

Package ‘SRPM’

January 2, 2012

Version 0.1-6

Date 2010-04-12

Depends R (>= 2.10.0), methods, stashR

Imports utils, filehash, cacheSweave, stashR

LazyLoad yes

Collate author.R reader.R html.R zzz.R

Title Shared Reproducibility Package Management

Description A package development and management system for distributed reproducible research

License GPL (>= 2)

Author Roger D. Peng

Maintainer Roger D. Peng <rpeng@jhsph.edu>

Repository CRAN

Date/Publication 2010-04-12 18:23:47

R topics documented:

code	2
codeChunkList-class	3
codeObject-class	4
getRemoteURL	4
makeSRP	5
setPackage	6
toHTML	6
URLObjct-class	7
view	7
Index	8

code

*Retrieve code, figures, and cache databases***Description**

List available code chunks, figures, and cache databases and retrieve them

Usage

```
code(name = NULL)
figure(name)
cache(name)
runcode(namevec, env = parent.frame(), useCache = TRUE)
loadcache(namevec, env = parent.frame())
article()
edit(name, ...)

## S4 method for signature 'codeObject'
edit(name, ...)
```

Arguments

name	the name of a code chunk/figure/cache database (character) or its sequence number
namevec	a vector of names of code chunks, or an object of class "codeObject"
env	environment in which to evaluate code or load a cache database
useCache	logical, indicating whether a cache database should be used if available
...	other arguments passed to methods

Details

When `code` is called without arguments, the list of available objects is printed. Otherwise, the corresponding object is returned by the function.

`figure` views a figure associated with a code chunk (if there is one). Currently, this just loads the figure in the PDF viewer. In particular, the code which produced the figure is not run.

`cache` returns the database object associated with a code chunk if a cache database exists for that chunk.

`runcode` takes a code chunk name or a sequence of code chunk names and evaluates the code associated with that code chunk. If a code chunk has a cache database associated with it, then the code for that code chunk is *not* run and the objects from the cache database are lazy-loaded instead. If `namevec` is an object of class "codeObject", then `runcode` executes the code in that object.

`article` simply loads the article text in its finished form (currently, just a PDF file).

`edit` allows the user to edit the code associated with a code chunk in an external editor. Once editing is finished, `edit` returns a modified "codeObject" object.

Value

code and figure both return an object of class "codeObject". cache returns an object inheriting from class "filehash". edit returns an object of class "codeObject".

Author(s)

Roger D. Peng <rpeng@jhsph.edu>

Examples

```
pkg <- system.file("SRP-ex", "srp_simple", package = "SRPM")
setPackage(pkg)

## Show annotated list of code chunks
code()

## Load cache from code chunk 2
loadcache(2)
ls()

## Make table from code chunk 3
runcode(3)

## Show Figure 1
## Not run:
figure(1)

## End(Not run)
```

codeChunkList-class *Class "codeChunkList"*

Description

A vector of code chunk names

Slots

.Data: Object of class "character"

Extends

Class "character", from data part. Class "vector", by class "character", distance 2.

Methods

show: prints the code chunk names along with their sequence numbers and indicators as to whether there are figures [F] or cache databases [C] associated with the chunk

codeObject-class *Class "codeObject"*

Description

This class essentially represents a code chunk in a Sweave document. If the results of the code are cached, the cache database is stored, as are any figures that are created.

Slots

text: Object of class "character" ~~
db: Object of class "filehashOrNULL" ~~
fig: Object of class "character" ~~

Methods

show: prints the code corresponding to the code chunk, including comments, if any
view: loads code in the pager
viewplot: if there is a figure associated with this code chunk, the figure is loaded in the PDF viewer (only works for PDF figures)

getRemoteURL *Setting Remote URLs*

Description

Tools for getting/setting remote URLs for remote databases

Usage

```
getRemoteURL()  
getLocalDir()
```

Value

getRemoteURL() returns the URL for the remote collection of databases. getLocalDir() returns the directory on the local machine where copies of remote databases will be stored.

