Class-Based Methods

`nlevels()`: Check number of levels, as in a typical factor.

`getValues(i, j, drop)`: Getter for on-disk data. Returns a factor with the appropriate number of dimensions.

`setValues(i, j, value)`: Setter for on-disk data. Data converted to integer, matching to levels for character input.

The following methods are inherited (from the corresponding class): setValues ("BigMatrix"), getValues ("BigMatrix"), attach ("BigMatrix"), save ("BigMatrix")

Author(s)

Peter M. Haverty

See Also

See Also `ReferenceClasses`, `BigMatrix`, `factor`, `BigMatrix`, `makeActiveBinding`,`filebacked.big.matrix`

Examples

```
showClass("BigMatrixFactor")
back.file = tempfile()
x = matrix( sample( 1:3, 9, replace=TRUE), ncol=3, dimnames=list(letters[1:3],LETTERS[1:3]))
ds = BigMatrixFactor(x,back.file,levels=c("AA","AB","BB"))
ds[1,3]
ds[3,3] = "AB"
ds[,]
nlevels(ds)
ds$nlevels()
levels(ds)
ds$levels
```

updateBackingfiles *Update directory for BigMatrix objects in a collection to new location*

Description

Update directory for BigMatrix objects in a collection to new location. Assumes files have already been moved on the filesystem. Assumes names of description and data files are the same. The collection can contain a mix of BigMatrix objects and other types. The other types will be not be touched and will be returned as they are.

Usage

```
updateBackingfiles(x, dir)
```

Arguments

x	list, SimpleList, environment or something with names and [[methods
dir	character, path to directory holding all BigMatrix files

Details

If you have renamed specific backing files, you will want to update the backingfile field of the relevant BigMatrix objects.

Value

x param, with modified BigMatrix objects.

Examples

```
## Not run:  list = updateBackingfiles(list, "/new/path/to/bigmat/dir")
## Not run:  assays(se) = updateBackingfiles(assays(se), "/new/path/to/bigmat/dir")
## Not run:  assayData(eset) = updateBackingfiles(assayData(eset), "/new/path/to/bigmat/dir")
```

updateBigMatrix *Update previous BigMatrix type to new BigMatrix*

Description

BigMatrix has changed some of its internal storage to eliminate the descriptor file and to keep the dimnames on the R side. This function will take a Version <= 1.3 type and update it to the Version >=1.4 type.

Usage

```
updateBigMatrix(object)
```

Arguments

object	BigMatrix

Value

BigMatrix

Index

∗Topic **classes**
BigMatrix-class, 3
BigMatrixFactor-class, 6
[,BigMatrix,ANY,ANY,ANY-method *(BigMatrix-class)*, 3
[<-,BigMatrix,ANY,ANY,ANY-method *(BigMatrix-class)*, 3
[<-,BigMatrix,ANY,ANY-method *(BigMatrix-class)*, 3

annotatedDataFrameFrom, 2
annotatedDataFrameFrom,BigMatrix-method *(BigMatrix-class)*, 3
apply,BigMatrix-method *(BigMatrix-class)*, 3
as.matrix,BigMatrix-method *(BigMatrix-class)*, 3

BigMatrix, 3, *5–7*
BigMatrix-class, 3
BigMatrixFactor, *5*, 6
BigMatrixFactor-class, 6
bigmemoryExtras-package, 2

coerce,BigMatrix,matrix-method *(BigMatrix-class)*, 3
coerce,BigMatrixFactor,factor-method *(BigMatrixFactor-class)*, 6
coerce,BigMatrixFactor,matrix-method *(BigMatrixFactor-class)*, 6

dim,BigMatrix-method *(BigMatrix-class)*, 3
dimnames,BigMatrix-method *(BigMatrix-class)*, 3
dimnames<-,BigMatrix,ANY-method *(BigMatrix-class)*, 3

envRefClass, *4*, *6*

factor, *7*
filebacked.big.matrix, *5*, *7*

length,BigMatrix-method *(BigMatrix-class)*, 3
levels,BigMatrixFactor-method *(BigMatrixFactor-class)*, 6
levels<-,BigMatrixFactor-method *(BigMatrixFactor-class)*, 6

makeActiveBinding, *5*, *7*

ncol,BigMatrix-method *(BigMatrix-class)*, 3
nlevels,BigMatrixFactor-method *(BigMatrixFactor-class)*, 6
nrow,BigMatrix-method *(BigMatrix-class)*, 3

ReferenceClasses, *5*, *7*

updateBackingfiles, 8
updateBigMatrix, 8