StrathE2E Version 2.0.0

Mike Heath (m.heath@strath.ac.uk)

9th March 2018

Introductory Guide for the R StrathE2E2 package which provides access to the StrathE2E (Version 1.0.5) end-to-end ecosystem model developed at the University of Strathclyde by Professor Mike Heath.

StrathE2E is a marine food web model of intermediate complexity which represents a spatially aggregated shelf sea region. The model simulates the fluxes of nutrient (nitrogen) through the ecosystem from dissolved inorganic (nitrate and ammonia), through plankton, benthos and fish, to birds and mammals, its regeneration through excretion and mineralization of detritus in the water column and sediment, and the physical oceanographic, land-sea, and air-sea exchanges across the geographic boundaries.

Contents

1 Quick Start Guide

Currently the R package comes with one predefined model:

Listing 1: List available models

```
library(StrathE2E2)
list_models()
List of models in package folder:
         NorthSea
```

To load the predefined model and perform a default run do:

Listing 2: Run NorthSea model

```
model <- read_model("NorthSea")
results <- StrathE2E(model)
plot_final_year(model, results)
```

2 Accessing Help

For help, start with:

Listing 3: Accessing help

```
?StrathE2E
```

3 The Model Object

The model object has 3 list items:

Listing 4: Model Object

```
model <- read_model("NorthSea")
names(model)
[1] "path" "run"  "data"
```

3.1 path

A informational item which details where the model was read from.

Listing 5: path item

```
model <- read_model("NorthSea")
str(model$path)
 chr "/home/user11/Rlibs/StrathE2E2/libs/StrathE2E2/extdata/models/NorthSea"
```

3.2 run

This item controls the length of an individual model run and defines various time vectors:

Listing 6: run item

```
model <- read_model("NorthSea")
str(model$run)
List of 9
 $ nyears    : num 20
 $ ndays     : num 7201
 $ drndays   : num 240
 $ times     : num [1:7201] 0 1 2 3 4 5 6 7 8 9 ...
 $ drtimes   : num [1:240] 15 45 75 105 135 165 195 225 255 285 ...
 $ sprectimes: num [1:7201] 0 1 2 3 4 5 6 7 8 9 ...
 $ daynum    : num [1:361] 0 1 2 3 4 5 6 7 8 9 ...
 $ AAA       : num 1
 $ oudir     : chr "./results/"
```

The **nyears** value is used to construct all of the day and time values (**ndays**, **drndays**, **times**, etc). Result files are written by the model to the folder specified by the **oudir** value, and the **AAA** setting is used to number these files according to the users requirements.

