

<http://stackoverflow.com/questions/28989855/the-difference-between-domc-and-doparallel-in-r>

<https://www.youtube.com/watch?v=wzKnU2ryT60>

<https://github.com/NREL/rplexos>

<http://stackoverflow.com/questions/28989855/the-difference-between-domc-and-doparallel-in-r>

The screenshot shows the RStudio interface. The source editor contains R code for setting up a parallel cluster using `makeCluster` and `registerDoParallel`. The console shows an error: `Error: could not find function "registerDoParallel"`. The Environment pane on the right shows the current environment with variables like `b.results`, `b.time`, `choi`, `c1`, and `combin`. The R Documentation pane on the right shows the documentation for `registerDoParallel`, including its description, usage, and arguments.

```
77 library("parallel", lib.loc="c:/Program Files/R/R-3.1.3/library")
78 library(doParallel)
79
80 detectCores()
81 # Create cluster with desired number of cores
82 c1 <- makeCluster(3)
83 c1 <- makeCluster(2)
84 # three clusters didnt work out was taking too much time ...
85 # Register cluster
86 registerDoParallel(c1)
87 # Register cluster
88 ??registerDoParallel
89
```

Console:

```
> c1 <- makeCluster(2)
> registerDoParallel(c1)
Error: could not find function "registerDoParallel"
> ??registerDoParallel
No documentation for 'registerDoParallel' in specified packages and
libraries:
you could try '??registerDoParallel'
> ??registerDoParallel
```

Environment:

- `b.results`: List of 3
 - `b.time`: Class 'proc_time' Named num [1:5] ...
 - `choi`: 50
- `c1`: List of 2
 - `combin`: int [1:3] 32 15 39

R Documentation: `registerDoParallel`

Description: The `registerDoParallel` function is used to register the parallel backend with the `foreach` package.

Usage: `registerDoParallel(c1, cores=NULL, ...)`

Arguments: `c1` A cluster object as returned by `makeCluster`, or the number of nodes to be created in the cluster. If not specified, on Windows a three worker cluster is created

The screenshot shows the RStudio interface after successful execution. The source editor contains R code using `foreach` with `%dopar%` to calculate the sum of squares. The console shows the output of the code. The Environment pane on the right shows the current environment with variables like `b.results`, `b.time`, `choi`, `c1`, `combin`, `i`, `test.1`, `x`, `xc`, and `xsummed`. The R Documentation pane on the right shows the documentation for `lockpick` function.

```
98 # contrast to a "for loop", foreach collects the results (step 3) and returns
99 # default. This can be changed through the ".combine" option:
100
101 # Use the concatenate function to combine results
102 xc <- foreach(i = 1:3, .combine = c) %dopar% sqrt(i)
103 xc
104 # Now xc is a vector
105
106 # Can also use + or * to combine results
107 xsummed <- foreach(i = 1:3, .combine = "+") %dopar% sqrt(i)
108 xsummed
109 # Now xsummed is a scalar, the sum of all the results
110
```

Console:

```
[[3]]
[1] 1.732051

> xc <- foreach(i = 1:3, .combine = c) %dopar% sqrt(i)
> xc
[1] 1.000000 1.414214 1.732051
> xsummed <- foreach(i = 1:3, .combine = "+") %dopar% sqrt(i)
> xsummed
[1] 4.146264
```

Environment:

- `b.results`: List of 3
 - `b.time`: Class 'proc_time' Named...
 - `choi`: 50
- `c1`: List of 2
 - `combin`: int [1:3] 32 15 39
- `i`: 3L
- `test.1`: "Number of iterations =..."
- `x`: List of 3
 - `xc`: num [1:3] 1 1.41 1.73
 - `xsummed`: 4.14626436994197

R Documentation: `lockpick` function (combination, ...)