

# Package ‘BioInstaller’

January 24, 2018

**Title** Lightweight Biology Software Installer

**Version** 0.3.3

**Description** Can be used to install and download massive bioinformatics analysis softwares and databases, such as NGS reads mapping tools with its required databases.

**Depends** R (>= 3.3.0)

**URL** <https://github.com/JhuangLab/BioInstaller>

**BugReports** <https://github.com/JhuangLab/BioInstaller/issues>

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** stringr (>= 1.2.0), futile.logger (>= 1.4.1), configr (>= 0.3.1.1), git2r (>= 0.0.3), R.utils (>= 2.5.0), RCurl (>= 1.95-4.8), rvest (>= 0.3.2), devtools (>= 1.13.2), stringi (>= 1.1.5),

**RoxygenNote** 6.0.1

**Suggests** knitr, rmarkdown, testthat

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Jianfeng Li [aut, cre]

**Maintainer** Jianfeng Li <lee\_jianfeng@sjtu.edu.cn>

**Repository** CRAN

**Date/Publication** 2018-01-24 14:56:38 UTC

## R topics documented:

BioInstaller . . . . .	2
change.info . . . . .	2
craw.all.versions . . . . .	3
del.info . . . . .	4

docker.pull . . . . .	5
docker.search . . . . .	5
get.info . . . . .	6
get.meta . . . . .	7
get.meta.files . . . . .	8
install.bioinfo . . . . .	9
install.github . . . . .	10
install.nongithub . . . . .	12
is.biossoftwares.db.active . . . . .	14
set.biossoftwares.db . . . . .	14
show.installed . . . . .	15

## Index 16

---

BioInstaller	<i>This package can be used to install and download massive bioinformatics analysis softwares and databases, such as NGS reads mapping tools with its required databases.</i>
--------------	---

---

### Description

This package can be used to install and download massive bioinformatics analysis softwares and databases, such as NGS reads mapping tools with its required databases.

### Author(s)

Li Jianfeng [lee\\_jianfeng@sjtu.edu.cn](mailto:lee_jianfeng@sjtu.edu.cn)

### See Also

Useful links:

<https://github.com/JhuangLab/BioInstaller>

Report bugs at <https://github.com/JhuangLab/BioInstaller/issues>

---

change.info	<i>Update biologically softwares information of system</i>
-------------	--

---

### Description

Update biologically softwares information of system

### Usage

```
change.info(name = "", installed = TRUE, source.dir = "", bin.dir = "",
  executable.files = "", db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE",
  tempfile()), ..., verbose = TRUE)
```

**Arguments**

name	Software name
installed	Whether be installed successful in system
source.dir	Directorie of softwares source code
bin.dir	Directorie of softwares bin
executable.files	Executable files in bin.dir
db	File saving softwares infomation
...	Other key and value paired need be saved in BioInstaller
verbose	Ligical indicating wheather show the log message

**Value**

Bool Value

**Examples**

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biossoftwares.db(db)
change.info(name = 'demo', installed = 'yes', source.dir = '',
bin.dir = '', excutable.files = c('demo'), others.customer = 'demo')
unlink(db)
```

---

crawl.all.versions      *A function can be used to crawl all source code from nongithub.cfg stored information*

---

**Description**

A function can be used to crawl all source code from nongithub.cfg stored information

**Usage**

```
crawl.all.versions(name, download.dir = "./",
nongithub.cfg = c(system.file("extdata", "config/nongithub/nongithub.toml",
package = "BioInstaller"), system.file("extdata", "config/db/db_main.toml",
package = "BioInstaller"), system.file("extdata", "config/db/db_annovar.toml",
package = "BioInstaller"), system.file("extdata", "config/db/db_blast.toml",
package = "BioInstaller")), parse.extra.params = list(extra.list = list(),
rcmd.parse = TRUE, bash.parse = TRUE, glue.parse = TRUE), license = "")
```

**Arguments**

name	Software name
download.dir	Download destdir
nongithub.cfg	Configuration file of installed by non github url, default is system.file('extdata', 'config/nongithub/nongithub.toml', package='BioInstaller')
parse.extra.params	Other parameters pass to <a href="#">parse.extra</a>
license	The BioInstaller download license code.

**Examples**

```
craw.all.versions('demo')
```

---

del.info	<i>Delete biologically softwares information of system</i>
----------	--

---

**Description**

Delete biologically softwares information of system

**Usage**

