MODULE FILES, METADATA AND INPUTS

Eliot McIntire, Ceres Barros , Ian Eddy

Research Scientists,

Pacific Forestry Service, Canadian Forest Service

SpaDES virtual workshop

June 11-13th, 2024

OUTLINE

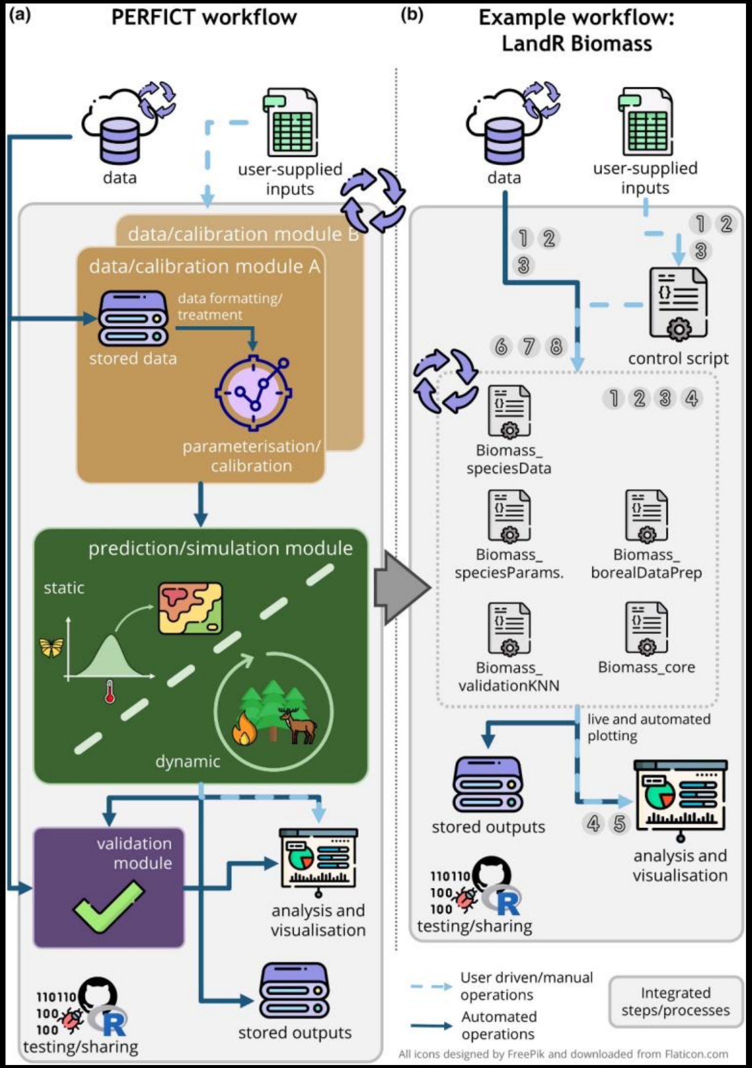
1. Module files
2. Module metadata
3. Types of module inputs

