

Package ‘BiocInstaller’

April 4, 2014

Title Install/Update Bioconductor and CRAN Packages

Description Installs/updates Bioconductor and CRAN packages

Version 1.12.0

Author Dan Tenenbaum and Biocore Team

Maintainer Bioconductor Package Maintainer <maintainer@bioconductor.org>

biocViews Software

Depends R (>= 3.0.0)

Suggests RUnit, BiocGenerics

License Artistic-2.0

R topics documented:

biocinstallRepos	2
biocLite	3
BiocUpgrade	5
biocValid	6
biocVersion	7
Package Groups	8
useDevel	9
Index	12

biocinstallRepos *Display current Bioconductor and CRAN repositories.*

Description

Displays the URLs of the repositories used by [biocLite](#) to install Bioconductor and CRAN packages.

Usage

```
biocinstallRepos(siteRepos=character())
```

Arguments

siteRepos character() representing an additional repository in which to look for packages to install. This repository will be prepended to the default repositories returned by the function.

Value

Named character() of repositories.

See Also

[biocLite](#) Installs/updates Bioconductor/CRAN packages.

[install.packages](#) installs the packages themselves.

[chooseBioCmirror](#) lets you choose from a list of all public Bioconductor mirror URLs.

[chooseCRANmirror](#) lets you choose from a list of all public CRAN mirror URLs.

Examples

```
biocinstallRepos()

## Choose mirrors
## Not run:
chooseCRANmirror()
chooseBioCmirror()

## End(Not run)
```

Description

biocLite installs or updates Bioconductor and CRAN packages in a Bioconductor release. Upgrading to a new Bioconductor release requires additional steps; see <http://bioconductor.org/install>.

Usage

```
biocLite (pkgs=c("Biobase", "IRanges", "AnnotationDbi"),
          suppressUpdates=FALSE,
          suppressAutoUpdate=FALSE,
          siteRepos=character(),
          ask=TRUE, ...)
```

Arguments

pkgs	character() of package names to install or update. A missing value and suppressUpdates=FALSE updates installed packages, perhaps also installing Biobase, IRanges, and AnnotationDbi if they are not already installed.
suppressUpdates	logical(1) indicating whether to suppress automatic updating of all installed packages, or character() of regular expressions specifying which packages to NOT automatically update.
suppressAutoUpdate	logical(1) indicating whether the BiocInstaller package updates itself.
siteRepos	character() representing an additional repository in which to look for packages to install. This repository will be prepended to the default repositories (which you can see with biocinstallRepos).
ask	logical(1) indicating whether to prompt user before installed packages are updated, or the character string 'graphics', which brings up a widget for choosing which packages to update. If TRUE, user can choose whether to update all outdated packages without further prompting, to pick and choose packages to update, or to cancel updating (in a non-interactive session, no packages will be updated). Otherwise, the value is passed to update.packages .
...	Additional arguments. <code>lib.loc</code> is passed to old.packages (used to determine the library location of installed packages to be updated). <code>lib</code> is passed to install.packages (used to determine the library location where pkgs are to be installed).

Value

biocLite() returns the pkgs argument, invisibly.

See Also

[biocinstallRepos](#) returns the Bioconductor and CRAN repositories used by biocLite.

[install.packages](#) installs the packages themselves.

[update.packages](#) updates all installed packages.

[chooseBioCmirror](#) lets you choose from a list of all public Bioconductor mirror URLs.

[chooseCRANmirror](#) lets you choose from a list of all public CRAN mirror URLs.

[monograph_group](#), [RBioinf_group](#) and [biocases_group](#) return package names associated with Bioconductor publications.

[all_group](#) returns the names of all Bioconductor software packages.

