

Kit de gestion des forêts sèches

Annexe : Script du programme

```
library(shiny)
library(raster)
library(rgdal)
library(sp)
library(maptools)
library(tcltk)
rm(list=ls())
ui <- fluidPage(
  titlePanel(«KIT POUR LES GESTIONNAIRES DES FORETS SECHES»),
  #fileInput(«file_input»,»Choose your file»),
  sidebarPanel(
    strong(h4(«DONNEES», style = «color:red»)),
    numericInput(«N», «Nombre d'annee d'exploitation (N): «, value = 1),
    numericInput(«DC», «Duree du cycle d'exploitation (DC):», value = 0),
    numericInput(«PropUB», «Proportion usage biomasse (charbon):», value = 0.5, min=0, max=1),
    #numericInput(«PropUB_BF», «Proportion usage biomasse (bois de feu):», value = 0.5, min=0,
    max=1),
    #numericInput(«PropUB_BO», «Proportion usage biomasse (bois d'oeuvre):», value = 0, min=0,
    max=1),
    numericInput(«PropT», «Proportion d'exploitation par type de foret (xerophile) : «, value=0.5,
    max=1),
    strong(h4(«PRELEVEMENT», style = «color:red»)),
    numericInput(«PrelC», «Prelevement de charbon en tonne/an (PrC) :», value = 0),
    numericInput(«PrelBF», «Prelevement de bois de feu en tonne/an (PrBF) :», value = 0),
    tags$style(«body{background-color:white ; color:black»),
    img(src = «image.png», height = 70, width = 450)
  ),
  mainPanel(
    navbarPage(title = «BIOMASSE»,
    tabPanel(title=»PRODUCTION TOTALE»,
    h5(«SURFACES : «, style = «color:black»),
    h5(textOutput(«STG_txtOutput»), textOutput(«S_txtOutput»)),
    h5(textOutput(«S_N_txtOutput»),textOutput(«SFXD_N_txtOutput2»)),
    h5(«EN SITUATION D'EQUILIBRE, LA QUANTITE A NE PAS DEPASSER : «, style = «color:black»),
    h5(textOutput(«T_txtOutput»), textOutput («TCHBF_txtOutput»)),
    h5(«PRELEVEMENT TOTAL : «, style = «color:black»),
    h5(textOutput(«T_txtOutput2»), textOutput(«TFXD_txtOutput2»)),
    h5(«BILAN : «, style = «color:black»),
    h5(textOutput(«T_txtOutput1»)),
    h5(textOutput(«TFX_txtOutput1»)),
    h5(textOutput(«TFD_txtOutput1»))
    ),
    tabPanel(title=»FOURRE XEROPHILE»,
    h5(«SURFACES : «, style = «color:black»),
    h5(textOutput(«SFX_txtOutput»), textOutput(«SFX_N_txtOutput»)), 191 190

    h5(«EN SITUATION D'EQUILIBRE, LA QUANTITE A NE PAS DEPASSER : «, style = «color:black»),
    h5(textOutput(«FX_txtOutput»), textOutput («CHBFFX_txtOutput»)),
    h5(«PRELEVEMENT : «, style = «color:black»),
```

```

h5(textOutput(«FX_txtOutput2»)),
h5(«BILAN : », style = «color:black»),
h5(textOutput(«FX_txtOutput1»)),
strong(h5(textOutput(«FX_txtOutput3»), style=»color:black»))
),
tabpanel(title=»FOURRE DEGRADE«,
h5(«SURFACES : », style = «color:black»),
h5(textOutput(«SFD_txtOutput»), textOutput(«SFD_N_txtOutput»)),
h5(«EN SITUATION D'EQUILIBRE, LA QUANTITE A NE PAS DEPASSER : », style = «color:black»),
h5(textOutput(«FD_txtOutput»), textOutput («CHBFFD_txtOutput»)),
h5(«PRELEVEMENT : », style = «color:black»),
h5(textOutput(«FD_txtOutput2»)),
h5(«BILAN : », style = «color:black»),
