

Package ‘RXLisp’

September 1, 2002

Version 0.3

Date 2002/09/01

Title Interface between R and XLisp.

Author Duncan Temple Lang <duncan@research.bell-labs.com>

Depends R (>= 1.5.0)

Maintainer Duncan Temple Lang <duncan@research.bell-labs.com>

Description This provides an interface to call XLisp-Stat functions from within R, inspired by Forrest Young’s remarks about dynamic graphics, XLisp-Stat and R on R-devel.

License GPL

URL <http://www.omegahat.org/RXLisp>, <http://www.omegahat.org>
<http://www.omegahat.org/bugs>

R topics documented:

parseEval	1
.XLispInit	2
.XLisp	3
[[.XLispReference	4
Index	6

parseEval	<i>Parse and Evaluate an S command</i>
-----------	--

Description

Parse and evaluate an S command and return the result. This is a convenience function used by C code to call a single function to do the parsing and evaluation of an expression.

Usage

parseEval(cmd)

Arguments

`cmd` a string (character vector of length 1) giving the S command to parse and evaluate.

Details**Value**

The result of evaluating the S command.

Author(s)

Duncan Temple Lang

References

<http://www.omegahat.org/RSPerl>

Examples

```
parseEval("1:10")
parseEval("plot(rnorm(10)); abline(h=5); TRUE")
```

.XLispInit

Initialize XLisp-Stat Session

Description

This initializes the XLisp-Stat engine so that it can be used to process calls to XLisp functions, etc. This must be called before any access to the XLisp interpreter is made.

Usage

```
.XLispInit(args=.XLispInitArgs, registerEvent = TRUE)
```

Arguments

`args` a character vector giving the command line arguments used to initialize the session. The first value should be the pseudo name given to the application as in `argv[0]`, the name of the application, passed from the shell in a regular C application. The subsequent arguments are processed in the standard fashion by the XLisp-Stat engine.

`registerEvent` a logical value indicating whether the initialization should also connect the XLisp event loop to the R event loop. Generally, this will be `TRUE`. One may want to override this if a different event loop strategy will be used or no XLisp graphics will be deployed.

Details

This initializes the XLisp-Stat engine, including the memory management, initialization code, etc.

Value

NULL.

Author(s)

Duncan Temple Lang <duncan@research.bell-labs.com>

References

<http://www.xlispstat.org>

See Also

[.XLisp](#)

Examples

```
.XLispInit()  
.XLisp("+", 1, 2, 3, 4)
```

<code>.XLisp</code>	<i>Invoke XLisp-Stat Function</i>
---------------------	-----------------------------------

Description

This provides an interface to calling XLisp-Stat functions from R, converting the arguments from R values to XLisp-Stat objects and similarly converting the resulting value back to an R object. Functions are currently identified by name.

Usage

```
.XLisp(fun, ..., .convert = TRUE, upper = TRUE)
```

Arguments

- | | |
|-----------------------|---|
| <code>fun</code> | the name of the XLisp-Stat function, given as a character string. |
| <code>...</code> | any arguments to be passed to the XLisp-Stat function, which will be converted from R objects to XLisp-Stat values. Any named elements in this list are treated as named arguments to the XLisp function. This means they are converted to upper-case and prefixed with a <code>:</code> and inserted into the XLisp argument list. |
| <code>.convert</code> | a logical value indicating whether to convert the result back to a regular R object. If this is <code>FALSE</code> , a reference to the XLisp-Stat object is returned which can be used in subsequent calls to XLisp. If this is <code>TRUE</code> , an attempt to convert the value to an R object using the conversion mechanism is made. |
| <code>upper</code> | a logical value controlling whether to convert the (XLisp-Stat) function name being called to upper-case. |

Details

Value

An R object representing the result of the call to the XLisp function.

Author(s)

Duncan Temple Lang <duncan@research.bell-labs.com>

References

<http://www.xlispstat.org>, <http://www.omegahat.org/RXLisp>

See Also

[.XLispInit](#)

Examples

```
.XLispInit()
.XLisp("+", 1, 2, 3)
.XLisp("mean", c(1, 2, 3))

# Generate some random numbers from a Poisson distn.
# Note the need to specify an integer.
.XLisp("poisson-rand", as.integer(30), 1.8)

# returns a symbol which is not currently handled by the conversion
# mechanism. But the load will work!
.XLisp("load", system.file("examples", "Rinit.lsp", package="RXLisp"))
.XLisp("mypow", 3, 2)
```

[[.XLispReference *Syntactic methods for XLisp Reference Objects*

Description

These methods provide a convenient way to invoke methods and access slots in XLisp objects from within R. The subsetting operator invokes the XLisp `send` function on the XLisp object associated with the R reference object. The slot accessor (`$`) merely creates a function that, when called, will invoke the `send` function.

Usage

```
"[[.XLispReference"(x, i, ...)
"$ .XLispReference"(x, name)
```

Arguments

<code>x</code>	the XLispReference object on which to operate.
<code>i</code>	the name of the slot to be accessed (either queried or set).
<code>name</code>	the name of the method or message to invoke on the specified XLisp object
<code>...</code>	additional arguments (named or unnamed) to be passed in the method invocation.

Value

`[[` returns the value of the specified slot, converted to an R object.

`$` returns a function that when invoked will call the specified XLisp method for this object passing it the arguments in the call to that R function.

Author(s)

Duncan Temple Lang <duncan@research.bell-labs.com>

References

<http://www.xlispstat.org>, <http://www.omegahat.org/RXLisp>

See Also

[.XLispInit](#)

Examples

```
h <- .XLisp("histogram", rnorm(100))
h[["title"]]
h$close()
```

Index

*Topic **interface**

- .XLisp, [3](#)
- .XLispInit, [2](#)
- [[.XLispReference, [4](#)
- parseEval, [1](#)
- .XLisp, [3](#), [3](#)
- .XLispInit, [2](#), [4](#), [5](#)
- [[.XLispReference, [4](#)
- \$.XLispReference (*[[.XLispReference*),
 [4](#)

parseEval, [1](#)