Package 'alabaster.files'

November 8, 2026

Title Wrappers to Save Common File Formats

Version 1.10.0

Date 2023-12-30

License MIT + file LICENSE

Description Save common bioinformatics file formats within the alabaster framework.
This includes BAM, BED, VCF, bigWig, bigBed, FASTQ, FASTA and so on.
We save and load additional metadata for each file,
and we support linkage between each file and its corresponding index.

Depends alabaster.base,

Imports methods, S4Vectors, BiocGenerics, Rsamtools

Suggests BiocStyle, rmarkdown, knitr, testthat, VariantAnnotation,
rtracklayer, Biostrings

VignetteBuilder knitr

RoxygenNote 7.2.3

biocViews DataRepresentation, DataImport

git_url https://git.bioconductor.org/packages/alabaster.files

git_branch RELEASE_3_23

git_last_commit 7b7d0dc

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-08

Author Aaron Lun [aut, cre]

Maintainer Aaron Lun <infinite.monkeys.with.keyboards@gmail.com>

Contents

BamFileReference 2
BcfFileReference 3
BedFileReference 4
BgzipIndexWrapper 5

BigBedFileReference 6
BigWigFileReference 7
FastaFileReference 8
FastqFileReference 9
FileReference 10
GffFileReference 10
GmtFileReference 11
TabixIndexWrapper 12
virtual-wrapper 13

Index **15**

`BamFileReference` *Reference to a BAM file*

Description

Reference to a BAM file, for saving and reading in the **alabaster** framework.

Usage

```
BamFileReference(path, index = NULL)
```

Arguments

`path`	String containing the path to a BAM file.
`index`	String specifying the path to a BAI or CSI index file, or `NULL` if no index is available.

Value

A BamFileReference instance that can be used in `saveObject`.

Author(s)

Aaron Lun

Examples

```
# Using a BAM file from Rsamtools.
fl <- system.file("extdata", "ex1.bam", package="Rsamtools", mustWork=TRUE)

# Creating a BamFileReference.
wrapped <- BamFileReference(fl)
wrapped

# Fetching the path information:
path(wrapped)
wrapped$index
```

```
# Saving to disk:
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Reading it back again:
readObject(dir)
```

`BcfFileReference` *Reference to a BCF file*

Description

Reference to a BCF file, for saving and loading in the **alabaster** framework.

Usage

```
BcfFileReference(path, index = NULL)
```

Arguments

`path`	String containing the path to a BCF file.
`index`	String specifying the path to an index file in tabix or CSI format, or `NULL` if no index is available.

Value

A BcfFileReference instance that can be used in `saveObject`.

Author(s)

Aaron Lun

Examples

```
# Using Rsamtools's example file.
fl <- system.file("extdata", "ex1.bcf.gz", package="Rsamtools")

# Creating a BcfFileReference.
wrapped <- BcfFileReference(fl)
wrapped

# Fetching the path information:
path(wrapped)
wrapped$index

# Staging the BcfFileReference.
```

```
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

`BedFileReference` *Reference to a BED file*

Description

Reference to a BED file, for saving and loading in the **alabaster** framework.

Usage

```
BedFileReference(path, index = NULL)
```

Arguments

`path`	String containing the path to a Gzip- or BGZF-compressed BED file.
`index`	String containing a path to a tabix file. If supplied, `path` should be coordinate-sorted and BGZF-compressed.

Value

A BedFileReference instance that can be used in `saveObject`.

Author(s)

Aaron Lun

Examples

```
# Mocking up a BED file.
raw <- tempfile(fileext=".bed")
bed <- write("chr1\t2222\t33333", file=raw)
tmp <- tempfile(fileext=".bed.bgz")
Rsamtools::bgzip(raw, tmp)

# Creating a BedFileReference.
wrapped <- BedFileReference(tmp)
wrapped

# Extracting the paths:
path(wrapped)
wrapped$index
```

```
# Saving it to disk.
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

BgzipIndexWrapper *Wrapper for a Bgzip index file*

Description

This class is deprecated and only listed here for back-compatibility purposes.