h5(textOutput(«FD_txtOutput1»)),
strong(h5(textOutput(«FD_txtOutput3»), style=»color:black»))
)
),
navbarPage(title = «PRODUITS»,
tabpanel(title=»CHARBON«,
h5(«PRODUCTION LIMITE : », style=»color:black»),
h5(textOutput(«CH1_txtOutput»), textOutput(«CH2_txtOutput»)),
h5(«PRODUCTION EFFECTIVE : », style=»color:black»),
h5(textOutput(«CH_txtOutput2»)),
h5(«EXCES DE PRODUCTION : », style=»color:black»),
h5(textOutput(«CH_txtOutput1»))
),
tabpanel(title=»BOIS DE FEU«,
h5(«PRODUCTION LIMITE : », style=»color:black»),
h5(textOutput(«BF1_txtOutput»), textOutput(«BF2_txtOutput»)),
h5(«PRODUCTION EFFECTIVE : », style=»color:black»),
h5(textOutput(«BF_txtOutput2»)),
h5(«EXCES DE PRODUCTION : », style=»color:black»),
h5(textOutput(«BF_txtOutput1»))
)
)
)
)
# Define server
server <- function(input, output) {
#selection de la limite TG
Data_OSFX<- raster('FX.tif')
mySHP<-tk_choose.files()
TG<-readOGR(mySHP)
TG<-spTransform(TG, crs(Data_OSFX))
crop_extent <- readOGR(mySHP)
crop_extent<-spTransform(crop_extent, crs(Data_OSFX))
#PRELEVEMENT
#proportion prelevement (charbon/bois de feu)
prop_CH<-reactive(input$PrelC/(input$PrelC+input$PrelBF))
prop_BF<-reactive(input$PrelBF/(input$PrelC+input$PrelBF))
#Prelevement total sans ponderation en biomasse
Prel_C <- reactive(input$PrelC/(.125*.45))

```

```

Prel_BF <- reactive((input$PrelBF/(1*.45)))
#quantite de biomasse prelevee dans le fourre xerophile
Prel_FX_C <- reactive(Prel_C()*input$PropT)
Prel_FX_BF<-reactive(Prel_BF()*input$PropT)
Prel_FX<-reactive(Prel_FX_C()+Prel_FX_BF())
#quantite prelevee de biomasse en fourre degrade
Prel_FD_C <- reactive(Prel_C()*(1-input$PropT))
Prel_FD_BF<-reactive(Prel_BF()*(1-input$PropT))
Prel_FD<-reactive(Prel_FD_C()+Prel_FD_BF())
#quantite prelevee de biomasse totale
Prel<-reactive(Prel_FX()+Prel_FD())
#surface necessaire en FX
SFX_N_C<-reactive(Prel_FX_C()/(.75*((input$DC/input$N)+1)))
SFX_N_BF<-reactive(Prel_FX_BF()/(.75*((input$DC/input$N)+1)))
SFX_N<-reactive(SFX_N_C()+SFX_N_BF())
#surface necessaire en FD
SFD_N_C<-reactive(Prel_FD_C()/(.258*((input$DC/input$N)+1)))
SFD_N_BF<-reactive(Prel_FD_BF()/(.258*((input$DC/input$N)+1)))
SFD_N<-reactive(SFD_N_C()+SFD_N_BF())
#*****
#PRODUCTIVITE
#FOURRE XEROPHILE
#production de fourre xerophile
Data_OSFX<- raster('FX.tif')
Data_OSFX2<-crop(Data_OSFX, crop_extent)
#surface TG
TG_OS<-rasterize(TG, Data_OSFX2)
P<-freq(TG_OS, value=1)
S_TG<-P*30*30/10000
#surface FX
PFX<-zonal(Data_OSFX2, TG_OS, 'sum')
SFX<-PFX[2]*30*30/10000
#production en charbon et bois de feu en fourre xerophile (sans ponderation) 193 192

ProdCH_FX<-reactive(SFX*.75*((input$DC/input$N)+1)*.125*.45)
ProdBF_FX<-reactive(SFX*.75*((input$DC/input$N)+1)*1*.45)
#production en charbon et bois de feu en fourre xerophile (avec ponderation)
ProdCH_FX_P <- reactive(ProdCH_FX()*prop_CH())
ProdBF_FX_P <- reactive(ProdBF_FX()*prop_BF())
#production de biomasse fourre xerophile
ProdBM_FX<-reactive((SFX*.75*((input$DC/input$N)+1)))
#FoURRE DEGRADE
