./scripts/F.R

```r
 1   generate.patches <-
 2   function(environment.raster, n.patches = 50, patch.size = 10)
 3   {
 4     dummy.raster <- environment.raster
 5
 6     n.cols <- ncol(dummy.raster)
 7     n.rows <- nrow(dummy.raster)
 8
 9     # Get the environment matrix
10     env.m <- getValues(dummy.raster, format = "matrix")
11     env.m[!is.na(env.m)] <- 1
12
13     # Select the number of patches
14     # n.patches <- 50
15     # Define the starting points for each patch
16     starts <- sample(x = which(!is.na(env.m)), size = n.patches, replace = FALSE)
17     # Select patch size
18     # patch.size <- sample(5:20, n.patches, replace = T)
19     ids <- vector(length = patch.size * n.patches)
20     elements <- matrix(1:(patch.size * n.patches), nrow = patch.size, ncol = n.patches)
21     for (run in 1:n.patches)
22     {
23       s <- starts[run]
24       # patch.s <- patch.size[run]
25       # Create the patch (same matrix as env.m, but with values of 2 where the patch exist)
26       patch <- expand(x = env.m, n.size = patch.size, start = s)
27       # Get patch cell ids
28
29       cur.ids <- which(patch == 2)
30       ids[elements[, run]] <- c(cur.ids,
31                         rep(NA, patch.size - length(cur.ids)))
32     }
33
34     # Attribute all patch values in the environment matrix (NA = sea, 0 = no patch, 1 = patch)
35     env.m[!(is.na(env.m))] <- 0
36     env.m[ids] <- 1
37
38     patch.raster <- setValues(dummy.raster,
39                   env.m)
40
41     return(patch.raster)
42   }
```