

Programming in R



Educational Materials
©2004 VJ Carey and R Gentleman

Working with environments

- you can access data using: `get`, `[[`, `$`.
- you can not use a numeric index (it doesn't make sense), you must extract items using their names.
- basic user functions accomplished with `get`: `get([key], [environment])`
`> GOtags <- get("1001_at", env = hgu95av2GO)`
- if you have multiple keys, use `mget`, which will return a list

Lists and Environments

- Both allow you to store, together a collection of R objects of any type.
- `list` access is either by numeric index or name (if the items have names).
- `environment` access is only by name.
- the environment can be *hashed*, or not. If you are going to store many things then hashing makes sense.
- in recent versions of R accessing list elements by name has been sped up (so the differences in performance between the two is not large)
- environments are **not** copied when they are passed as values - so be careful

Using Lists

- `list` is a very flexible storage format in R

```
> x <- list()
> x[[1]] <- c(1, 2, 3)
> x[[2]] <- "foo"
> x[[3]] <- x
> names(x) <- c("bag", "nam", "lis")
> print(x)
```

```
$bag
```

```
[1] 1 2 3
```

```
$nam
```

```
[1] "foo"
```

```
$lis
```

```
$lis[[1]]
```

```
[1] 1 2 3
```

```
$lis[[2]]
```

```
[1] "foo"
```

List Element Access

```
> x[[1]]
```

```
[1] 1 2 3
```

```
> x[["bag"]]
```

```
[1] 1 2 3
```

```
> x$bag
```

```
[1] 1 2 3
```

```
> x["bag"]
```

```
$bag
```

```
[1] 1 2 3
```

Environments - Insertion

You can do the same things with an environment, but you *must* give all objects a name.

We first create an environment and then populate it.

```
> e1 = new.env(hash = TRUE)
> e1$bag = c(1, 2, 3)
> e1$nam = "foo"
> e1$lis = x
```

Environments - Extraction

To get the elements back out we can use the same operators.

```
> e1$nam
```

```
[1] "foo"
```

```
> e1[["bag"]]
```

```
[1] 1 2 3
```

```
> get("nam", env = e1)
```

```
[1] "foo"
```

```
> ls(env = e1)
```

```
[1] "bag" "lis" "nam"
```

```
> zz = as.list(e1)
```

Environments

Most of the Bioconductor meta-data packages are stored in environments. You can use the operators described above to get data out of an environment in a meta-data package.

```
> hgu95av2SYMBOL
```

```
<environment: 0x699bc08>
```

```
> ls(hgu95av2SYMBOL)[1:5]
```

```
[1] "1000_at" "1001_at" "1002_f_at" "1003_s_at" "1004_at"
```

```
> get("1001_at", env = hgu95av2G0)
```

```
 $"G0:0004714"
```

```
 $"G0:0004714"$G0ID
```

```
[1] "G0:0004714"
```

```
 $"G0:0004714"$Evidence
```

```
[1] "TAS"
```


Environments

Some other operations that can be performed on environments.

- They can be locked or sealed. This means that no new bindings can be added (`lockEnvironment`).
- `eapply` will be in the next release of R
- `DPEXplorer` from the *tkWidgets* package can be used to explore these data.