MODULE FILES

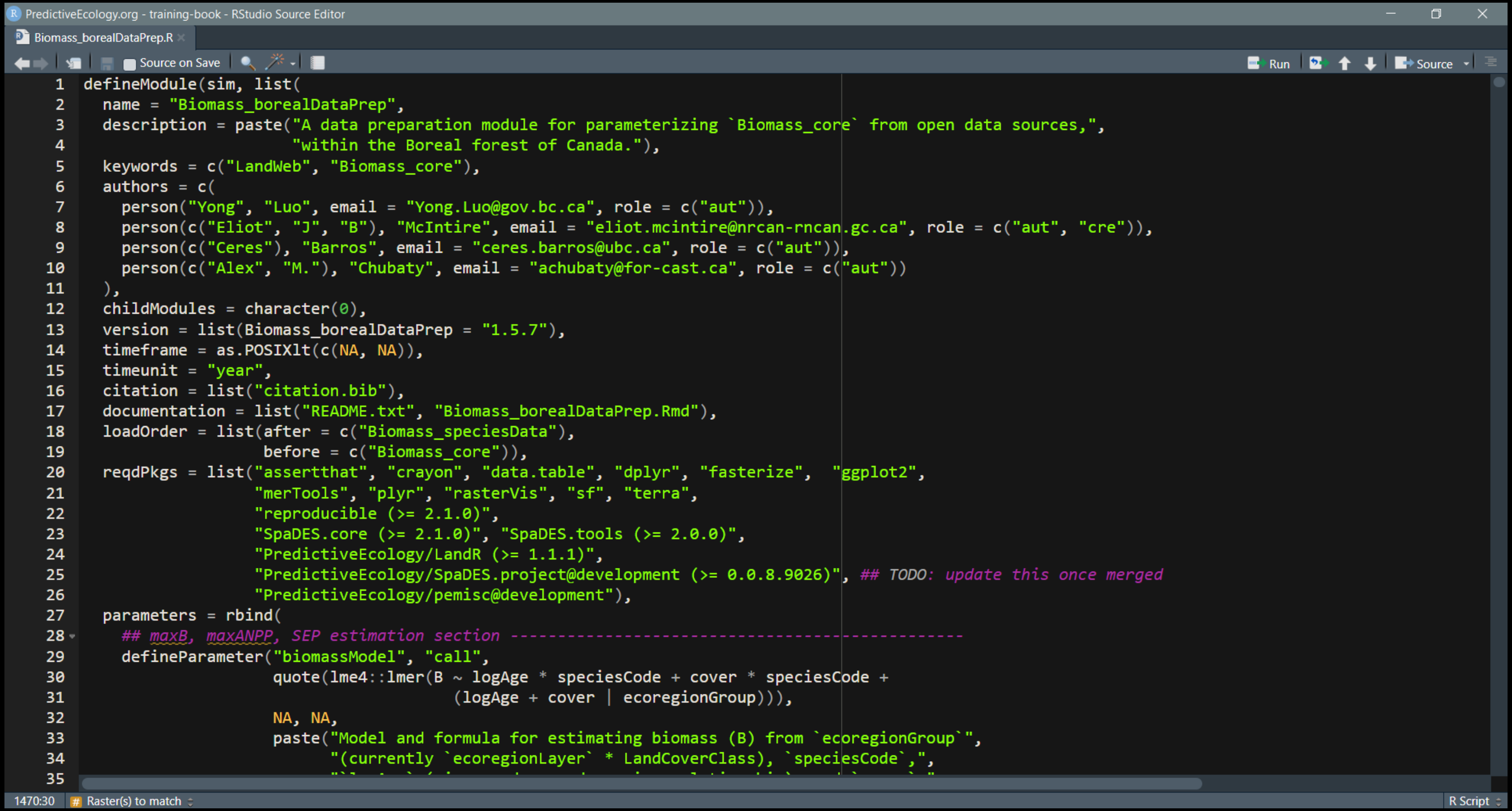
SpaDES sources each <module>.R script,
which contains the "instructions" for scheduling,
and any other .R scripts in the <module>/R/ folder

Both can contain functions used in events

```
/moduleRepository
  |_ moduleName/
      |_ R/                    # contains additional/optional .R (helper) files
      |_ data/                 # directory for all included data
          |_ CHECKSUMS.txt     # contains checksums for data files
      |_ tests/                # contains (optional) unit tests for module code
      |_ citation.bib          # bibtex citation for the module
      |_ LICENSE.txt           # describes module's legal usage
      |_ moduleName.R          # module code file (incl. metadata)
      |_ moduleName.Rmd        # documentation, usage info, etc.
```

