rbgzf Package

May 14, 2017

Type Package

Title BGZF (Blocked GNU Zip format) library

Version 1.0

Date 2016-10-02

Author Alain Viari

Maintainer `<alain.viari@inria.fr>`

Description a (thin) interface between R and the C bgzf implementation
of the htslib library available at https://github.com/samtools/htslib.

License GPL

Depends methods

LazyData TRUE

RoxygenNote 5.0.1

R topics documented:

`bgzf.close` *close bgzf file*

Description

close bgzf file

Usage

```
bgzf.close(bgzf)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open

Value

0 on success and -1 on error

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.close(fh)
```

`bgzf.flush` *flush bgzf file*

Description

flush bgzf file

Usage

```
bgzf.flush(bgzf)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open

Value

0 on success and -1 on error

bgzf.has.index *tell if index is present*

Description

tell if index is present

Usage

```
bgzf.has.index(bgzf)
```

Arguments

bgzf	BGZFPtr object

Value

logical TRUE if index is present

Note

this function just check for the index presence not its validity

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.has.index(fh)
bgzf.reindex(fh)
bgzf.has.index(fh)
bgzf.close(fh)
```

bgzf.index.build.init *setup indexing*

Description

this instructs bgzf to build index while compressing or uncompressing file.

Usage

```
bgzf.index.build.init(bgzf)
```

Arguments

bgzf	BGZFPtr object created by bgzf.open

Value

0 on success and -1 on error

Note

this function is mostly used when creating a file for writing. when reading from an existing file you may consider using bgzf.reindex instead.

See Also

bgzf.reindex, bgzf.index.load

`bgzf.index.dump` *dump index to file*

Description

dump index to file

Usage

```
bgzf.index.dump(bgzf, path)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open
`path`	path of index file

Value

0 on success and -1 on error

See Also

bgzf.reindex for a dynamic version. bgzf.index.dump

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.reindex(fh)
tmp <- tempfile()
bgzf.index.dump(fh, tmp)
bgzf.index.remove(fh)
bgzf.index.load(fh, tmp)
bgzf.useek(fh, 1024)
x <- bgzf.read(fh, 255)
identical(as.integer(x[(1:8)%%8==1])-127L, 1:32)
bgzf.close(fh)
unlink(tmp)
```

bgzf.index.load *load index from file*

Description

load index from file

Usage

```
bgzf.index.load(bgzf, path)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open
`path`	path to index file

Value

0 on success and -1 on error

See Also

bgzf.reindex for a dynamic version. bgzf.index.dump

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.reindex(fh)
tmp <- tempfile()
bgzf.index.dump(fh, tmp)
bgzf.index.remove(fh)
bgzf.index.load(fh, tmp)
bgzf.useek(fh, 1024)
x <- bgzf.read(fh, 255)
identical(as.integer(x[(1:8)%%8==1])-127L, 1:32)
bgzf.close(fh)
unlink(tmp)
```

bgzf.index.remove *remove index*

Description

remove index

Usage

```
bgzf.index.remove(bgzf)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open

Value

0 on success and -1 on error

`bgzf.is.bgzf` *check if a file is in the BGZF format*

Description

check if a file is in the BGZF format

Usage

```
bgzf.is.bgzf(path)
```

Arguments

`path`	file pathname

Value

1 if file is BGZF; 0 if not or on I/O error

Examples

```
path <- file.path(path.package(package="rbgzf"), "samples/test.bin.bgz")
bgzf.is.bgzf(path)
```

`bgzf.open` *open bgzf file*

Description

open bgzf file

Usage

```
bgzf.open(path, mode = "r")
```

Arguments

`path`	file pathname
`mode`	open mode, one of "r" or "w"

Value

BGZFPtr object or NULL on error

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.close(fh)
```

bgzf.read *read raw bytes from bgzf file*

Description

read raw bytes from bgzf file

Usage

```
bgzf.read(bgzf, nbytes)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open
`nbytes`	double number of bytes to read

Value

raw array of size nbytes (or less if EOF reached)

Note

to overcome the 32 bits integer limitation, nbytes is interpreted as a double. this is valid up to 53 bits (i.e 9007199254740992), larger values will result in incorrect number of bytes read.

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
x <- bgzf.read(fh, 128)
identical(as.integer(x[(1:8)%%8==1]), 0:15)
bgzf.close(fh)
```

bgzf.reindex *force reindexing whole bgzf file*

Description

this function (re)index bgzf file by performing the following actions:

