

# Package: mauricer (via r-universe)

December 5, 2024

**Type** Package

**Title** Work with 'BEAST2' Packages

**Version** 2.5.5

**Maintainer** Richèl J.C. Bilderbeek <rjcbilderbeek@gmail.com>

**Description** 'BEAST2' (<<https://www.beast2.org>>) is a widely used Bayesian phylogenetic tool, that uses DNA/RNA/protein data and many model priors to create a posterior of jointly estimated phylogenies and parameters. 'BEAST2' is commonly accompanied by 'BEAUti 2' (<<https://www.beast2.org>>), which, among others, allows one to install 'BEAST2' package. This package allows to work with 'BEAST2' packages from 'R'.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Imports** stringr, beastier (>= 2.5)

**Suggests** beautier (>= 2.6.11), curl, knitr, rmarkdown, testthat (>= 2.1.0), tracerer

**VignetteBuilder** knitr

**URL** <https://docs.ropensci.org/mauricer/> (website)  
<https://github.com/ropensci/mauricer>

**BugReports** <https://github.com/ropensci/mauricer/issues>

**SystemRequirements** BEAST2 (<https://www.beast2.org/>)

**Config/pak/sysreqs** libglpk-dev make default-jdk libicu-dev libxml2-dev  
libx11-dev zlib1g-dev

**Repository** <https://ropensci.r-universe.dev>

**RemoteUrl** <https://github.com/ropensci/mauricer>

**RemoteRef** master

**RemoteSha** ee96fa2c975dea505735e30be3b1e066fd0e8d1a

## Contents

default_params_doc . . . . .	2
get_beast2_pkg_names . . . . .	3
get_mrc_path . . . . .	4
get_mrc_paths . . . . .	5
install_beast2_pkg . . . . .	5
is_beast2_ns_pkg_installed . . . . .	6
is_beast2_pkg_installed . . . . .	7
uninstall_beast2_pkg . . . . .	8

## Index 10

---

default_params_doc	<i>This function does nothing. It is intended to inherit is parameters' documentation.</i>
--------------------	--

---

### Description

This function does nothing. It is intended to inherit is parameters' documentation.

### Usage

```
default_params_doc(beast2_folder, has_internet, name, show_warnings, verbose)
```

### Arguments

beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use <a href="#">get_default_beast2_folder</a> to get the default BEAST2 folder. Use <a href="#">get_default_beast2_bin_path</a> to get the full path to the default BEAST2 executable. Use <a href="#">get_default_beast2_jar_path</a> to get the full path to the default BEAST2 jar file.
has_internet	boolean to indicate if the user has access to the internet. By default, this value equals the result of <code>curl::has_internet</code>
name	the package's name
show_warnings	set to TRUE to show warnings
verbose	set to TRUE for extra output, as can be used in debugging

### Note

This is an internal function, so it should be marked with `@noRd`. This is not done, as this will disallow all functions to find the documentation parameters

### Author(s)

Richèl J.C. Bilderbeek

---

get\_beast2\_pkg\_names *Get all BEAST2 package names*

---

### Description

List all BEAST2 packages that are available and installed. Will **stop** if there is no internet connection

### Usage

```
get_beast2_pkg_names(  
  beast2_folder = beastier::get_default_beast2_folder(),  
  has_internet = curl::has_internet(),  
  verbose = FALSE  
)
```

### Arguments

**beast2\_folder** the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use [get\\_default\\_beast2\\_folder](#) to get the default BEAST2 folder. Use [get\\_default\\_beast2\\_bin\\_path](#) to get the full path to the default BEAST2 executable. Use [get\\_default\\_beast2\\_jar\\_path](#) to get the full path to the default BEAST2 jar file.