#production de fourre degrade
Data_OSFD<- raster('FD.tif')
Data_OSFD2<-crop(Data_OSFD, crop_extent)
#surface FD
PFD<-zonal(Data_OSFD2, TG_OS, 'sum')
SFD<-PFD[2]*30*30/10000
#production en charbon et bois de feu en fourre degrade (sans ponderation)
ProdCH_FD<-reactive(SFD*.258*((input$DC/input$N)+1)*.125*.45)
ProdBF_FD<-reactive(SFD*.258*((input$DC/input$N)+1)*1*.45)
#production en charbon et bois de feu en fourre degrade (avec ponderation)

```

```

ProdCH_FD_P <- reactive(ProdCH_FD()*prop_CH())
ProdBF_FD_P <- reactive(ProdBF_FD()*prop_BF())
#production de biomasse fourree degrade
ProdBM_FD<-reactive((SFD*.258*((input$DC/input$N)+1)))
#production totale par type de produit
ProdCH<-reactive(ProdCH_FX_P()+ProdCH_FD_P())
ProdBF<-reactive(ProdBF_FX_P()+ProdBF_FD_P())
#production totale
ProdBM<-reactive(ProdBM_FX()+ProdBM_FD())
#*****
#perte en biomasse annuelle
#en fourre xerophile
PrS_FX<-reactive(Prel_FX()-ProdBM_FX())
#en fourre degrade
PrS_FD<-reactive(Prel_FD()-ProdBM_FD())
#en tous type de foret
PrS<-reactive(PrS_FX()+PrS_FD())
#*****
#duree de tarissement
DT_FX<-reactive((51.973*SFX)/PrS_FX())
DT_FD<-reactive((16.843*SFD)/PrS_FD())
DT_T<-reactive(((51.973*SFX)+(16.843*SFD))/(PrS()))
#OUTPUTS
#Output surface TG, FX et FD
output$STG_txtOutput= renderText({
paste0(«La superficie totale du TG est de : », round(S_TG, 2), « ha»)
})
output$S_txtOutput = renderText({
paste0(«La superficie de foret exploitable totale est de : », round(SFX+SFD, 2), « ha»)
})
output$SFX_txtOutput = renderText({
paste0(«La superficie du fourre xerophile est de : », round(SFX, 2), « ha»)
})
output$SFD_txtOutput = renderText({
paste0(«La superficie du fourre degrade est de : », round(SFD, 2), « ha»)
})
output$S_N_txtOutput = renderText({
paste0(«La superficie de foret necessaire totale est de : », round(SFX_N()+SFD_N(), 2), « ha»)
})
output$SFX_N_txtOutput = renderText({
paste0(«La superficie de foret necessaire en fourre xerophile est de : », round((SFX_N()), 2), « ha»)
})
output$SFXD_N_txtOutput2 = renderText({
paste0(« dont », round((SFX_N()), 2), « ha pour la fourre xerophile et », round(SFD_N(), 2), « ha pour
la fourre degrade»)
})
output$SFD_N_txtOutput = renderText({
paste0(«La superficie de foret necessaire en fourre degrade est de : », round(SFD_N(), 2), « ha»)
})
# output$SFD_N_txtOutput2 = renderText({
# paste0(«et », round(SFD_N(), 2), « ha pour la fourre degrade»)
# })

```

```

#Output fourre xerophile
output$FX_txtOutput = renderText({
paste0(«La quantite de biomasse est de : », round((ProdBM_FX()), 2), « tonne/an»)
})
output$FX_txtOutput1 = renderText({
if ((Prel_FX())< ProdBM_FX())
paste0(«La quantite de biomasse restante est de : », round(ProdBM_FX()-(Prel_FX()),2), « tonne/an»)
else
paste0(«Le nombre d'annee restant pour l'exploitation (DT) est de : », round(DT_FX(),2))
})
output$FX_txtOutput2 = renderText({
paste0(«La quantite de biomasse convertie en produit est de : », round(Prel_FX(),2), « tonne/an»)
})
output$FX_txtOutput3 = renderText({
