

# Package ‘basilisk.utils’

October 16, 2020

**Version** 1.0.4

**Date** 2020-05-17

**Title** Basilisk Installation Utilities

**Imports** utils, methods, rappdirs

**Suggests** knitr, rmarkdown, BiocStyle, testthat, BiocFileCache

**biocViews** Infrastructure

**Description** Implements utilities for installation of the basilisk package, primarily to avoid re-writing the same R code in both the configure script (for centrally administered R installations) and in the lazy installation mechanism (for distributed binaries). It is highly unlikely that developers - or, heaven forbid, end-users! - will need to interact with this package directly; they should be using the basilisk package instead.

**License** GPL-3

**RoxygenNote** 7.1.0

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/basilisk.utils>

**git\_branch** RELEASE\_3\_11

**git\_last\_commit** 4f4cc83

**git\_last\_commit\_date** 2020-05-19

**Date/Publication** 2020-10-16

**Author** Aaron Lun [aut, cre, cph]

**Maintainer** Aaron Lun <[infinite.monkeys.with.keyboards@gmail.com](mailto:infinite.monkeys.with.keyboards@gmail.com)>

## R topics documented:

clearExternalDir . . . . .	2
destroyOldVersions . . . . .	3
getBasiliskDir . . . . .	3
getBinaries . . . . .	4
getEnvironmentDir . . . . .	5
getExternalDir . . . . .	6
getLockFile . . . . .	7
installMiniconda . . . . .	7
isWindows . . . . .	8
useSystemDir . . . . .	9

---

clearExternalDir	<i>Clear the external installation directory</i>
------------------	--

---

### Description

Clear the external installation directory by removing old Miniconda instances or environments installed for different versions of **basilisk** with the same middle version number (i.e., same Bioconductor release).

### Usage

```
clearExternalDir()  
  
clearObsoleteDir(path = getExternalDir())
```

### Arguments

path                   String containing the path to the latest version of the directory of interest.

### Details

clearObsoleteDir can also be applied to the output of [getEnvironmentDir](#), as the package version is also suffixed onto those directory paths. This is useful for clearing out obsolete versions of package environments.

### Value

For clearExternalDir, all Miniconda instances (and associated environments) of the same Bioconductor release as the current **basilisk** installation are destroyed.

The same applies for clearObsoleteDir except that the Miniconda instance generated by the latest **basilisk** installation is retained.

### Author(s)

Aaron Lun

### See Also

[getExternalDir](#), which determines the location of the external directory.  
[installMiniconda](#), for the motivation behind this function.

### Examples

```
# We can't actually run clearExternalDir() here, as it  
# relies on basilisk already being installed.  
print("dummy test to pass BiocCheck")
```

---

destroyOldVersions      *Destroy old versions?*

---

### Description

Should we destroy old installations of Miniconda from previous versions of **basilisk** or its client packages?

### Usage

```
destroyOldVersions()
```

### Details

The default value is TRUE, in order to save some hard drive space. This can be changed by setting BASILISK\_NO\_DESTROY environment variable to "1".

### Value

Logical scalar providing an answer to the above.

### Author(s)

Aaron Lun

### See Also

[installMiniconda](#), where this function is used.

[clearObsoleteDir](#), which may be triggered by this function.

---

getBasiliskDir      *Get the **basilisk** Miniconda directory*

---

### Description

Find the installation directory for the **basilisk**-managed Miniconda instance.

### Usage

```
getBasiliskDir(installed = TRUE)
```

### Arguments

installed      Logical scalar indicating whether **basilisk** is already installed.

**Details**

By default, Miniconda is installed to a location specified by `getExternalDir`. This ensures that R package build systems do not attempt to generate binaries that include the Miniconda installation; such binaries are not relocatable due to the presence of hard-coded paths, resulting in run-time failures.

If the `BASILISK_EXTERNAL_ANACONDA` environment variable is set to a path to an existing Miniconda installation, the function will return it directly without modification. This allows users to use their own Miniconda instances with **basilisk** but, in turn, they are responsible for managing it.