**has\_internet** boolean to indicate if the user has access to the internet. By default, this value equals the result of `curl::has_internet`

**verbose** set to TRUE for extra output, as can be used in debugging

### Value

a data frame with columns

1. name package name, for example, bdmm
2. installed\_version the installed version, for example, 2.6.2. installed\_version will be NA if the package is not installed
3. latest\_version version number of the latest version, for example, 2.6.3
4. dependencies packages the package depends on, for example BEASTLabs, GEO\_SPHERE. dependencies will be empty if there are no dependencies
5. description description of the package, for example Nested sampling for model selection and posterior inference

### Author(s)

Richèl J.C. Bilderbeek

**Examples**

```
if (beastier::is_beast2_installed() && curl::has_internet()) {  
  get_beast2_pkg_names()  
}
```

---

`get_mrc_path`*Get the full path of a mauricer file*

---

**Description**

Get the full path of a file in the inst/extdata folder. If there is no mauricer file, [get\\_mrc\\_path](#) will [stop](#).

**Usage**

```
get_mrc_path(filename)
```

**Arguments**

filename            the file's name, without the path

**Value**

the full path of the filename, if and only if the file is present. Will stop otherwise.

**Author(s)**

Richèl J.C. Bilderbeek

**See Also**

for more files, use [get\\_mrc\\_paths](#)

**Examples**

```
get_mrc_path("anthus_aco_sub.fas")
```

---

get_mrc_paths	<i>Get the full path of one or more mauricer files</i>
---------------	--

---

**Description**

Get the full paths of files in the inst/extdata folder. If there is a mauricer file absent, [get\\_mrc\\_paths](#) will *stop*.

**Usage**

```
get_mrc_paths(filenamees)
```

**Arguments**

filenamees      the files' names, without the path

**Value**

the filenamees' full paths

**Author(s)**

Richard J.C. Bilderbeek

**See Also**

for one file, use [get\\_mrc\\_path](#)

**Examples**

```
get_mrc_paths(c("anthus_aco_sub.fas", "anthus_nd2_sub.fas"))
```

---

install_beast2_pkg	<i>Install a BEAST2 package</i>
--------------------	---------------------------------

---

**Description**

Install a BEAST2 package. If the package is already installed, (see [is\\_beast2\\_pkg\\_installed](#)), this function *stops*.

**Usage**

```
install_beast2_pkg(  
  name,  
  beast2_folder = beastier::get_default_beast2_folder(),  
  verbose = FALSE,  
  has_internet = curl::has_internet()  
)
```

**Arguments**

name	the package's name
beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use <a href="#">get_default_beast2_folder</a> to get the default BEAST2 folder. Use <a href="#">get_default_beast2_bin_path</a> to get the full path to the default BEAST2 executable. Use <a href="#">get_default_beast2_jar_path</a> to get the full path to the default BEAST2 jar file.
verbose	set to TRUE for extra output, as can be used in debugging
has_internet	boolean to indicate if the user has access to the internet. By default, this value equals the result of <code>curl::has_internet</code>

**Value**

nothing. It does install the BEAST2 package

**Note**

Installing or uninstalling a BEAST2 package for a (singular) BEAST2 installation, does so for all BEAST2 installations

**Author(s)**

Richèl J.C. Bilderbeek

**Examples**

```
## Not run:
  install_beast2_pkg("NS")

## End(Not run)
```

---

```
is_beast2_ns_pkg_installed
```

*Is the BEAST2 NS package installed?*

---

**Description**

Determine if the BEAST2 NS package is installed.

**Usage**

```
is_beast2_ns_pkg_installed(
  show_warnings = FALSE,
  verbose = FALSE,
  beast2_folder = beastier::get_default_beast2_folder()
)
```

## Arguments

show\_warnings set to TRUE to show warnings

verbose set to TRUE for extra output, as can be used in debugging

beast2\_folder the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use [get\\_default\\_beast2\\_folder](#) to get the default BEAST2 folder. Use [get\\_default\\_beast2\\_bin\\_path](#) to get the full path to the default BEAST2 executable. Use [get\\_default\\_beast2\\_jar\\_path](#) to get the full path to the default BEAST2 jar file.