if ((Prel_FX())< ProdBM_FX())
paste0(«EXPLOITATION DURABLE»)
else
paste0(«EXPLOITATION NON DURABLE»)
}) 195 194

output$FX_txtOutput31 = renderText({
if ((Prel_FX())< ProdBM_FX())
paste0(«EXPLOITATION DURABLE»)
else
paste0(«EXPLOITATION NON DURABLE»)
})
#Output fourre degrade
output$FD_txtOutput = renderText({
paste0(«La quantite de biomasse est de : », round((ProdBM_FD()),2), « tonne/an»)
})
output$FD_txtOutput1 = renderText({
if ((Prel_FD())< ProdBM_FD())
paste0(«La quantite de biomasse restante est de : », round(ProdBM_FD()-(Prel_FD()),2), «
tonne/an»)
else
paste0(«Le nombre d'annee restant pour l'exploitation (DT) est de : », round(DT_FD(),2))
})
output$FD_txtOutput2 = renderText({
paste0(«La quantite de biomasse convertie en produit est de : », round(Prel_FD(),2), « tonne/an»)
})
output$FD_txtOutput3 = renderText({
if ((Prel_FD())< ProdBM_FD())
paste0(«EXPLOITATION DURABLE»)
else
paste0(«EXPLOITATION NON DURABLE»)
})
output$FD_txtOutput31 = renderText({
if ((Prel_FD())< ProdBM_FD())
paste0(«EXPLOITATION DURABLE»)
else
paste0(«EXPLOITATION NON DURABLE»)
})

```

```

#Output total
output$T_txtOutput = renderText({
paste0(«La quantite de biomasse totale est de : », round((ProdBM()),2), « tonne/an» )
})
output$TFX_txtOutput1 = renderText({
if ((Prel_FX())<ProdBM_FX())
paste0(«La quantite de biomasse en fourre xerophile restante est de : », round(ProdBM_FX()-
(Prel_FX()), 2), « tonne/an»)
else
paste0(«Le nombre d’annee restant pour l’exploitation du fourree xerophile (DT) est de : »,
round(DT_FX(), 2))
})
output$TFD_txtOutput1 = renderText({
if ((Prel_FD())<ProdBM_FD())
paste0(«La quantite de biomasse en fourre degrade restante est de : », round(ProdBM_FD()-
(Prel_FD()), 2), « tonne/an»)
else
paste0(«Le nombre d’annee restant pour l’exploitation du fourre degrade (DT) est de : »,
round(DT_FD(), 2))
})
output$T_txtOutput1 = renderText({
if ((Prel())<ProdBM())
paste0(«La quantite de biomasse totale restante est de : », round(ProdBM()-Prel(), 2), « tonne/an»)
else
paste0(«Le nombre d’annee restant pour l’exploitation totale (DT) est de : », round(DT_T(), 2))
})
output$T_txtOutput2 = renderText({
paste0(«La quantite de biomasse convertie en produit est de : », round(Prel(),2), « tonne/an»)
})
output$TFXD_txtOutput2 = renderText({
paste0(«dont », round(Prel_FX(),2), « tonne/an en fourre xerophile et », round(Prel_FD(),2), «
tonne/an en fourre degrade»)
})
# output$TFD_txtOutput2 = renderText({
# paste0(round(Prel_FD(),2), « tonne/an en fourre degrade»)
# })
output$TCHBF_txtOutput = renderText({
paste0(«soit pour une quantite de charbon de »,
round((ProdCH_FX()+ProdCH_FD())*input$PropUB,2), « tonne/an et une quantite de bois de feu de
», round((ProdBF_FX()+ProdBF_FD())*(1-input$PropUB),2), « tonne/an»)
})
# output$TBF_txtOutput = renderText({
# paste0(«et une quantite de bois de feu de : », round((ProdBF_FX()+ProdBF_FD())*(1-
input$PropUB),2), « tonne/an»)
# })
output$T_txtOutput3 = renderText({