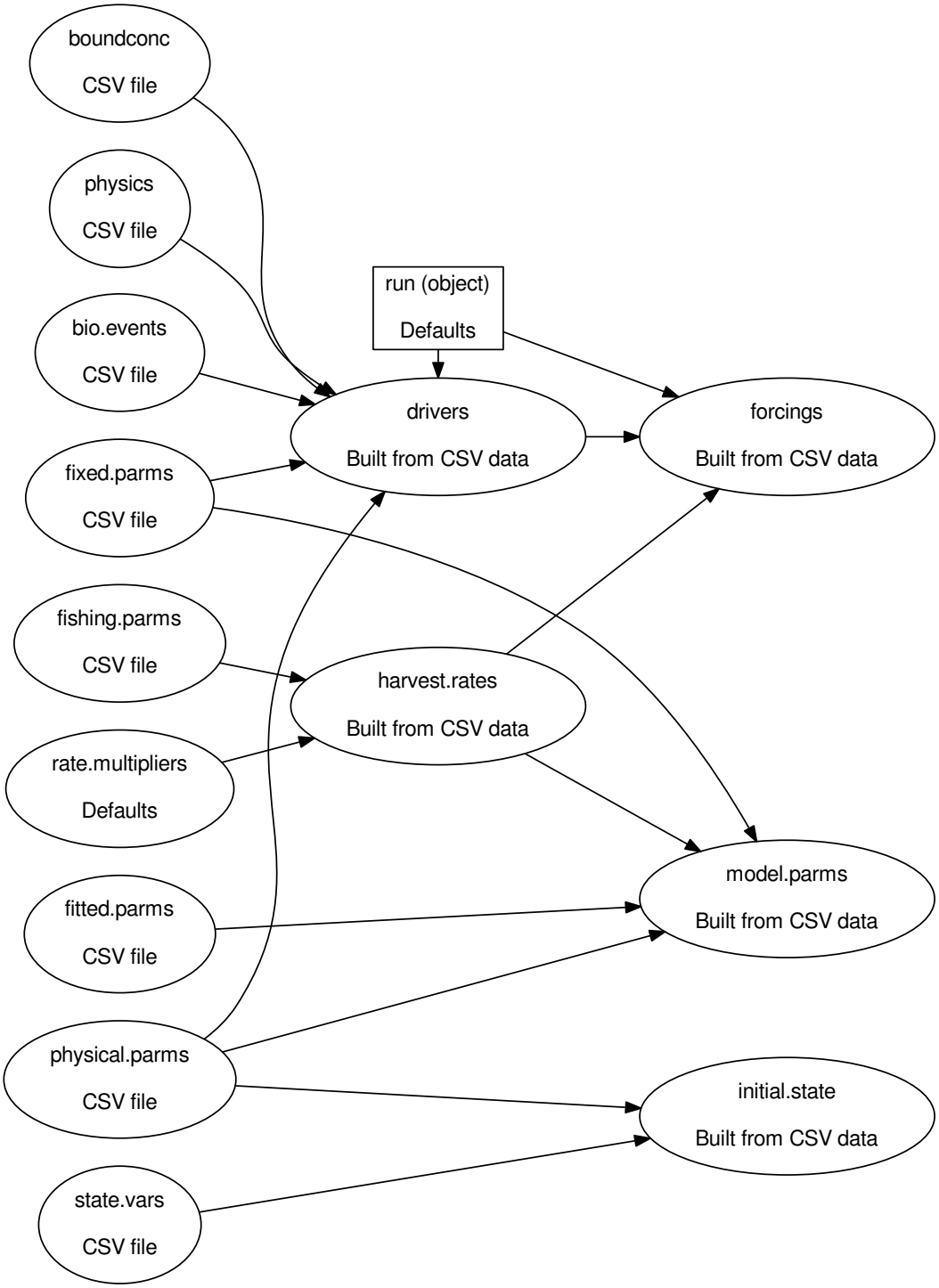
3.3 data

This item contains the main parameter sets, derived data, etc required for the model run.

Listing 7: data item

```
model <- read_model("NorthSea")
str(model$data, max.level=1)
List of 14
 $ rate.multipliers:List of 4
 $ fixed.parms     :List of 25
 $ fitted.parms    :'data.frame':       1 obs. of  76 variables:
 $ fishing.parms   :List of 9
 $ physical.parms  :List of 6
 $ state.vars      :List of 23
 $ physics         :'data.frame':       12 obs. of  11 variables:
 $ boundconc       :'data.frame':       12 obs. of  18 variables:
 $ bio.events      :List of 8
 $ initial.state   : Named num [1:93] 0.61363 1.16719 9.75664 0.00999 ...
 $ drivers         :List of 27
 $ harvest.rates   :List of 9
 $ model.parms     : Named num [1:120] 2.8e+01 4.2e+01 1.0e-01 4.5e-01 ...
 $ forcings        :List of 32
```

The graph shows which components of the **data** item are read in from CSV files and which are built from them:



4 Modifying The Model

It is possible to modify both the **run** and **data** sections of the model but some rules must be followed to make sure the model object remains consistent.

4.1 Modifying the run item

A default **run** item is setup when you call the **read_model()** function. You can use the **nyears** parameter (defaullt is 20 years) to change the length of the model run:

Listing 8: Modifying length of model run to 40 years

```
model <- read_model("NorthSea", nyears=40)
results <- StrathE2E(model)
```

The **run** slots **oudir** (results folder) and **AAA** (output file numbering suffix) can be modified before calling **StrathE2E**:

Listing 9: Modifying outdir/AAA

```
model <- read_model("NorthSea")
model$run$AAA <- 100
model$run$oudir <- "run100/"
results <- StrathE2E(model)
```

4.2 Modifying the data item

It is possible to modify the model data in 2 ways: modify the parameters in the CSV parameter files, or modify the relevant sections of the data item.

