

# Plotting with `simPlot` in SpaDES

Eliot McIntire

Natural Resources Canada, Pacific Forestry Centre  
email: emcintir@nrcan.gc.ca

Alex M. Chubaty

Natural Resources Canada, Pacific Forestry Centre  
email: achubaty@nrcan.gc.ca

August 15, 2014

## Contents

1	Plotting with <code>simPlot</code>
---	------------------------------------

2
---

## 1 Plotting with simPlot

One of the major features of the SpaDES package is that it brings together many of R's powerful plotting and visualization into a small family of related functions. These new plotting functions make it easy to quickly produce visualizations useful at every stage of model development. Furthermore, conventional R plotting still works, so you can use the features provided in this package or you can use base plotting functions without having to relearn a completely new set of plotting commands.

```
> library(raster)
> library(RColorBrewer)
> # Give dimensions of dummy raster
> nx <- 1e2
> ny <- 1e2
> template <- raster(nrows=ny, ncols=nx, xmn=-nx/2, xmx=nx/2, ymn =-ny/2, ymx=ny/2)
> # Make dummy maps for testing of models:
> # - digital elevation model (DEM)
> # - forest age
> # - forest cover
> # - percent pine
> DEM <- round(GaussMap(template, scale=300, var=0.03, speedup=1), 1)*1000
> forestAge <- round(GaussMap(template, scale=10, var=0.1, speedup=1), 1)*20
> forestCover <- round(GaussMap(template, scale=50, var=1, speedup=1), 2)*10
> percentPine <- round(GaussMap(template, scale=50, var=1, speedup=1), 1)
> # Scale them as needed
> forestAge <- forestAge/maxValue(forestAge)*100
> percentPine <- percentPine/maxValue(percentPine)*100
> # Make layers that are derived from other layers
> habitatQuality <- (DEM+10 + (forestCover+5)*10)/100
> habitatQuality <- habitatQuality/maxValue(habitatQuality)
> # Stack them into a single stack for plotting
> habitat <- stack(list(DEM, forestAge, forestCover, habitatQuality, percentPine))
> names(habitat) <- c("DEM", "forestAge", "forestCover", "habitatQuality", "percentPine")
> cols <- list(
+   transparent.greys <- c("#00000000", paste(brewer.pal(8, "Greys"), "66", sep="")[8:1]),
+   grey <- brewer.pal(9, "Greys"),
+   spectral <- brewer.pal(8, "Spectral"),
+   terrain <- rev(terrain.colors(100)),
+   heat <- heat.colors(10),
+   topo <- topo.colors(10)
+ )
> simPlot(habitat, col=cols[c(2:5,3)])
```