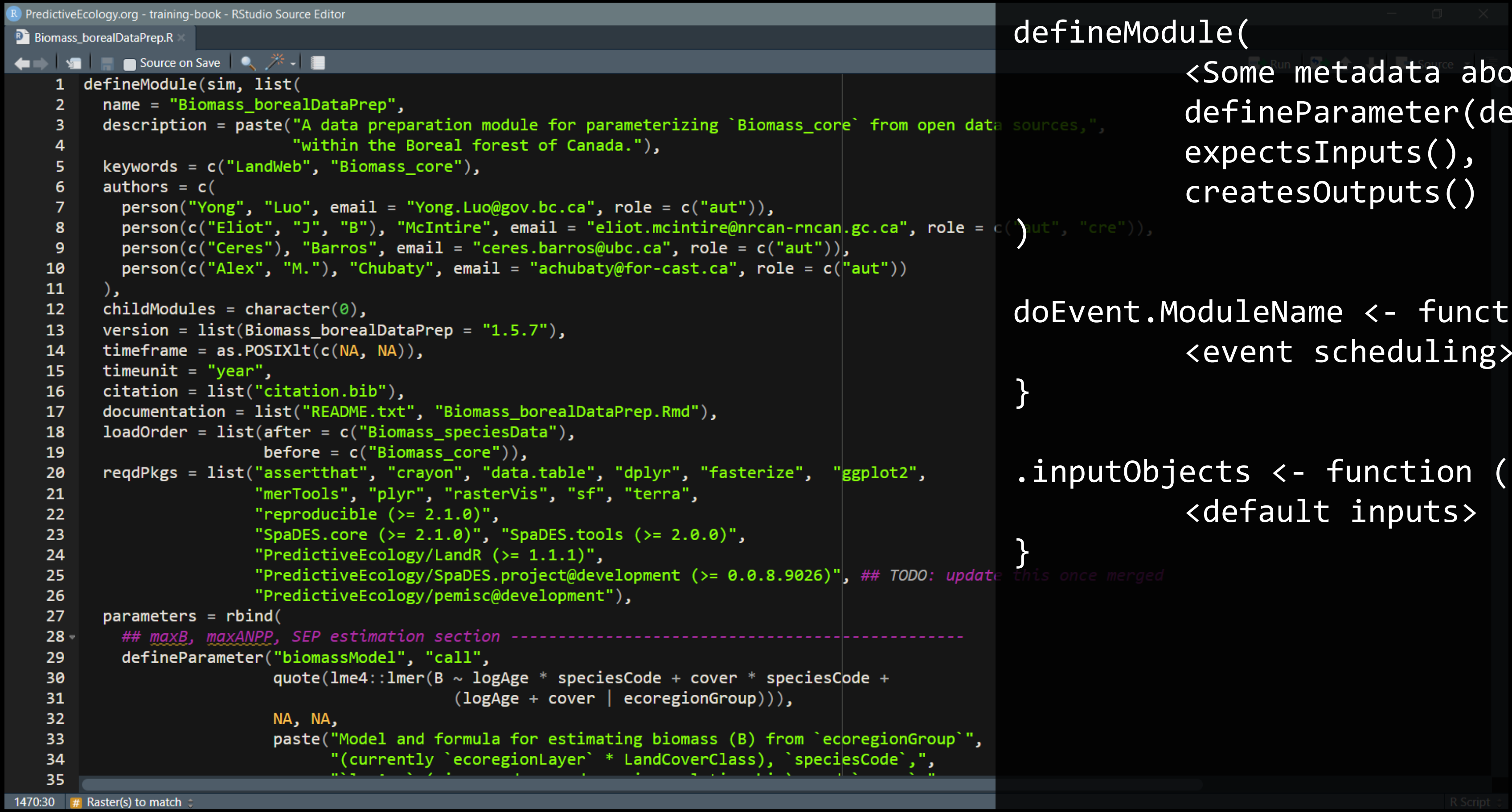

(a)
PERFICT workflow
(b)
Example workflow: LandR Biomass
data
user-supplied inputs
data/calibration module B
data/calibration module A
data formatting/ treatment
stored data
parameterisation/ calibration
prediction/simulation module
static
dynamic
validation module
analysis and visualisation
stored outputs
testing/sharing
control script
Biomass_ speciesData
Biomass_ speciesParams.
Biomass_ borealDataPrep
Biomass_ validationKNN
Biomass_core
live and automated plotting
User driven/manual operations
Automated operations
Integrated steps/processes
All icons designed by FreePik and downloaded from Flaticon.com