if ((Prel())<ProdBM())
paste0(«EXPLOITATION GLOBALEMENT DURABLE»)
else
paste0(«EXPLOITATION NON DURABLE EN PARTIE OU EN TOTALITE»)
})
#production en charbon

```

```

output$CH1_txtOutput = renderText({paste0(«La quantité de charbon pouvant être produite sans
degradation est de : », round(ProdCH(),2), « tonne/an»)})
output$CH2_txtOutput = renderText({paste0(« dont », round(ProdCH_FX_P(), 2), « tonne/an en
fourre xerophile et », round(ProdCH_FD_P(), 2), « tonne/an en fourre degrade»)})
#output$CH3_txtOutput = renderText({paste0(round(ProdCH_FD_P(), 2), « tonne/an en fourre
degrade»)})
output$CH_txtOutput1 = renderText({
paste0(«La production supplémentaire de charbon est de : », round((ProdCH()-input$PrelC),2), «
tonne/an»)
})
output$CH_txtOutput2 = renderText({
paste0(«La quantité produite en charbon est de : », round(input$PrelC,2), « tonne/an»)
})
output$CHBFFX_txtOutput = renderText({paste0(«soit une quantité de charbon de »,
round(ProdCH_FX()*input$PropUB,2), « tonne/an et une quantité de bois de feu de »,
round(ProdBF_FX()*(1-input$PropUB),2), « tonne/an»)
})
output$CHBFFD_txtOutput = renderText({paste0(«soit une quantité de charbon de »,
round(ProdCH_FD()*input$PropUB,2), « tonne/an et une quantité de bois de feu de »,
round(ProdBF_FD()*(1-input$PropUB),2), « tonne/an»)})
#production en bois de feu 197 196

output$BF1_txtOutput = renderText({
paste0(«La quantité de bois de feu pouvant être produite sans degradation est de : »,
round(ProdBF(),2), « tonne/an»)
})
output$BF2_txtOutput = renderText({paste0(« dont », round(ProdBF_FX_P(), 2), « tonne/an en
fourre xerophile et », round(ProdBF_FD_P(), 2), « tonne/an en fourre degrade»)})
#output$BF3_txtOutput = renderText({paste0(round(ProdBF_FD_P(), 2), « tonne/an en fourre
degrade»)})
output$BF_txtOutput1 = renderText({
paste0(«La production supplémentaire de bois de feu est de : », round((ProdBF()-input$PrelBF),2), «
tonne/an»)
})
output$BF_txtOutput2 = renderText({
paste0(«La quantité produite en bois de feu est de : », round(input$PrelBF,2), « tonne/an»)
})
#output$BFFX_txtOutput = renderText({paste0(«et une quantité de bois de feu de : »,
round(ProdBF_FX()*(1-input$PropUB),2), « tonne/an»)})
#output$BFFD_txtOutput = renderText({paste0(«et une quantité de bois de feu de : »,
round(ProdBF_FD()*(1-input$PropUB),2), « tonne/an»)})
#valeur de la production en charbon
output$VCH_txtOutput = renderText({
paste0(«La valeur de la production en charbon est de : », round(input$PrelC*input$PrixCH), «
Ariary/an»)
})
#valeur de la production en bois de feu
output$VBF_txtOutput = renderText({
paste0(«La valeur de la production en bois de feu est de : », round(input$PrelBF*input$PrixBF),
«Ariary/ an»)
})
#valeur totale de la production

```

```
output$VT_txtOutput = renderText({
  paste0(«La valeur totale de la production est de : »,
  round((input$PreIBF*input$PrixBF)+(input$PreIC*input$PrixCH)), «Ariary/an»)
})
}
# Run the application
shinyApp(ui = ui, server = server)
```