```
del.info(name = "", db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE", tempfile()),
         verbose = TRUE)
```

**Arguments**

name	Software name
db	File saving softwares information
verbose	Logical indicating whether show the log message

**Value**

Bool Value

**Examples**

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biosoftwares.db(db)
change.info(name = 'bwa', installed = 'yes', source.dir = '',
            bin.dir = '', executable.files = c('demo'), others.customer = 'demo')
del.info('bwa')
unlink(db)
```

---

docker.pull                      *Use docker to pull image*

---

**Description**

Use docker to pull image

**Usage**

```
docker.pull(repo, name, version = NULL, docker.bin = NULL,
  all.tags = FALSE, disable.content.trust = TRUE, verbose = TRUE)
```

**Arguments**

repo,	Repository name of docker hub, e.g life2cloud
name	Software name, e.g bwa
version	Image version
docker.bin	Docker executable file, default is 'docker' in \$PATH
all.tags	Download all tagged images in the repository
disable.content.trust	Skip image verification (default true)
verbose	Logical indicating wheather show the log message

**Value**

Bool Value

**Examples**

```
docker.bin <- unname(Sys.which('docker'))
if (docker.bin != '') {
  docker.pull(repo = 'learn', name = 'tutorial')
}
```

---

docker.search                      *Search softwares docker infomation in BioInstaller docker database*

---

**Description**

Search softwares docker infomation in BioInstaller docker database

**Usage**

```
docker.search(name, docker.db = system.file("extdata",
  "config/docker/docker.toml", package = "BioInstaller"))
```

**Arguments**

name	Software name, e.g bwa
docker.db	A list including docker repo information, default to use built-in config/docker/docker.toml

**Value**

A list

**Examples**

```
docker.search('bwa')
```

---

get.info	<i>Show biologically softwares information of system</i>
----------	--

---

**Description**

Show biologically softwares information of system

**Usage**

```
get.info(name = "", db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE", tempfile()),
         verbose = TRUE)
```

**Arguments**

name	Software name
db	File saving softwares information
verbose	Logical indicating whether show the log message

**Value**

Bool Value

**Examples**

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biosoftwares.db(db)
change.info(name = 'bwa', installed = 'yes', source.dir = '',
            bin.dir = '', executable.files = c('demo'), others.customer = 'demo')
get.info('bwa')
unlink(db)
```

---

get.meta	<i>Get meta information of BioInstaller collected sources, such as database, GitHub source, non-GitHub source, web source</i>
----------	---

---

## Description

Get meta information of BioInstaller collected sources, such as database, GitHub source, non-GitHub source, web source

## Usage

```
get.meta(value = NULL, config = NULL, get.meta.files.params = NULL,  
read.config.params = NULL)
```

## Arguments

value	Avaliable option for 'db', 'github', 'nongithub': 'cfg_meta', 'item'; for web: 'item'
config	Avaliable option: 'db', 'db_meta_file', 'github', 'github_meta_file', 'nongithub', 'nongithub_meta_file', 'web', 'web_meta_file'
get.meta.files.params	Params pass to <a href="#">get.meta.files</a>
read.config.params	Params pass to <a href="#">read.config</a>

## Value

List contain the meta files path of BioInstaller collected sources

## Examples

```
meta <- get.meta()  
db_cfg_meta <- get.meta(config = 'db', value = 'cfg_meta')  
db_meta_file <- get.meta(config = 'db_meta_file')  
db_cfg_meta_parsed <- get.meta(value = 'cfg_meta', config = 'db',  
read.config.params = list(rcmd.parse = TRUE))
```

---

get.meta.files	<i>Get all BioInstaller meta files path, such as database, GitHub source, non-GitHub source, web source</i>
----------------	---

---

### Description

Get all BioInstaller meta files path, such as database, GitHub source, non-GitHub source, web source

### Usage