Author(s)

Roger D. Peng <rpeng@jhsph.edu>

`makeSRP`*Make a Reproducible Research Package*

Description

Create a reproducible research package from a Sweave file

Usage

```
makeSRP(pkg, SweaveFile = NULL, clean = TRUE)
makeWebpage(clear = FALSE)
```

Arguments

<code>pkg</code>	name of the package to be created
<code>SweaveFile</code>	name of the Sweave file
<code>clean</code>	logical, delete any files created in the package generation process?
<code>clear</code>	logical, should the directory in which the webpages will be written be removed if it already exists?

Details

`makeSRP` reads the data map file created by the **cacheSweave** package and uses the information to generate a "reproducible research package" (RRP). Any cached computation databases are copied into the package as well as any figures and the article itself. The figures and the article should be in PDF format.

After creating a reproducible research package with `makeSRP`, one can register the package with `setPackage` and use the `makeWebpage` function to construct a rudimentary web interface to the package. `makeWebpage` places the HTML files in a directory called "html" under the main directory of the reproducible research package.

If `SweaveFile` is `NULL` then it defaults to using `<pkg>.Rnw`.

Value

A list of objects of class `codeObject` corresponding to the code chunks in the Sweave file is returned (invisibly).

Author(s)

Roger D. Peng <rpeng@jhsph.edu>

setPackage	<i>Set and retrieve the currently registered package</i>
------------	--

Description

Set and retrieve the currently registered package

Usage

```
setPackage(name)
currentPackage()
```

Arguments

name	the name of the reproducible research package to be registered or a URL to a remote zip file
------	--

Details

setPackage sets the currently registered package so that functions such as code, figure, and cache know where to look for code chunks, figures, and cache databases. currentPackage returns the currently registered database.

If name is a URL to a remote zip file, the zip file is downloaded and an attempt is made to extract it. Upon successful extraction, the package is registered as if it were a local package.

Value

For currentPackage the name of the current package is returned.

toHTML	<i>Methods for Function toHTML</i>
--------	------------------------------------

Description

toHTML is a generic function which takes R objects as input and writes HTML representations to a connection or file.

Methods

x = "localDB" Writes a simple "index.html" file listing the keys in the database and for each key, a summary of the associated object is provided.

x = "ANY" The default method, essentially writes out `str(x)` to a file.

x = "function" This method prints out the entire function body.

x = "data.frame" The data frame is written out using `write.table`.

URLObject-class	Class "URLObject"
-----------------	-------------------

Description

A simple class for URLs

Slots

url: character, a URL

Methods

show: loads the URL into a web browser via the browseURL function

Examples

```
showClass("URLObject")
```

view	<i>View code objects</i>
------	--------------------------

Description

View code objects and their associated plots

Usage

```
view(object, ...)  
viewplot(object, ...)
```

Arguments

object	an object to be viewed
...	other arguments passed to methods

Details

Currently, there are methods for objects of class codeObject. view loads the code associated with a code object in the pager and viewplot launches a PDF viewer to view the associated plot.

Value

viewplot returns its argument invisibly.

Index

*Topic **classes**

codeChunkList-class, 3
codeObject-class, 4
URLObject-class, 7

*Topic **methods**

toHTML, 6
view, 7

*Topic **package**

makeSRP, 5

*Topic **utilities**

code, 2
getRemoteURL, 4
setPackage, 6

article (code), 2

cache (code), 2

character, 3

code, 2

codeChunkList-class, 3

codeObject-class, 4

currentPackage (setPackage), 6

edit (code), 2

edit, codeObject-method (code), 2

figure (code), 2

getLocalDir (getRemoteURL), 4

getRemoteURL, 4

loadcache (code), 2

makeSRP, 5

makeWebpage (makeSRP), 5

runcode (code), 2

setPackage, 6

show, codeChunkList-method
(codeChunkList-class), 3

show, codeObject-method
(codeObject-class), 4

show, URLObject-method
(URLObject-class), 7

toHTML, 6

toHTML, ANY-method (toHTML), 6

toHTML, data.frame-method (toHTML), 6

toHTML, function-method (toHTML), 6

toHTML, localDB-method (toHTML), 6

toHTML-methods (toHTML), 6

URLObject-class, 7

vector, 3

view, 7

view, codeObject-method
(codeObject-class), 4

viewplot (view), 7

viewplot, codeObject-method
(codeObject-class), 4