Usage

```
BgzipIndexWrapper(path)
```

Arguments

path	String containing the path to a Bgzip index file.

Details

The BgzipIndexWrapper class is a subclass of a Wrapper, so all of the methods of the latter can also be used here, e.g., path.

Value

A BgzipIndexWrapper instance that can be used in stageObject.

Author(s)

Aaron Lun

Examples

```
# Mocking up a FASTA index file.
input <- system.file("extdata", "ce2dict1.fa", package="Rsamtools")
temp <- tempfile(fileext=".fa.bgz")
copy <- Rsamtools::bgzip(input, dest=temp)
Rsamtools::indexFa(copy)

# Creating a BgzipIndexWrapper.
wrapped <- BgzipIndexWrapper(paste0(copy, ".gzi"))
wrapped

# Staging the BgzipIndexWrapper.
```

```
dir <- tempfile()
library(alabaster.base)
info <- stageObject(wrapped, dir, "tab")
invisible(.writeMetadata(info, dir))
list.files(dir, recursive=TRUE)

# Loading it back again:
meta <- acquireMetadata(dir, "tab/file.fa.bgz.gzi")
loadObject(meta, dir)
```

`BigBedFileReference` *Reference to a bigBed file*

Description

Reference to a bigBed file, for saving and loading in the **alabaster** framework.

Usage

```
BigBedFileReference(path)
```

Arguments

`path`	String containing the path to a bigBed file.

Value

A BigBedFileReference instance that can be used in `stageObject`.

Author(s)

Aaron Lun

Examples

```
# Mocking up a bigBed file.
test_path <- system.file("tests", "test.bb", package = "rtracklayer")

# Creating a BigBedFileReference.
wrapped <- BigBedFileReference(test_path)
wrapped

# Staging the BigBedFileReference.
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

`BigWigFileReference` *Reference to a bigWig file*

Description

Reference to a bigWig file, for saving and loading in the **alabaster** framework.

Usage

```
BigWigFileReference(path)
```

Arguments

`path`	String containing the path to a bigWig file.

Value

A BigWigFileReference instance that can be used in `stageObject`.

Author(s)

Aaron Lun

Examples

```
# Mocking up a bigWig file.
test_path <- system.file("tests", "test.bw", package = "rtracklayer")

# Creating a BigWigFileReference.
wrapped <- BigWigFileReference(test_path)
wrapped

# Staging the BigWigFileReference.
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

`FastaFileReference` *Reference to a FASTA file*

Description

Reference to a FASTA file, for saving and loading in the **alabaster** framework.

Usage

```
FastaFileReference(path, seqtype = "DNA", faindex = NULL, gzindex = NULL)
```

Arguments

`path`	String containing the path to a Gzip- or BGZF-compressed FASTA file.
`seqtype`	String specifying the sequence type. This should be one of `"DNA"`, `"RNA"`, `"AA"` or `"custom"`.
`faindex`	String specifying the path to an FASTA index file, or `NULL` if no index is available. If an index is supplied, the file at `path` should be BGZF-compressed, and `gzindex` should also be supplied.
`gzindex`	String specifying the path to a BGZF index file, or `NULL` if no index is available. If an index is supplied, the file at `path` should be BGZF-compressed, and `faindex` should also be supplied.

Value

A FastaFileReference instance that can be used in `saveObject`.