MODULE FILES - .R



```r
defineModule(sim, list(
  name = "Biomass_borealDataPrep",
  description = paste("A data preparation module for parameterizing `Biomass_core` from open data sources,",
                      "within the Boreal forest of Canada."),
  keywords = c("LandWeb", "Biomass_core"),
  authors = c(
    person("Yong", "Luo", email = "Yong.Luo@gov.bc.ca", role = c("aut")),
    person(c("Eliot", "J", "B"), "McIntire", email = "eliot.mcintire@nrcan-rncan.gc.ca", role = c("aut", "cre")),
    person(c("Ceres"), "Barros", email = "ceres.barros@ubc.ca", role = c("aut")),
    person(c("Alex", "M."), "Chubaty", email = "achubaty@for-cast.ca", role = c("aut"))
  ),
  childModules = character(0),
  version = list(Biomass_borealDataPrep = "1.5.7"),
  timeframe = as.POSIXlt(c(NA, NA)),
  timeunit = "year",
  citation = list("citation.bib"),
  documentation = list("README.txt", "Biomass_borealDataPrep.Rmd"),
  loadOrder = list(after = c("Biomass_speciesData"),
                   before = c("Biomass_core")),
  reqdPkgs = list("assertthat", "crayon", "data.table", "dplyr", "fasterize",  "ggplot2",
                  "merTools", "plyr", "rasterVis", "sf", "terra",
                  "reproducible (>= 2.1.0)",
                  "SpaDES.core (>= 2.1.0)", "SpaDES.tools (>= 2.0.0)",
                  "PredictiveEcology/LandR (>= 1.1.1)",
                  "PredictiveEcology/SpaDES.project@development (>= 0.0.8.9026)", ## TODO: update this once merged
                  "PredictiveEcology/pemisc@development"),
  parameters = rbind(
    ## maxB, maxANPP, SEP estimation section -----------------------------------------------
    defineParameter("biomassModel", "call",
                    quote(lme4::lmer(B ~ logAge * speciesCode + cover * speciesCode +
                                       (logAge + cover | ecoregionGroup))),
                    NA, NA,
                    paste("Model and formula for estimating biomass (B) from `ecoregionGroup`",
                          "(currently `ecoregionLayer` * LandCoverClass), `speciesCode`,",
```