4.2.1 Modifying CSV files

You cannot (or should not) modify the CSV parameter files built in to the package. To change the CSV parameter files, first make a copy of the desired model:

Listing 10: Copying a package model

```
copy_model("NorthSea", "mymodels", overwrite=TRUE)
```

This copies the **NorthSea** model to the local folder **mymodels**, creating the folder if necessary and overwriting the existing copy if desired.

The user can then edit some of the CSV files with the **mymodels/NorthSea** folders and then instruct **read_model()** to read this model rather than the package version by using the **model.path** parameter:

Listing 11: Loading a modified model

```
mymodel <- read_model("NorthSea", model.path="mymodels")
results <- StrathE2E(mymodel)
```

You may want to rename the local model to a different name (by renaming the model folder) to prevent confusion with the package model.

4.2.2 Modifying data item directly

If you want to explore the effect of changing a particular parameter on the model outcome, you can do this directly by modifying the model data component.

From the graph we can see that modifying a model data component such as **rate.multipliers** will leave the downstream components **harvest.rates**, **model.parms** and **forcings** in an inconsistent state.

Therefore, you must rebuild the model data component once you have modified it:

Listing 12: Modifying model data directly

```
model <- read_model("NorthSea")
model$data$rate.multipliers$DFratemult <- 2.0
model$data$rate.multipliers$PFratemult <- 2.0
model <- rebuild_model(model)
results <- StrathE2E(model)
```

of course you could put this in a loop to explore a parameter range.

5 Model Output

The StrathE2E model produces a comprehensive set of outputs:

Component	Description	Units
time	description	units
s_detritus	description	units
d_detritus	description	units
x_detritus	description	units
discard	description	units
corpse	description	units
s_ammonia	description	units
d_ammonia	description	units
x_ammonia	description	units
s_nitrate	description	units
d_nitrate	description	units
x_nitrate	description	units
s_phyt	description	units
d_phyt	description	units
herb	description	units
carn	description	units
benths	description	units
benthc	description	units
fishp	description	units
fishplar	description	units
fishd	description	units
fishdlar	description	units
bird	description	units
totpprod	description	units
totnitup	description	units
totammup	description	units
phytprod	description	units
herbprod	description	units
carnprod	description	units
benthsprod	description	units
benthcprod	description	units
fishplarprod	description	units
fishpprod	description	units
fishdlarprod	description	units
fishdprod	description	units
birdprod	description	units
catchp	description	units
catchd	description	units
catchsb	description	units
catchcb	description	units
wcmineral	description	units
sedmineral	description	units
wcnitrif	description	units
sednitrif	description	units
wcdenitrif	description	units
seddenitrif	description	units
wcammprod	description	units
sedammprod	description	units
wcnitprod	description	units
sednitprod	description	units

fluxherb_carn	description	units
fluxherb_pfishlar	description	units
fluxherb_dfishlar	description	units
fluxherb_pfish	description	units
fluxcarn_pfish	description	units
fluxcarn_dfish	description	units
fluxpfishlar_carn	description	units
fluxpfishlar_pfish	description	units
fluxpfishlar_dfish	description	units
fluxpfish_dfish	description	units
fluxpfish_bird	description	units
fluxbens_dfish	description	units
fluxbens_benc	description	units
fluxbenc_dfish	description	units
fluxdfishlar_carn	description	units
fluxdfishlar_pfish	description	units
fluxdfishlar_dfish	description	units
fluxdfish_dfish	description	units
fluxdfish_bird	description	units
fluxdisc_dfish	description	units
fluxdisc_bird	description	units
fluxcorp_benthc	description	units
fluxcorp_fishd	description	units
fluxcorp_bird	description	units
fluxDINinflow	description	units
fluxDINoutflow	description	units
fluxPARTinflow	description	units
fluxPARToutflow	description	units
atmosDINinput	description	units
rivDINinflow	description	units
rivPARTinflow	description	units
vertnitflux	description	units
horiznitflux	description	units
PNP	description	units
Pfish_spawn	description	units
Pfish_recruit	description	units
Dfish_spawn	description	units
Dfish_recruit	description	units
discpel	description	units
discdem	description	units
discsfb	description	units
disccfb	description	units
fluxpartwc_sed	description	units