## Details

Unlike [is\\_beast2\\_pkg\\_installed](#), this function does not need an internet connection. Instead, the function calls BEAST2 to read a BEAST2 XML file that uses NS.

## Value

TRUE if the BEAST2 NS package is installed, FALSE otherwise

## Author(s)

Richèl J.C. Bilderbeek

## Examples

```
## Not run:  
  is_beast2_ns_pkg_installed()  
  
## End(Not run)
```

---

is\_beast2\_pkg\_installed

*Is a BEAST2 package installed?*

---

## Description

Checks if a BEAST2 package is installed.

## Usage

```
is_beast2_pkg_installed(  
  name,  
  beast2_folder = beastier::get_default_beast2_folder(),  
  has_internet = curl::has_internet()  
)
```

### Arguments

name	the package's name
beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use <a href="#">get_default_beast2_folder</a> to get the default BEAST2 folder. Use <a href="#">get_default_beast2_bin_path</a> to get the full path to the default BEAST2 executable. Use <a href="#">get_default_beast2_jar_path</a> to get the full path to the default BEAST2 jar file.
has_internet	boolean to indicate if the user has access to the internet. By default, this value equals the result of <code>curl :has_internet</code>

### Details

To be able to check this, an internet connection is needed. Without an internet connection, NULL is returned.

### Value

- TRUE if the BEAST2 package is installed
- FALSE if the BEAST2 package is not installed
- NULL if there is no internet connection

### Author(s)

Richèl J.C. Bilderbeek

### See Also

Use [is\\_beast2\\_ns\\_pkg\\_installed](#) to see if the NS package is installed without an internet connection

### Examples

```
## Not run:  
  is_beast2_pkg_installed("Beasy")  
  
## End(Not run)
```

---

uninstall\_beast2\_pkg *Uninstall a BEAST2 package*

---

### Description

Uninstall a BEAST2 package



**Usage**

```
uninstall_beast2_pkg(  
  name,  
  beast2_folder = beastier::get_default_beast2_folder(),  
  verbose = FALSE,  
  has_internet = curl::has_internet()  
)
```

**Arguments**

<code>name</code>	the package's name
<code>beast2_folder</code>	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use <a href="#">get_default_beast2_folder</a> to get the default BEAST2 folder. Use <a href="#">get_default_beast2_bin_path</a> to get the full path to the default BEAST2 executable. Use <a href="#">get_default_beast2_jar_path</a> to get the full path to the default BEAST2 jar file.
<code>verbose</code>	set to TRUE for extra output, as can be used in debugging
<code>has_internet</code>	boolean to indicate if the user has access to the internet. By default, this value equals the result of <code>curl::has_internet</code>

**Value**

nothing. It does install the BEAST2 package

**Note**

Installing or uninstalling a BEAST2 package for a (singular) BEAST2 installation, does so for all BEAST2 installations

**Author(s)**

Richèl J.C. Bilderbeek

**Examples**

```
## Not run:  
  uninstall_beast2_pkg("Beasy")  
  
## End(Not run)
```

# Index

`default_params_doc`, [2](#)

`get_beast2_pkg_names`, [3](#)

`get_default_beast2_bin_path`, [2](#), [3](#), [6–9](#)

`get_default_beast2_folder`, [2](#), [3](#), [6–9](#)

`get_default_beast2_jar_path`, [2](#), [3](#), [6–9](#)

`get_mrc_path`, [4](#), [4](#), [5](#)

`get_mrc_paths`, [4](#), [5](#), [5](#)

`install_beast2_pkg`, [5](#)

`is_beast2_ns_pkg_installed`, [6](#), [8](#)

`is_beast2_pkg_installed`, [5](#), [7](#), [7](#)

`stop`, [3–5](#)

`uninstall_beast2_pkg`, [8](#)