MODULE FILES - .R



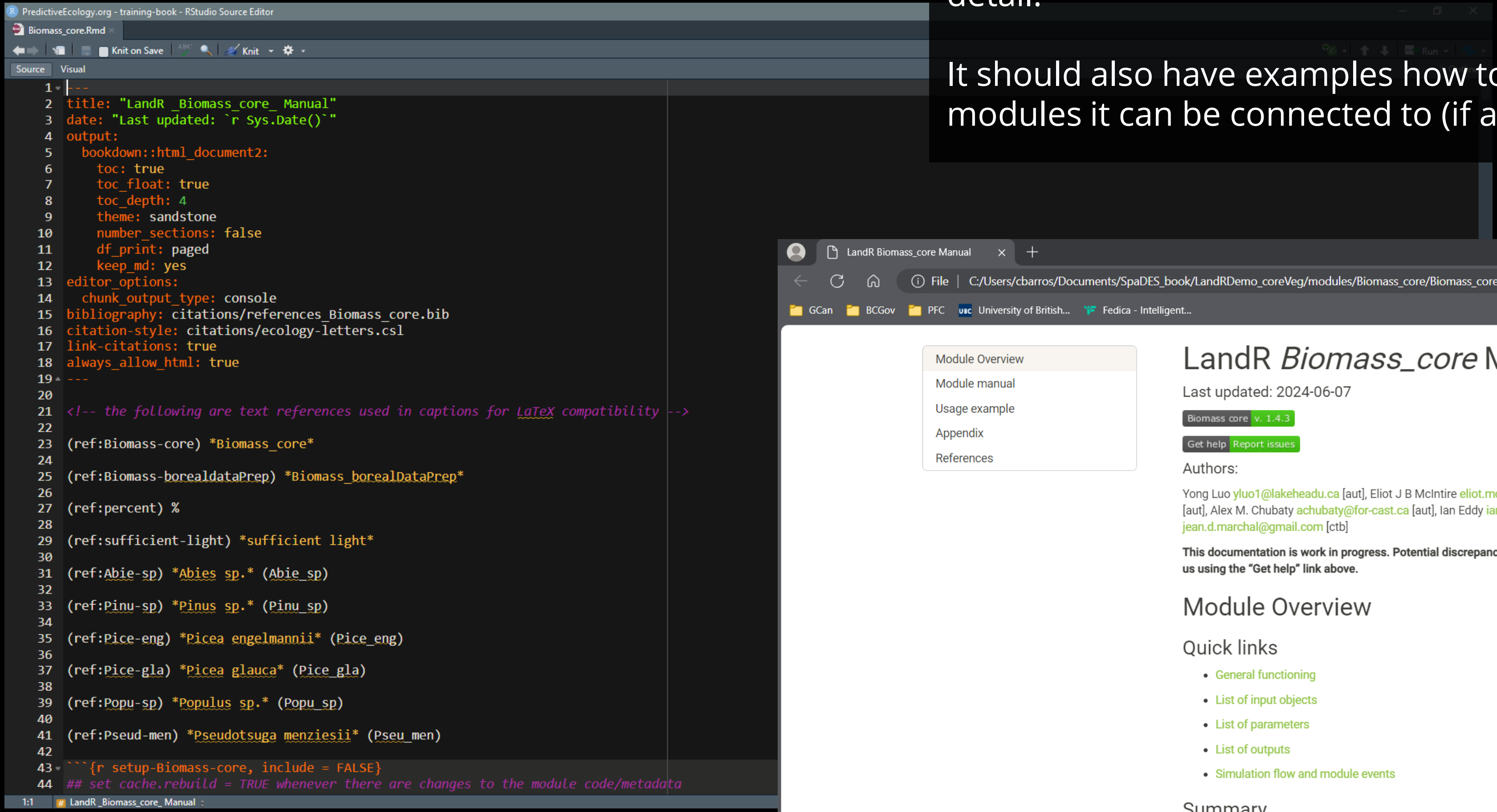
```r
defineModule(sim, list(
  name = "Biomass_borealDataPrep",
  description = paste("A data preparation module for parameterizing `Biomass_core` from open data sources,",
                      "within the Boreal forest of Canada."),
  keywords = c("LandWeb", "Biomass_core"),
  authors = c(
    person("Yong", "Luo", email = "Yong.Luo@gov.bc.ca", role = c("aut")),
    person(c("Eliot", "J", "B"), "McIntire", email = "eliot.mcintire@nrcan-rncan.gc.ca", role = c("aut", "cre")),
    person(c("Ceres"), "Barros", email = "ceres.barros@ubc.ca", role = c("aut")),
    person(c("Alex", "M."), "Chubaty", email = "achubaty@for-cast.ca", role = c("aut"))
  ),
  childModules = character(0),
  version = list(Biomass_borealDataPrep = "1.5.7"),
  timeframe = as.POSIXlt(c(NA, NA)),
  timeunit = "year",
  citation = list("citation.bib"),
  documentation = list("README.txt", "Biomass_borealDataPrep.Rmd"),
  loadOrder = list(after = c("Biomass_speciesData"),
                   before = c("Biomass_core")),
  reqdPkgs = list("assertthat", "crayon", "data.table", "dplyr", "fasterize",  "ggplot2",
                  "merTools", "plyr", "rasterVis", "sf", "terra",
                  "reproducible (>= 2.1.0)",
                  "SpaDES.core (>= 2.1.0)", "SpaDES.tools (>= 2.0.0)",
                  "PredictiveEcology/LandR (>= 1.1.1)",
                  "PredictiveEcology/SpaDES.project@development (>= 0.0.8.9026)", ## TODO: update this once merged
                  "PredictiveEcology/pemisc@development"),
  parameters = rbind(
    ## maxB, maxANPP, SEP estimation section -------------------------------------------------
    defineParameter("biomassModel", "call",
                    quote(lme4::lmer(B ~ logAge * speciesCode + cover * speciesCode +
                                       (logAge + cover | ecoregionGroup))),
                    NA, NA,
                    paste("Model and formula for estimating biomass (B) from `ecoregionGroup`",
                          "(currently `ecoregionLayer` * LandCoverClass), `speciesCode`,",
```

```r
defineModule(
        <Some metadata about module>,
        defineParameter(default parameter values),
        expectsInputs(),
        createsOutputs()
)

doEvent.ModuleName <- function (sim) {
        <event scheduling>
}

.inputObjects <- function (sim) {
        <default inputs>
}
```