```
get.meta.files(db.meta = system.file("extdata", "config/db/db_meta.toml",
  package = "BioInstaller"), github.meta = system.file("extdata",
  "config/github/github_meta.toml", package = "BioInstaller"),
  nongithub.meta = system.file("extdata",
  "config/nongithub/nongithub_meta.toml", package = "BioInstaller"),
  web.meta = system.file("extdata", "config/web/web_meta.toml", package =
  "BioInstaller"))
```

### Arguments

db.meta	Database source meta file path, default is system.file('extdata', 'config/db/db_meta.toml', package = 'BioInstaller')
github.meta	Github source meta file path, default is system.file('extdata', 'config/github/github_meta.toml', package = 'BioInstaller')
nongithub.meta	non-Github source meta file path, default is system.file('extdata', 'config/nongithub/nongithub_meta.toml', package = 'BioInstaller')
web.meta	Web source meta file path, default is system.file('extdata', 'config/web/web_meta.toml', package = 'BioInstaller')

### Value

List contain the meta files path of BioInstaller collected sources

### Examples

```
get.meta.files()
```



---

install.bioinfo	<i>Download and install biology software or database</i>
-----------------	--

---

## Description

Download and install biology software or database

## Usage

```
install.bioinfo(name = c(), download.dir = c(), destdir = c(),
  name.saved = NULL, github.cfg = system.file("extdata",
  "config/github/github.toml", package = "BioInstaller"),
  nongithub.cfg = c(system.file("extdata", "config/nongithub/nongithub.toml",
  package = "BioInstaller"), system.file("extdata", "config/db/db_main.toml",
  package = "BioInstaller"), system.file("extdata", "config/db/db_annovar.toml",
  package = "BioInstaller"), system.file("extdata", "config/db/db_blast.toml",
  package = "BioInstaller")), version = c(), local.source = NULL,
  show.all.versions = FALSE, show.all.names = FALSE,
  db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE", system.file("extdata",
  "demo/software_s_db_demo.yaml", package = "BioInstaller")),
  download.only = FALSE, decompress = TRUE, dependence.need = TRUE,
  showWarnings = FALSE, extra.list = list(), rcmd.parse = TRUE,
  bash.parse = TRUE, glue.parse = TRUE, glue.flag = "!!glue",
  save.to.db = TRUE, license = "", verbose = TRUE, ...)
```

## Arguments

name	Software name
download.dir	A string, point the source code download destdir
destdir	A string, point the install path
name.saved	Software name when you want to install different version, you can use this to point the installed softwares name like 'GATK-3.7'
github.cfg	Configuration file of installed by github url, default is system.file('extdata', 'config/github/github.toml', package='BioInstaller')
nongithub.cfg	Configuration file of installed by non github url, default is c(system.file('extdata', 'config/nongithub/nongithub.toml', package = 'BioInstaller'), system.file('extdata', 'config/db/db_main.toml', package = 'BioInstaller'), system.file('extdata', 'config/db/db_annovar.toml', package = 'BioInstaller'), system.file('extdata', 'config/db/db_blast.toml', package = 'BioInstaller'))
version	Software version
local.source	Install from local source, github softwares need a cloned dir, and nongithub softwares can be installed from a compressed file (if it is a dir, you need set decompress to FALSE)
show.all.versions	Logical wheather show all avialable versions can be install

show.all.names	Logical wheather show all avaiable names can be install
db	File of saving softwares infomation, default is Sys.getenv('BIO_SOFTWARES_DB_ACTIVE', system.file('extdata', 'demo/softwares_db_demo.yaml', package = 'BioInstaller'))
download.only	Logicol indicating wheather only download source or file (non-github)
decompress	Logicol indicating wheather need to decompress the downloaded file, default is TRUE
dependence.need	Logical should the dependence should be installed
showWarnings	Logical should the warnings on failure be shown?
extra.list	A list that can replace the configuration file 'debug' by list(debug = TRUE), and debug will be setted to TRUE
rcmd.parse	Logical wheather parse '@>@str_replace('abc', 'b', 'c')@<@' in config to 'acc'
bash.parse	Logical wheather parse '#>#echo \$HOME#<#' in config to your HOME PATH
glue.parse	Logical wheather parse '!!glue1:5' in config to ['1','2','3','4','5']; ['nochange', '!!glue(1:5)', 'nochange'] => ['nochange', '1', '2', '3', '4', '5', 'nochange']
glue.flag	A character flage indicating wheater run glue() function to parse (Default is !!glue)
save.to.db	Ligical indicating wheather save the install infomation in db
license	The BioInstaller download license code.
verbose	Ligical indicating wheather show the log message
...	Other key and value paired need be saved in BioInstaller passed to <a href="#">change.info</a>

**Value**

Bool Value or a list

**Examples**

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biosoftwares.db(db)
install.bioinfo('bwa', show.all.versions = TRUE)
unlink(db)
```