If the `BASILISK_USE_SYSTEM_DIR` environment variable is set to "1", the function will return a path to a location inside the **basilisk** system installation directory. This is the ideal approach when installing from source as any Miniconda and **basilisk** re-installations are synchronized. It also ensures that any R process that can load **basilisk** will also have permissions to access the Miniconda instance, which makes life easier for sysadmins of clusters or other shared resources.

We suggest always calling this function after an `installMiniconda` call, which guarantees the presence of the Miniconda installation directory (or dies trying). Setting `installed=FALSE` should only happen inside the **basilisk** configure script.

**Value**

String containing the path to the Miniconda instance.

**Author(s)**

Aaron Lun

**Examples**

```
# Setting the environment variable to run this example:
# all other modes rely on installation of basilisk.
old <- Sys.getenv("BASILISK_USE_SYSTEM_DIR")
Sys.setenv(BASILISK_USE_SYSTEM_DIR=1)

getBasiliskDir(installed=FALSE)

Sys.setenv(BASILISK_USE_SYSTEM_DIR=old)
```

---

getBinaries

*Get binary paths*

---

**Description**

Get binary paths

**Usage**

```
getCondaBinary(loc)

getPythonBinary(loc)
```

**Arguments**

`loc` String containing the path to the root of a conda installation or environment.

## Details

This code is largely copied from **reticulate**, and is only present here as they do not export these utilities for general consumption.

## Value

String containing the path to the conda or Python executable inside loc. If loc is not supplied, the relative path from the root of the environment is returned.

## Author(s)

Aaron Lun

## Examples

```
getCondaBinary()  
  
getPythonBinary()
```

---

getEnvironmentDir	<i>Get the <b>basilisk</b> environment directory</i>
-------------------	--

---

## Description

Find the installation directory for the **basilisk** Python environments for a particular client package.

## Usage

```
getEnvironmentDir(pkgname, installed = TRUE)
```

## Arguments

pkgname	String containing the name of the <b>basilisk</b> client package responsible for generating the environment.
installed	Logical scalar indicating whether the client package is already installed.

## Details

By default, **basilisk** environments are installed to a location specified by [getExternalDir](#). This ensures that R package build systems do not attempt to generate binaries that include the Python/conda packages; such binaries are not relocatable due to the presence of hard-coded paths, resulting in runtime failures.

If the `BASILISK_USE_SYSTEM_DIR` environment variable is set to "1", the function will return a path to a location inside the client package's system installation directory. This is the ideal approach when installing from source as we guarantee synchronization in Python and R re-installations. It also ensures that any R process that can load the client package will also have permissions to access its environments, which makes life easier for sysadmins of clusters or other shared resources.

## Value

String containing the path to the environment directory.

**Author(s)**

Aaron Lun

**Examples**

```
# Setting the environment variable to run this example:
# all other modes rely on installation of the client.
old <- Sys.getenv("BASILISK_USE_SYSTEM_DIR")
Sys.setenv(BASILISK_USE_SYSTEM_DIR=1)

getEnvironmentDir("client.of.basilisk", installed=FALSE)

Sys.setenv(BASILISK_USE_SYSTEM_DIR=old)
```

---

getExternalDir

*Get external Miniconda installation*

---

**Description**

Define an external location for installing the Miniconda instance and **basilisk** environments.

**Usage**

```
getExternalDir()
```

**Details**

We add the version information so that re-installation of **basilisk** will install a new instance of Miniconda. (This assumes that **basilisk** and **basilisk.utils** have synchronized version bumps.) See [installMiniconda](#) for more details on how this external location is managed.

If the `BASILISK_EXTERNAL_DIR` environment variable is set to some location, this will be used instead as the installation directory. Setting this variable is occasionally necessary if the default path returned by [user\\_cache\\_dir](#) has spaces; or on Windows, if the 260 character limit is exceeded after combining the default path with deeply nested Miniconda paths.

**Value**

String containing a path to an appropriate external folder.

**Author(s)**

Aaron Lun

**See Also**

[getBasiliskDir](#), where this function is used for Windows or MacOSX.