MODULE FILES - .RMD

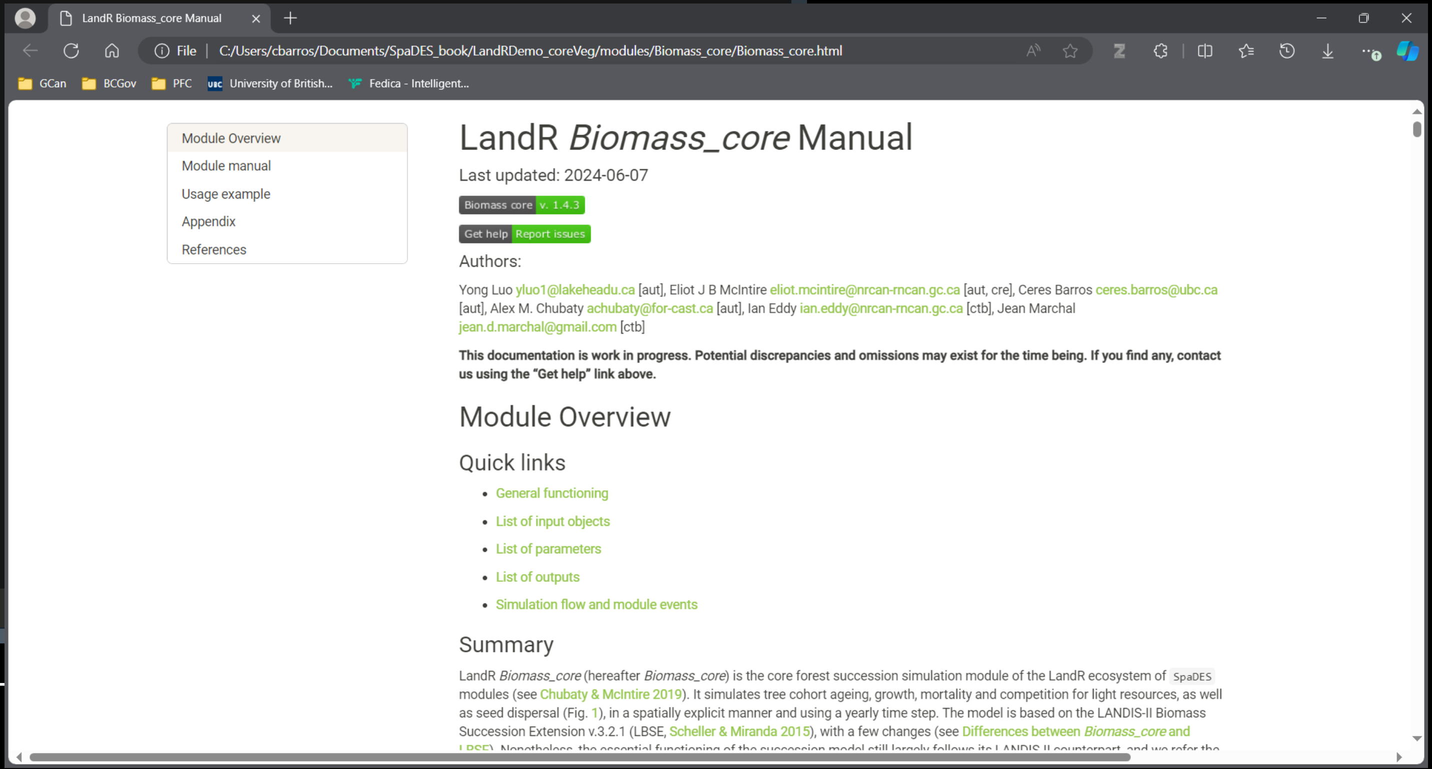
Should describe the module, its inputs, outputs and functioning in detail.