---

install.github

*Install or download softwares from Github*

---

**Description**

Install or download softwares from Github

**Usage**

```
install.github(name = "", download.dir = NULL, destdir = NULL,
  version = NULL, local.source = NULL, show.all.versions = FALSE,
  name.saved = NULL, github.cfg = system.file("extdata",
  "config/github/github.toml", package = "BioInstaller"),
  db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE", system.file("extdata",
  "demo/software_s_db_demo.yaml", package = "BioInstaller")),
  download.only = FALSE, showWarnings = FALSE, dependence.need = TRUE,
  extra.list = list(), rcmd.parse = TRUE, bash.parse = TRUE,
  glue.parse = TRUE, glue.flag = "!!glue", save.to.db = TRUE,
  verbose = TRUE, ...)
```

**Arguments**

name	Software name
download.dir	A string, point the source code download destdir
destdir	A string, point the install path
version	Software version
local.source	Install from local source, github softwares need a cloned dir
show.all.versions	Logical wheather show all avaiable version can be install
name.saved	Software name when you want to install different version, you can use this to point the installed softwares name like 'GATK-3.7'
github.cfg	Configuration file of installed by github url, default is system.file('extdata', 'config/github/github.toml', package='BioInstaller')
db	File of saving softwares infomation, default is Sys.getenv('BIO_SOFTWARES_DB_ACTIVE', system.file('extdata', 'demo/software_s_db_demo.yaml', package = 'BioInstaller'))
download.only	Logicol indicating wheather only download source or file (non-github)
showWarnings	Logical should the warnings on failure be shown?
dependence.need	Logical should the dependence should be installed
extra.list	A list that can replace the configuration file 'debug' by list(debug = TRUE), and debug will be setted to TRUE
rcmd.parse	Logical wheather parse '@>@str_replace('abc', 'b', 'c')@<@' in config to 'acc'
bash.parse	Logical wheather parse '#>#echo \$HOME#<#' in config to your HOME PATH
glue.parse	Logical wheather parse '!!glue1:5' in config to ['1','2','3','4','5']; ['nochange', '!!glue(1:5)', 'nochange'] => ['nochange', '1', '2', '3', '4', '5', 'nochange']
glue.flag	A character flage indicating wheater run glue() function to parse (Default is !!glue)
save.to.db	Ligical indicating wheather save the install infomation in db
verbose	Ligical indicating wheather show the log message
...	Other key and value paired need be saved in BioInstaller passed to <a href="#">change.info</a>

**Value**

Bool Value

**Examples**

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biosoftwares.db(db)
install.github('bwa', show.all.versions = TRUE)
unlink(db)
```

---

install.nongithub      *Install or download softwares from non-Github Web site*

---

**Description**

Install or download softwares from non-Github Web site

**Usage**