Author(s)

Aaron Lun

Examples

```
# Mocking up a FASTA file.
tmp <- tempfile(fileext=".fa.gz")
write(">FOOBAR\nacgtacgt", gzfile(tmp))

# Creating a FastaFileReference.
wrapped <- FastaFileReference(tmp)
wrapped

# Saving to disk:
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

`FastqFileReference` *Reference to a FASTQ file*

Description

Reference to a FASTQ file, for saving and loading in the **alabaster** framework.

Usage

```
FastqFileReference(
  path,
  seqtype = "DNA",
  qualtype = "phred",
  qualoffset = 33,
  faindex = NULL,
  gzindex = NULL
)
```

Arguments

`path`	String containing the path to a Gzip- or BGZF-compressed FASTQ file.
`seqtype`	String specifying the sequence type. This should be one of `"DNA"`, `"RNA"`, `"AA"` or `"custom"`.
`qualtype`	String specifying the type of the quality strings. This should be one of `"phred"` or `"solexa"`.
`qualoffset`	Integer specifying the encoding offset for the quality strings. This is only used when `qualtype="phred"`, in which case it should either be 33 or 64.
`faindex`	String specifying the path to an FASTA index file, or `NULL` if no index is available. If an index is supplied, the file at `path` should be BGZF-compressed, and `gzindex` should also be supplied.
`gzindex`	String specifying the path to a BGZF index file, or `NULL` if no index is available. If an index is supplied, the file at `path` should be BGZF-compressed, and `faindex` should also be supplied.

Value

A FastqFileReference instance that can be used in `saveObject`.

Author(s)

Aaron Lun

Examples

```
# Mocking up a FASTQ file.
tmp <- tempfile(fileext=".fq.gz")
write("@FOOBAR\nacgtacgt\n+134987382", gzfile(tmp))

# Creating a FastqFileReference.
wrapped <- FastqFileReference(tmp)
wrapped

# Staging the FastqFileReference.
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

FileReference *Virtual file reference class*

Description

A virtual class for file reference objects. This implements common methods for `path`, `[[` and `$`.

Author(s)

Aaron Lun

GffFileReference *Reference to a GFF file*

Description

Reference to a GFF2/3 file, for saving and loading in the **alabaster** framework.

Usage

```
GffFileReference(path, index = NULL)
```

Arguments

`path`	String containing the path to a Gzip- or BGZF-compressed GFF file. The format is automatically detected from the file extension (`.gff2`, `.gff3` or `.gtf`).
`index`	String specifying the path to a tabix file in tabix format, or `NULL` if no index is available. If supplied, `path` should be coordinate-sorted and BGZF-compressed.

Value

A GffFileReference instance that can be used in saveObject.

Author(s)

Aaron Lun

Examples

```
# Using rtracklayer's example GFF file.
src <- system.file("tests", "genes.gff3", package = "rtracklayer")
fl <- tempfile(fileext=".gff3.gz")
writeLines(con=gzfile(fl), readLines(src))

# Creating a GffFileReference.
wrapped <- GffFileReference(fl)
wrapped

# Saving it:
dir <- tempfile()
saveObject(wrapped, dir)
list.files(dir, recursive=TRUE)

# Loading it back again:
readObject(dir)
```

GmtFileReference *Reference to a GMT file*

Description

Reference to a GMT file, for saving and loading in the **alabaster** framework.

Usage

```
GmtFileReference(path)
```

Arguments

path	String containing the path to a Gzip-compressed GMT file.

Value

A GmtFileReference instance that can be used in stageObject.

Author(s)

Aaron Lun

Examples

```
# Mocking up a GMT file.
tmp <- tempfile(fileext=".gmt.gz")
write("SET1\tdescription\tgene1\tgene2\tgene3", file=gzfile(tmp))

# Creating a GmtFileReference.
wrapped <- GmtFileReference(tmp)
wrapped

# Saving to disk:
dir <- tempfile()
saveObject(wrapped, dir)

# Loading it back again:
readObject(dir)
```

TabixIndexWrapper *Wrapper for a Tabix file*

Description

This class is deprecated and only listed here for back-compatibility purposes.

Usage

```
TabixIndexWrapper(path)
```

Arguments

path	String containing the path to a Tabix file.

Details

The TabixIndexWrapper class is a subclass of a Wrapper, so all of the methods of the latter can also be used here, e.g., path.

Value

A TabixIndexWrapper instance that can be used in stageObject.