It should also have examples how to run the module and what other modules it can be connected to (if any).



PredictiveEcology.org - training-book - RStudio Source Editor
Biomass_core.Rmd
Knit on Save
Knit
Source
Visual
LandR_Biomass_core_ Manual

~~~
---
title: "LandR _Biomass_core_ Manual"
date: "Last updated: `r Sys.Date()`"
output:
  bookdown::html_document2:
    toc: true
    toc_float: true
    toc_depth: 4
    theme: sandstone
    number_sections: false
    df_print: paged
    keep_md: yes
editor_options:
  chunk_output_type: console
bibliography: citations/references_Biomass_core.bib
citation-style: citations/ecology-letters.csl
link-citations: true
always_allow_html: true
---

<!-- the following are text references used in captions for LaTeX compatibility -->

(ref:Biomass-core) *Biomass_core*

(ref:Biomass-borealdataPrep) *Biomass_borealDataPrep*

(ref:percent) %

(ref:sufficient-light) *sufficient light*

(ref:Abie-sp) *Abies sp.* (Abie_sp)

(ref:Pinu-sp) *Pinus sp.* (Pinu_sp)

(ref:Pice-eng) *Picea engelmannii* (Pice_eng)

(ref:Pice-gla) *Picea glauca* (Pice_gla)

(ref:Popu-sp) *Populus sp.* (Popu_sp)

(ref:Pseud-men) *Pseudotsuga menziesii* (Pseu_men)