Examples

```
## Not run:
## Change default Bioconductor and CRAN mirrors
chooseBioCmirror()
chooseCRANmirror()

## If you dont have the BiocInstaller package installed, you can
## quickly install and load it as follows:
source("http://bioconductor.org/biocLite.R")

## The most recent version of the BiocInstaller package is now loaded.
## No need to load it with library().

# installs default packages (if not already installed) and updates
# previously installed packages
biocLite()

## Now install a CRAN package:
biocLite("survival")

## install a Bioconductor package, but dont update all installed
## packages as well:
biocLite("GenomicRanges", suppressUpdates=TRUE)

## Install default packages, but do not update any package whose name
## starts with "org." or "BSgenome."
biocLite(suppressUpdates=c("^org\\.", "^BSgenome\\."))

## install a package from source:
biocLite("IRanges", type="source")

## install all Bioconductor software packages
biocLite(all_group())
```

```
## End(Not run)
## Show the Bioconductor and CRAN repositories that will be used to
## install/update packages.
biocinstallRepos()
```

BiocUpgrade	<i>Upgrade Bioconductor to the latest version available for this version of R</i>
-------------	---

Description

Downloads the latest version of the BiocInstaller package, and upgrades all currently installed packages to the latest repositories for this version of R.

To upgrade, use:

```
source("http://bioconductor.org/biocLite.R")
biocLite("BiocUpgrade")
```

See Also

[biocLite](#) Installs/updates Bioconductor/CRAN packages.

[chooseBioCmirror](#) lets you choose from a list of all public Bioconductor mirror URLs.

[chooseCRANmirror](#) lets you choose from a list of all public CRAN mirror URLs.

[biocinstallRepos](#) returns the Bioconductor and CRAN repositories used by biocLite.

[install.packages](#) installs the packages themselves.

Examples

```
## Not run:
source("http://bioconductor.org/biocLite.R")
biocLite("BiocUpgrade")

## End(Not run)
```

 biocValid

Validate installed package versions against biocLite versions.

Description

Check that installed packages are consistent (neither out-of-date nor too new) with the version of R and Bioconductor in use, using `biocLite` for validation.

Usage

```
biocValid(pkgs = installed.packages(lib.loc, priority = priority),
  lib.loc = NULL, priority = "NA", type = getOption("pkgType"),
  filters = NULL, silent = FALSE, ..., fix=FALSE)
```

Arguments

<code>pkgs</code>	A character list of package names for checking, or a matrix as returned by installed.packages .
<code>lib.loc</code>	The library location(s) of packages to be validated; see installed.packages .
<code>priority</code>	check validity of all, “base”, or “recommended” packages; see installed.packages .
<code>type</code>	The type of available package (e.g., binary, source) to check validity against; see available.packages .
<code>filters</code>	Filter available packages to check validity against; see available.packages .
<code>silent</code>	Report how packages are invalid (<code>silent=FALSE</code> , default) and abort execution, or return a logical(1) (<code>silent=TRUE</code>) indicating the overall validity of installed packages.
<code>...</code>	Additional arguments, passed to biocLite when <code>fix=TRUE</code> .
<code>fix</code>	When TRUE, invoke <code>biocLite</code> to reinstall (update or downgrade, as appropriate) invalid packages.

Details

This function compares the version of installed packages to the version of packages associated with the version of R and Bioconductor appropriate for the `BiocInstaller` package currently in use.

Packages are reported as ‘out-of-date’ if a more recent version is available at the repositories specified by `biocinstallRepos()`. Usually, `biocLite()` is sufficient to update packages to their most recent version.

Packages are reported as ‘too new’ if the installed version is more recent than the most recent available in the `biocinstallRepos()` repositories. It is possible to down-grade by re-installing a too new package “PkgA” with `biocLite("PkgA")`. It is important for the user to understand how their installation became too new, and to avoid this in the future.

Value

logical(1) indicating overall validity of installed packages.

Author(s)

Martin Morgan mtmorgan@fhcrc.org

See Also

[biocLite](#) to update installed packages.

Examples

```
try(biocValid())
```

biocVersion	<i>Bioconductor version</i>
-------------	-----------------------------

Description

This function reports the Bioconductor version in use, as determined by the BiocInstaller package.

Usage

```
biocVersion()
```

Value

package_version representing the Bioconductor version in use.