Author(s)

Aaron Lun

Examples

```
# Mocking up a Tabix file.
test_tbx <- system.file("extdata", "example.gtf.gz.tbi", package="Rsamtools")

# Creating a TabixIndexWrapper.
wrapped <- TabixIndexWrapper(test_tbx)
wrapped

# Staging the TabixIndexWrapper.
dir <- tempfile()
library(alabaster.base)
info <- stageObject(wrapped, dir, "tab")
invisible(.writeMetadata(info, dir))
list.files(dir, recursive=TRUE)

# Loading it back again:
meta <- acquireMetadata(dir, "tab/file.tbi")
loadObject(meta, dir)
```

`virtual-wrapper` *Virtual wrapper classes*

Description

Defines some base classes for the concrete wrappers for specific file formats. This provides a standard set of methods that can be applied to all Wrapper instances.

Wrapper methods

Any instance `x` of a base Wrapper class can be used with the `path(x)` method, which returns a string containing the path to the file on the current file system.

The Wrapper class inherits from the Annotated class, so users can also get and set metadata via `metadata(x)`.

IndexedWrapper methods

The IndexedWrapper class inherits from the Wrapper class and can be used with all its methods. It additionally implements the `index(x)` method, which returns another Wrapper object for the associated index file (or `NULL`, if no index file exists).

CompressedWrapper methods

The CompressedWrapper class inherits from the Wrapper class and can be used with all its methods. It additionally implements the `compression(x)` method, which returns a string specifying the compression strategy.

CompressedIndexedWrapper methods

The CompressedIndexedWrapper class inherits from both the IndexedWrapper and CompressedWrapper classes and can be used with all their methods.

Author(s)

Aaron Lun

Index

[[, *10*
[[,FileReference-method
(FileReference), 10
[[<-,FileReference-method
(FileReference), 10
$, *10*
$,FileReference-method (FileReference), 10
$<-,FileReference-method
(FileReference), 10

Annotated, *13*

BamFileReference, 2
BamFileReference-class
(BamFileReference), 2
BamIndexWrapper (BamFileReference), 2
BamIndexWrapper-class
(BamFileReference), 2
BamWrapper (BamFileReference), 2
BamWrapper-class (BamFileReference), 2
BcfFileReference, 3
BcfFileReference-class
(BcfFileReference), 3
BedFileReference, 4
BedFileReference-class
(BedFileReference), 4
BedWrapper (BedFileReference), 4
BedWrapper-class (BedFileReference), 4
BgzipIndexWrapper, 5
BgzipIndexWrapper-class
(BgzipIndexWrapper), 5
BigBedFileReference, 6
BigBedFileReference-class
(BigBedFileReference), 6
BigBedWrapper (BigBedFileReference), 6
BigBedWrapper-class
(BigBedFileReference), 6
BigWigFileReference, 7
BigWigFileReference-class
(BigWigFileReference), 7
BigWigWrapper (BigWigFileReference), 7
BigWigWrapper-class
(BigWigFileReference), 7

CompressedIndexedWrapper-class
(virtual-wrapper), 13
CompressedWrapper-class
(virtual-wrapper), 13
compression (virtual-wrapper), 13

FaIndexWrapper (FastaFileReference), 8
FaIndexWrapper-class
(FastaFileReference), 8
FastaFileReference, 8
FastaFileReference-class
(FastaFileReference), 8
FastaWrapper (FastaFileReference), 8
FastaWrapper-class
(FastaFileReference), 8
FastqFileReference, 9
FastqFileReference-class
(FastqFileReference), 9
FastqWrapper (FastqFileReference), 9
FastqWrapper-class
(FastqFileReference), 9
FileReference, 10
FileReference-class (FileReference), 10