- remove previous index (if any) and init new index
- rewind file to origin
- read by blocks (this force indexing while uncompressing)
- reposition to initial offset

Usage

```
bgzf.reindex(bgzf)
```

Arguments

bgzf	BGZFPtr object created by bgzf.open

Value

0 on success and -1 on error

Note

this procedure is quite quick (about 1Gb of uncompressed data per second). So, for files of moderate size it is therefore not useful to save the index on disk. For very large files (> 500 Gb) bgzf.index.load could be more advantageous.

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.reindex(fh)
bgzf.useek(fh, 1024)
x <- bgzf.read(fh, 255)
identical(as.integer(x[(1:8)%%8==1])-127L, 1:32)
bgzf.close(fh)
```

bgzf.useek *position file at uncompressed file offset*

Description

position file at uncompressed file offset

Usage

```
bgzf.useek(bgzf, uoffset)
```

Arguments

bgzf	BGZFPtr object created by bgzf.open
uoffset	offset in uncompressed file

Value

0 on success and -1 on error

Note

the file should have been indexed (at least up to requested position)

to overcome the 32 bits integer limitation, uoffset is interpreted as a double. this is valid up to 53 bits (i.e 9007199254740992), larger values may result in incorrect position.

See Also

bgzf.reindex, bgzf.index.load

Examples

```
fh <- bgzf.open(file.path(path.package(package="rbgzf"), "samples/test.bin.bgz"))
bgzf.reindex(fh)
bgzf.useek(fh, 1024)
x <- bgzf.read(fh, 255)
identical(as.integer(x[(1:8)%%8==1])-127L, 1:32)
bgzf.close(fh)
```

`bgzf.utell` *tell byte position in uncompressed file*

Description

tell byte position in uncompressed file

Usage

```
bgzf.utell(bgzf)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open

Value

the current offset in uncompressed file on success and -1 on error

Note

the file should have been indexed (at least up to current position)

See Also

bgzf.reindex, bgzf.index.load

`bgzf.write` *write raw bytes to bgzf file*

Description

write raw bytes to bgzf file

Usage

```
bgzf.write(bgzf, data)
```

Arguments

`bgzf`	BGZFPtr object created by bgzf.open
`data`	array of raw data

Value

number of bytes written (negative on error)

Examples

```
tmp <- tempfile()
fh <- bgzf.open(tmp, "w")
bgzf.write(fh, as.raw((1:1024)%%256))
bgzf.flush(fh)
bgzf.close(fh)
fh <- bgzf.open(tmp, "r")
x <- bgzf.read(fh, 255)
identical(as.integer(x), 1:255)
bgzf.close(fh)
unlink(tmp)
```

`print.BGZFPtr` *print method for BGZFPtr object*

Description

print method for BGZFPtr object

Usage

```
## S3 method for class 'BGZFPtr'
print(x, ...)
```

Arguments

`x`	a BGZFPtr object
`...`	further arguments passed to or from other methods

rbgzf *BGZF file format R interface*

Description

Bgzf (Blocked GNU Zip Format) is a variant of GZip format that allows direct access into compressed file using offset in the uncompressed file. The basic idea is to zip file by blocks (of 65K by default) and to maintain an index between blocks offsets in compressed and uncompressed data.

This R package provides a (thin) interface between R and the bgzf implementation of the htslib library available at:

`https://github.com/samtools/htslib`. (appropriate LICENSE file is available in the src directory).

The code has been slightly modified to remove dependencies with other library components (SAM, CRAM, ...).

Note: for WGInR users: WGInR contains an older (but compatible) version of bgzip. the code for multithreading read operations has unfortunately not (yet) been transfered from this version (to be done in future).

Details

Package:	rbgzf
Type:	Package
Version:	1.0
Date:	2016-11-02
License:	GPL

Author(s)

Alain Viari

Index