See Also

[biocLite](#) Installs/updates Bioconductor/CRAN packages.

[BiocUpgrade](#) Upgrading to new versions.

Examples

```
biocVersion()
```

Package Groups	<i>Convenience functions to return package names associated with Bioconductor publications.</i>
----------------	---

Description

Returns character vectors of packages associated with Bioconductor publications, which can then be passed to [biocLite\(\)](#)

Usage

```
monograph_group()
RBioinf_group()
biocases_group()
all_group()
```

Value

character() of package names.

See Also

[biocLite](#) Installs/updates Bioconductor/CRAN packages.
[biocinstallRepos](#) returns the Bioconductor and CRAN repositories used by [biocLite](#).
[install.packages](#) installs the packages themselves.
[chooseBioCmirror](#) lets you choose from a list of all public Bioconductor mirror URLs.
[chooseCRANmirror](#) lets you choose from a list of all public CRAN mirror URLs.

Examples

```
## Get the names of packages used in the book
## "Bioconductor Case Studies":
biocases_group()

## Get the names of packages used in the book
## "R Programming for Bioinformatics":
RBioinf_group()

## Get the names of packages used in the monograph
## "Bioinformatics and Computational Biology Solutions
## Using R and Bioconductor":
monograph_group()

## Get the names of all Bioconductor software packages
all_group()
```

useDevel *Get the 'devel' version of the BiocInstaller package.*

Description

Downloads the 'devel' version of the BiocInstaller package so that all subsequent invocations of [biocLite](#) and [biocinstallRepos](#) use the devel repositories.

Displays the URLs of the repositories used by [biocLite](#) to install Bioconductor and CRAN packages.

Should only be used with a release (or patched) version of R, freshly installed.

Usage

```
useDevel(devel=TRUE)
```

Arguments

`devel` Whether to look in the devel (TRUE) or release (FALSE) repositories in subsequent invocations of [biocLite](#) and [biocinstallRepos](#).

Details

With R going to a yearly release schedule and Bioconductor keeping its twice-yearly release schedule, the same version of R (3.0) can be used with two different versions of Bioconductor (2.11, release, and 2.12, devel). The version number of the BiocInstaller package is what is used to determine whether to download packages from the BioC 2.11 or 2.12 repositories. In keeping with Bioconductor versioning conventions, if the middle number (y in x.y.z) is even, the package is part of a release version; if odd, it's part of a devel version.

By default, when BiocInstaller is first installed on R-3.0, it will be set up to download BioC 2.11 packages.

If you want to change this, you can run the [useDevel](#) function. With argument TRUE (the default), it will download the devel version of BiocInstaller and subsequently all packages downloaded with [biocLite](#) will be from the BioC 2.12 (devel) repository. You should run [useDevel](#) only once.

It is possible to keep BioC 2.11 and 2.12 libraries separate, within the same installation of R.

The trick is to use the `R_LIBS_USER` environment variable. First, create two separate directories for your BioC release and devel packages. Suggested directory names are as follows:

Linux:

```
~/R/x86_64-unknown-linux-gnu-library/3.0-bioc-release
```

```
~/R/x86_64-unknown-linux-gnu-library/3.0-bioc-devel
```

Mac OS:

```
~/Library/R/3.0-bioc-release/library
```

```
~/Library/R/3.0-bioc-devel/library
```

Windows:

```
C:\Users\YOUR_USER_NAME\Documents\R\win-library\3.0-bioc-release
```

```
C:\Users\YOUR_USER_NAME\Documents\R\win-library\3.0-bioc-devel
```

(change YOUR_USER_NAME to your user name)

You can then invoke "R for bioc-devel" or "R for bioc-release" from the command line as follows:

Linux:

```
R_LIBS_USER=~R/x86_64-unknown-linux-gnu-library/3.0-bioc-release R
```

```
R_LIBS_USER=~R/x86_64-unknown-linux-gnu-library/3.0-bioc-devel R
```

Mac OS X:

```
R_LIBS_USER=~Library/R/3.0-bioc-release/library R R_LIBS_USER=~Library/R/3.0-bioc-devel/library R
```

Windows:

```
cmd /C "set R_LIBS_USER=C:\Users\YOUR_USER_NAME\Documents\R\win-library\3.0-bioc-release && R"
```

```
cmd /C "set R_LIBS_USER=C:\Users\YOUR_USER_NAME\Documents\R\win-library\3.0-bioc-devel && R"
```

(Note: this assumes that R.exe is in your PATH.)