```{r setup-Biomass-core, include = FALSE}
## set cache.rebuild = TRUE whenever there are changes to the module code/metadata
~~~



LandR *Biomass_core* Manual

Last updated: 2024-06-07

Authors:

Yong Luo yluo1@lakeheadu.ca [aut], Eliot J B McIntire eliot.mcintire@nrcan-rncan.gc.ca [aut, cre], Ceres Barros ceres.barros@ubc.ca [aut], Alex M. Chubaty achubaty@for-cast.ca [aut], Ian Eddy ian.eddy@nrcan-rncan.gc.ca [ctb], Jean Marchal jean.d.marchal@gmail.com [ctb]

This documentation is work in progress. Potential discrepancies and omissions may exist for the time being. If you find any, contact us using the "Get help" link above.

Module Overview

Quick links

- General functioning
- List of input objects
- List of parameters
- List of outputs
- Simulation flow and module events

Summary

LandR *Biomass_core* (hereafter *Biomass_core*) is the core forest succession simulation module of the LandR ecosystem of `SpaDES` modules (see Chubaty & McIntire 2019). It simulates tree cohort ageing, growth, mortality and competition for light resources, as well as seed dispersal (Fig. 1), in a spatially explicit manner and using a yearly time step. The model is based on the LANDIS-II Biomass Succession Extension v.3.2.1 (LBSE, Scheller & Miranda 2015), with a few changes (see Differences between *Biomass_core* and LBSE). Nonetheless, the essential functioning of the succession model still largely follows its LANDIS-II counterpart, and we refer the

MODULE METADATA

- Defined at the top of the module with `defineModule()`
- Includes:

?SpaDES.core::defineModule()

name	Module name. Must match the filename (without the .R extension). This is currently not parsed by SpaDES; it is for human readers only.
description	Brief description of the module. This is currently not parsed by SpaDES; it is for human readers only.
keywords	Author-supplied keywords. This is currently not parsed by SpaDES; it is for human readers only.
childModules	If this contains any character vector, then it will be treated as a parent module. If this is a parent module, then only this list entry will be read. (…)
authors	Module author information (as a vector of person() objects. This is currently not parsed by SpaDES; it is for human readers only.
version	Module version number (will be coerced to numeric_version() if a character or numeric are supplied). The module developer should update (…)
spatialExtent	The spatial extent of the module supplied via terra::ext. This is currently unimplemented. (…)
timeframe	Vector (length 2) of POSIXt dates specifying the temporal extent of the module. Currently unimplemented. (…)
timeunit	Time scale of the module (e.g., "day", "year"). If this is not specified, then .timeunitDefault() will be used. (…)
citation	List of character strings specifying module citation information. Alternatively, a list of filenames of .bib or similar files. (…)
documentation	List of filenames referring to module documentation sources. This is currently not parsed by SpaDES; it is for human readers only.
loadOrder	Named list of length 0, 1, or 2, with names being after and before. Each element should be a character string/vector naming 1 or more modules(…)
reqdPkgs	List of R package names required by the module. These packages will be loaded when simInit is called. Require::Require() will be used internally to (…)
parameters	A data.frame specifying the parameters used in the module. Usually produced by rbind-ing the outputs of multiple defineParameter() calls. These (…)
inputObjects	A data.frame specifying the data objects expected as inputs to the module, with columns objectName (class character), objectClass (class character), sourceURL (class character), and other (currently spades does (…)
outputObjects	A data.frame specifying the data objects output by the module, with columns identical to those in inputObjects. Like inputObjects above, (…0

MODULE INPUTS

?SpaDES.core::defineModule()

name	Module name. Must match the filename (without the .R extension). This is currently not parsed by SpaDES; it is for human readers only.
description	Brief description of the module. This is currently not parsed by SpaDES; it is for human readers only.
keywords	Author-supplied keywords. This is currently not parsed by SpaDES; it is for human readers only.
childModules	If this contains any character vector, then it will be treated as a parent module. If this is a parent module, then only this list entry will be read. (…)
authors	Module author information (as a vector of person() objects. This is currently not parsed by SpaDES; it is for human readers only.
version	Module version number (will be coerced to numeric_version() if a character or numeric are supplied). The module developer should update (…)
spatialExtent	The spatial extent of the module supplied via terra::ext. This is currently unimplemented. (…)
timeframe	Vector (length 2) of POSIXt dates specifying the temporal extent of the module. Currently unimplemented. (…)
timeunit	Time scale of the module (e.g., "day", "year"). If this is not specified, then .timeunitDefault() will be used. (…)
citation	List of character strings specifying module citation information. Alternatively, a list of filenames of .bib or similar files. (…)
documentation	List of filenames referring to module documentation sources. This is currently not parsed by SpaDES; it is for human readers only.
loadOrder	Named list of length 0, 1, or 2, with names being after and before. Each element should be a character string/vector naming 1 or more modules(…)
reqdPkgs	List of R package names required by the module. These packages will be loaded when simInit is called. Require::Require() will be used internally to (…)
parameters	A data.frame specifying the parameters used in the module. Usually produced by rbind-ing the outputs of multiple defineParameter() calls. These (…)
inputObjects	A data.frame specifying the data objects expected as inputs to the module, with columns objectName (class character), objectClass (class character), sourceURL (class character), and other (currently spades does (…)
outputObjects	A data.frame specifying the data objects output by the module, with columns identical to those in inputObjects. Like inputObjects above, (…0

MODULE INPUTS

Both *parameters* and *inputs* can be supplied/overridden by the user:

- Parameters are usually specific to a module (except for "globals") and supplied as lists for each module via `simInit(…, params)`
- Inputs are *shared* across modules and are supplied via `simInit(…, inputs)` or via `simInit(…, objects)` – usually more complex objects

Both can/should have defaults defined by the developer

- Parameter defaults are defined in the metadata (using `defineParameter()`)
- Input defaults are defined in the function `.inputObjects()`

TIME TO LOOK AT SOME CODE

[Robust and nimble scientific workflows, using SpaDES]

[Workshop Agenda]

This is a *hands-on* workshop

Please **ask questions**, share your troubles and successes

Last 15 min of each WOYO is for discussion

We are always available via **Teams chat**