```
install.nongithub(name = "", download.dir = NULL, destdir = NULL,
  version = NULL, local.source = NULL, show.all.versions = FALSE,
  name.saved = NULL, nongithub.cfg = c(system.file("extdata",
  "config/nongithub/nongithub.toml", package = "BioInstaller"),
  system.file("extdata", "config/db/db_main.toml", package = "BioInstaller"),
  system.file("extdata", "config/db/db_annovar.toml", package = "BioInstaller"),
  system.file("extdata", "config/db/db_blast.toml", package = "BioInstaller")),
  db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE", system.file("extdata",
  "demo/softwares_db_demo.yaml", package = "BioInstaller")),
  download.only = FALSE, decompress = TRUE, dependence.need = TRUE,
  showWarnings = FALSE, extra.list = list(), rcmd.parse = TRUE,
  bash.parse = TRUE, glue.parse = TRUE, glue.flag = "!!glue",
  save.to.db = TRUE, verbose = TRUE, ...)
```

**Arguments**

name	Software name
download.dir	A string, point the source code download destdir
destdir	A string, point the install path
version	Software version
local.source	Install from local source (a compressed file, if it is a dir, you need set decompress to FALSE)
show.all.versions	Logical wheather show all avaiable version can be install
name.saved	Software name when you want to install different version, you can use this to point the installed softwares name like 'GATK-3.7'

nongithub.cfg	Configuration file of installed by non github url, default is c(system.file('extdata', 'config/nongithub/nongithub.toml', package = 'BioInstaller'), system.file('extdata', 'config/db/db_main.toml', package = 'BioInstaller'), system.file('extdata', 'config/db/db_annotar.toml', package = 'BioInstaller'), system.file('extdata', 'config/db/db_blast.toml', package = 'BioInstaller'))
db	File of saving softwares infomation, default is Sys.getenv('BIO_SOFTWARES_DB_ACTIVE', system.file('extdata', 'demo/softwares_db_demo.yaml', package = 'BioInstaller'))
download.only	Logical indicating wheather only download source or file (non-github)
decompress	Logical indicating wheather need to decompress the downloaded file, default is TRUE
dependence.need	Logical should the dependence should be installed
showWarnings	Logical should the warnings on failure be shown?
extra.list	A list that can replace the configuration file 'debug' by list(debug = TRUE), and debug will be setted to TRUE
rcmd.parse	Logical wheather parse '@>@str_replace('abc', 'b', 'c')@<@' in config to 'acc'
bash.parse	Logical wheather parse '#>#echo \$HOME#<#' in config to your HOME PATH
glue.parse	Logical wheather parse '!!glue1:5' in config to ['1','2','3','4','5']; ['nochange', '!!glue(1:5)', 'nochange'] => ['nochange', '1', '2', '3', '4', '5', 'nochange']
glue.flag	A character flage indicating wheater run glue() function to parse (Default is !!glue)
save.to.db	Logical indicating wheather save the install infomation in db
verbose	Logical indicating wheather show the log message
...	Other key and value paired need be saved in BioInstaller passed to <a href="#">change.info</a>

## Value

Bool Value

## Examples

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biossoftwares.db(db)
install.nongithub('gmap', show.all.versions = TRUE)
unlink(db)
```

is.biosoftwares.db.active  
*Test active configuration file*

---

**Description**

Check whether a Bio Softwares DB is active

**Usage**

```
is.biosoftwares.db.active(biosoftwares.db)
```

**Arguments**

biosoftwares.db  
Configuration filename of bio-softwares db

**Value**

Logical indicating whether the specified configuration file is active

**Examples**

```
is.biosoftwares.db.active('config.cfg')
```

---

set.biosoftwares.db *Set BIO\_SOFWARES\_DB\_ACTIVE as the BioInstaller db*

---

**Description**

Set BIO\_SOFWARES\_DB\_ACTIVE as the BioInstaller db

**Usage**

```
set.biosoftwares.db(biosoftwares.db)
```

**Arguments**

biosoftwares.db  
Configuration filename of bio-softwares db

**Value**

Logical indicate wheather set db successful

**Examples**

```
set.biosoftwares.db(sprintf('%s/.BioInstaller', tempdir()))
```

---

show.installed	<i>Show all installed bio-sofwarens in system</i>
----------------	---

---

**Description**

Show all installed bio-sofwarens in system

**Usage**

```
show.installed(db = Sys.getenv("BIO_SOFTWARES_DB_ACTIVE", tempfile()),
  only.installed = TRUE, verbose = TRUE)
```

**Arguments**

db	File saving softwares infomation
only.installed	Logical wheather only show installed softwares in db
verbose	Ligical indicating wheather show the log message

**Value**

Bool Value

**Examples**

```
db <- sprintf('%s/.BioInstaller', tempdir())
set.biosoftwares.db(db)
change.info(name = 'bwa', installed = 'yes', source.dir = '',
  bin.dir = '', executable.files = c('demo'), others.customer = 'demo')
show.installed()
unlink(db)
```

# Index

BioInstaller, [2](#)  
BioInstaller-package (BioInstaller), [2](#)

change.info, [2](#), [10](#), [11](#), [13](#)  
crawl.all.versions, [3](#)

del.info, [4](#)  
docker.pull, [5](#)  
docker.search, [5](#)

get.info, [6](#)  
get.meta, [7](#)  
get.meta.files, [7](#), [8](#)

install.bioinfo, [9](#)  
install.github, [10](#)  
install.nongithub, [12](#)  
is.biossoftwares.db.active, [14](#)

parse.extra, [4](#)

read.config, [7](#)

set.biossoftwares.db, [14](#)  
show.installed, [15](#)