If you launch R in this way and then invoke `.libPaths`, you'll see that the first item is your special release or devel directory. Packages will be installed to that directory and that is the first place that `library` will look for them. `biocLite`, `install.packages`, `update.packages` and friends all respect this setting.

On Linux and Mac OS X, you can create a bash alias to save typing. Add the following to your `~/bash_profile`:

Linux

```
alias Rdevel='R_LIBS_USER=~R/x86_64-unknown-linux-gnu-library/3.0-bioc-devel R'
```

```
alias Rrelease='R_LIBS_USER=~R/x86_64-unknown-linux-gnu-library/3.0-bioc-release R'
```

Mac OS X

```
alias Rdevel='R_LIBS_USER=~Library/R/3.0-bioc-devel/library R' alias Rrelease='R_LIBS_USER=~Library/R/3.0-bioc-release/library R'
```

You can then invoke these from the command line as

```
Rdevel
```

...and...

```
Rrelease
```

On Windows, you can create two shortcuts, one for devel and one for release. Go to My Computer and navigate to a directory that is in your PATH. Then right-click and choose New->Shortcut.

in the "type the location of the item" box, put:

```
cmd /C "set R_LIBS_USER=C:\Users\YOUR_USER_NAME\Documents\R\win-library\3.0-bioc-release && R"
```

...for release and

```
cmd /C "set R_LIBS_USER=C:\Users\YOUR_USER_NAME\Documents\R\win-library\3.0-bioc-devel && R"
```

...for devel.

(again, it's assumed R.exe is in your PATH)

Click "Next".

In the "Type a name for this shortcut" box, type

Rdevel

or

Rrelease

You can invoke these from the command line as

Rdevel.lnk

...and...

Rrelease.lnk

(You must type in the .lnk extension.)

Because R_LIBS_USER is an environment variable, its value should be inherited by any subprocesses started by R, so they should do the right thing as well.

Value

Invisible NULL.

See Also

[biocinstallRepos](#) returns the Bioconductor and CRAN repositories used by `biocLite`.

[biocLite](#) Installs/updates Bioconductor/CRAN packages.

[install.packages](#) installs the packages themselves.

[chooseBioCmirror](#) lets you choose from a list of all public Bioconductor mirror URLs.

[chooseCRANmirror](#) lets you choose from a list of all public CRAN mirror URLs.

Examples

```
## Not run:  
useDevel()
```

```
## End(Not run)
```

Index

*Topic **environment**

- biocinstallRepos, 2
- biocLite, 3
- BiocUpgrade, 5
- biocValid, 6
- biocVersion, 7
- Package Groups, 8
- useDevel, 9
- .libPaths, 10

- all_group, 4
- all_group (Package Groups), 8
- available.packages, 6

- biocases_group, 4
- biocases_group (Package Groups), 8
- biocinstallRepos, 2, 3–5, 8, 9, 11
- biocLite, 2, 3, 5–11
- BiocUpgrade, 5, 7
- biocValid, 6
- biocVersion, 7

- chooseBioCmirror, 2, 4, 5, 8, 11
- chooseCRANmirror, 2, 4, 5, 8, 11

- install.packages, 2–5, 8, 10, 11
- installed.packages, 6

- library, 10

- monograph_group, 4
- monograph_group (Package Groups), 8

- old.packages, 3

- Package Groups, 8

- RBioinf_group, 4
- RBioinf_group (Package Groups), 8

- update.packages, 3, 4, 10
- useDevel, 9, 9