GffFileReference, 10
GffFileReference-class
(GffFileReference), 10
GffWrapper (GffFileReference), 10
GffWrapper-class (GffFileReference), 10
GmtFileReference, 11
GmtFileReference-class
(GmtFileReference), 11
GmtWrapper (GmtFileReference), 11
GmtWrapper-class (GmtFileReference), 11

index (virtual-wrapper), 13
index,IndexedWrapper-method (virtual-wrapper), 13
IndexedWrapper-class (virtual-wrapper), 13

loadBamIndexWrapper (BamFileReference), 2
loadBamWrapper (BamFileReference), 2
loadBedWrapper (BedFileReference), 4
loadBgzipIndexWrapper (BgzipIndexWrapper), 5
loadBigBedWrapper (BigBedFileReference), 6
loadBigWigWrapper (BigWigFileReference), 7
loadFaIndexWrapper (FastaFileReference), 8
loadFastaWrapper (FastaFileReference), 8
loadFastqWrapper (FastqFileReference), 9
loadGffWrapper (GffFileReference), 10
loadGmtWrapper (GmtFileReference), 11
loadTabixIndexWrapper (TabixIndexWrapper), 12
loadVcfWrapper (BcfFileReference), 3

metadata, *13*

path, *10*
path (virtual-wrapper), 13
path,FileReference-method (FileReference), 10
path,Wrapper-method (virtual-wrapper), 13

readBamFileReference (BamFileReference), 2
readBcfFileReference (BcfFileReference), 3
readBedFileReference (BedFileReference), 4
readBigBedFileReference (BigBedFileReference), 6
readBigWigFileReference (BigWigFileReference), 7
readFastaFileReference (FastaFileReference), 8
readFastqFileReference (FastqFileReference), 9
readGffFileReference (GffFileReference), 10
readGmtFileReference (GmtFileReference), 11

saveObject, *2–4*, *8*, *9*, *11*
saveObject,BamFileReference-method (BamFileReference), 2
saveObject,BcfFileReference-method (BcfFileReference), 3
saveObject,BedFileReference-method (BedFileReference), 4
saveObject,BigBedFileReference-method (BigBedFileReference), 6
saveObject,BigWigFileReference-method (BigWigFileReference), 7
saveObject,FastaFileReference-method (FastaFileReference), 8
saveObject,FastqFileReference-method (FastqFileReference), 9
saveObject,GffFileReference-method (GffFileReference), 10
saveObject,GmtFileReference-method (GmtFileReference), 11
show,CompressedIndexedWrapper-method (virtual-wrapper), 13
show,CompressedWrapper-method (virtual-wrapper), 13
show,FastaWrapper-method (FastaFileReference), 8
show,FastqWrapper-method (FastqFileReference), 9
show,FileReference-method (FileReference), 10
show,IndexedWrapper-method (virtual-wrapper), 13
show,Wrapper-method (virtual-wrapper), 13
stageObject, *5–7*, *11*, *12*
stageObject,BamIndexWrapper-method (BamFileReference), 2
stageObject,BamWrapper-method (BamFileReference), 2
stageObject,BedWrapper-method (BedFileReference), 4
stageObject,BgzipIndexWrapper-method (BgzipIndexWrapper), 5
stageObject,BigBedWrapper-method (BigBedFileReference), 6

stageObject,BigWigWrapper-method
(BigWigFileReference), 7
stageObject,FaIndexWrapper-method
(FastaFileReference), 8
stageObject,FastaWrapper-method
(FastaFileReference), 8
stageObject,FastqWrapper-method
(FastqFileReference), 9
stageObject,GffWrapper-method
(GffFileReference), 10
stageObject,GmtWrapper-method
(GmtFileReference), 11
stageObject,TabixIndexWrapper-method
(TabixIndexWrapper), 12
stageObject,VcfWrapper-method
(BcfFileReference), 3

TabixIndexWrapper, 12
TabixIndexWrapper-class
(TabixIndexWrapper), 12

VcfWrapper (BcfFileReference), 3
VcfWrapper-class (BcfFileReference), 3
virtual-wrapper, 13

Wrapper, *5*, *12*
Wrapper-class (virtual-wrapper), 13