**Examples**

```
# We can't actually run getExternalDir() here, as it
# either relies on basilisk already being installed.
print("dummy test to pass BiocCheck")
```

---

getLockFile	<i>Get lock file</i>
-------------	----------------------

---

**Description**

Get the path to a lock file, typically used to mark an installation in progress (or an incomplete installation that needs to be replaced).

**Usage**

```
getLockFile(path)
```

**Arguments**

path	String containing a path to a Miniconda installation or conda environment that is to be installed.
------	--

**Value**

String containing a path to a lock file, to be touched before installation starts and deleted after installation finishes *successfully*.

**Author(s)**

Aaron Lun

**Examples**

```
getLockFile("AAAA")
```

---

installMiniconda	<i>Install Miniconda</i>
------------------	--------------------------

---

**Description**

Install Miniconda (version 3, 2019.10) to a destination path that depends on the operating system. This skips the installation if said path already exists.

**Usage**

```
installMiniconda(installed = TRUE)
```

**Arguments**

installed	Logical scalar indicating whether <b>basilisk</b> is already installed. Should only be set to FALSE in <b>basilisk</b> configure scripts.
-----------	---

## Details

This function was originally created from code in <https://github.com/hafen/rminiconda>, also borrowing code from **reticulate**'s `install_miniconda` for correct Windows installation. We use **BiocFileCache** if available to avoid re-downloading the Miniconda installer upon **basilisk** re-installation.

Whenever `installMiniconda` is re-run (and `BASILISK_USE_SYSTEM_DIR` is not set, see `?getBasiliskDir`), the previous Miniconda installation and its various **basilisk** environments are destroyed. This avoids duplication of Miniconda instances that would otherwise chew up disk space at 3 GB a pop.

After the destruction of the previous instance, we rely on the client packages to recreate their required environments. They should do this automatically if they are using **basilisk** correctly.

Users can disable this destruction by setting the `BASILISK_NO_DESTROY` environment variable to "1". This may be necessary on rare occasions when running multiple R instances on the same Bioconductor release. (Setting this variable is not required for instances using different Bioconductor releases.)

## Value

An Miniconda instance is created at the location specified by `getBasiliskDir`. Nothing is performed if the instance already exists. A logical scalar is returned indicating whether a new instance was created.

## Author(s)

Aaron Lun

## Examples

```
# We can't actually run installMiniconda() here, as it
# either relies on basilisk already being installed or
# it has a hard-coded path to the basilisk system dir.
print("dummy test to pass BiocCheck")
```

---

isWindows

*Find the operating system*

---

## Description

Indicate whether we are on Windows or MacOSX.

## Usage

```
isWindows()
```

```
isMacOSX()
```

## Value

Logical scalar indicating whether we are on the specified OS.

## Author(s)

Aaron Lun



**Examples**

```
isWindows()  
isMacOSX()
```

---

useSystemDir	<i>Use the R system directory?</i>
--------------	------------------------------------

---

**Description**

Should we use the R system directory for installing **basilisk**'s Miniconda instance or client environments?

**Usage**

```
useSystemDir()
```

**Details**

The default value is FALSE to avoid problems with position-dependent code in packaged binaries. This can be changed by setting BASILISK\_USE\_SYSTEM\_DIR environment variable to "1".

**Value**

Logical scalar providing an answer to the above.

**Author(s)**

Aaron Lun

**See Also**

[getBasiliskDir](#) and [getEnvironmentDir](#), where this function is used.

# Index

clearExternalDir, 2  
clearObsoleteDir, 3  
clearObsoleteDir (clearExternalDir), 2  
  
destroyOldVersions, 3  
  
getBasiliskDir, 3, 6, 8, 9  
getBinaries, 4  
getCondaBinary (getBinaries), 4  
getEnvironmentDir, 2, 5, 9  
getExternalDir, 2, 4, 5, 6  
getLockFile, 7  
getPythonBinary (getBinaries), 4  
  
installMiniconda, 2-4, 6, 7  
isMacOSX (isWindows), 8  
isWindows, 8  
  
user\_cache\_dir, 6  
useSystemDir, 9