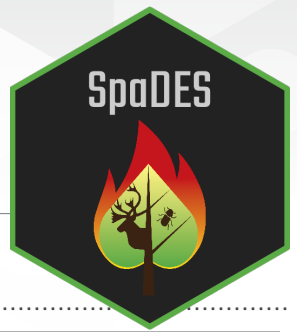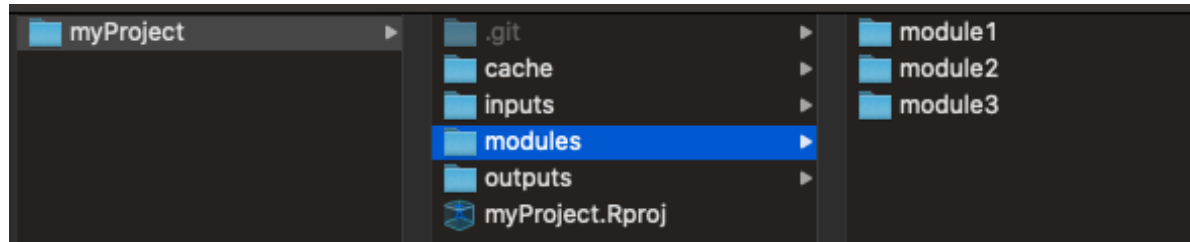# Simulation modelling in SpaDES :: CHEAT SHEET

Project setup

1. PROJECT DIRECTORY STRUCTURE

2. GET OR CREATE MODULES

```
downloadModule("module4", path = "modules")

copyModule("module4", "module5", "modules", open = TRUE)

newModule("module6", path = "modules")
```

3. WORKSPACE SETUP

```
setPaths(
    cachePath = "cache",          use relative paths
    inputPath = "inputs",
    modulePath = "modules",
    outputPath = "outputs"
)
```

Running simulations

```
mySim <- simInit(…)                 initialize a simulation

mySimOut <- spades(Copy(mySim), …)  run single simulation

myExpt <- experiment(mySim, …)      multiple simulations (in parallel)

restartSpades()                     if something goes wrong
```

Accessing the simList

Accessor	Description
globals	global (i.e., non-module-specific) parameters
params, P	module-specific parameters
inputs, outputs	module input and output objects
ls.objects, ls, str, obj	list objects stored in the simList environment
paths, cachePath, modulePath, inputPath, outputPath, dataPath	simulation paths
times, end, start, time	simulation times
events, current, completed	simulation events
modules	modules in use
packages	simulation and module package dependencies
depends	simulation module dependencies (advanced)
envir	the simList environment (advanced)

Module Examples

```
system.file("sampleModules", package = "SpaDES.core")

getOption("spades.modulesRepo")
```

Module Development

1. METADATA

2. DEFINE AND SCHEDULE EVENTS

```
scheduleEvent(sim, time(sim) + 10, "module5",
              "myEvent", .normal())

scheduleConditionalEvent(sim, …)
```

3. DEFAULT OBJECTS CREATED IN .inputObjects

PLOTTING (WITHIN MODULE)

```
Plot()        fast, modular plotting
clearPlot()
rePlot()
newPlot()
dev()         change plot device
```


PLOTTING (SIMULATION-LEVEL)

```
moduleDiagram(mySim, …)

objectDiagram(mySim, …)

eventDiagram(mySim, …)
```

Simulation caching

```
mySimOut <- Cache(spades(mySim), …)

showCache(…)

keepCache(…)
```

Package options

```
?SpaDES.core
```