



Galaxy-Bricks

Vers une plateforme d'analyse de données collaborative

Simon Bénateau¹, Sébastien Turpin¹, Yvan Le Bras²

Muséum National d'Histoire Naturelle

1. UMR Centre d'Ecologie et des Sciences de la CONservation
2. UMS PATRImoine NATurel

15 mai 2019



European
Commission

Horizon 2020
European Union funding
for Research & Innovation

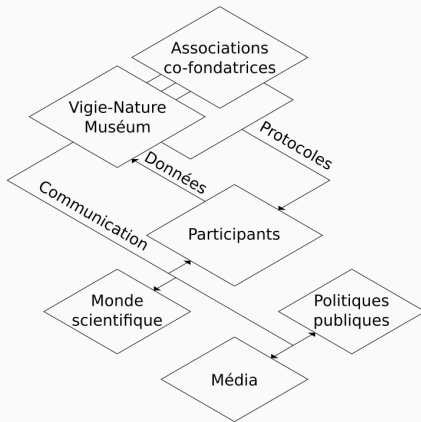
VIGIENATURE
Un réseau de citoyens
qui fait avancer la science

Vigie nature



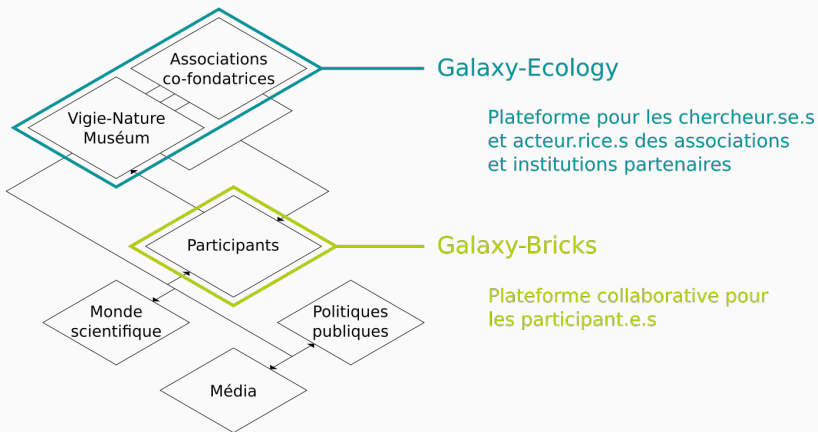
- bulle -> clarifier -> regarder slide Karine

Organisation du réseau d'acteurs



test

Destination des plateformes d'analyse de données



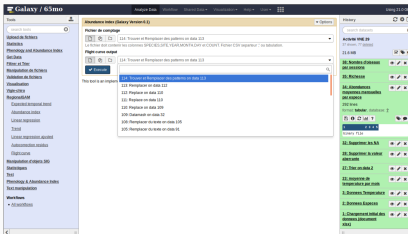
Objectifs identifiés - “grand public”

- Proposer de nouvelles possibilités pour la participation citoyenne
- Donner un accès aux données et permettre leur exploitation
- Donner aux participants les moyens de répondre aux questions qu'ils se posent sur les données

- Proposer un nouvel outil pour l'apprentissage de la démarche scientifique
- Formation à l'analyse de données
- Possibilité de proposer une approche interdisciplinaire

Réflexion sur l'ergonomie

```
site serie = as.POSIXlt(paste0(as.Date(origin), length = nday,  
dayno = as.numeric(journal(date, origin = as.Date(origin),  
month = as.numeric(strftime(date, format = "%m")),  
week = as.numeric(strftime(date, format = "%W"))),  
format = "%Y-%m-%d %H:%M:%S", date, series, format = "%Y-%m-%d  
%H:%M:%S"), as.numeric(strftime(date, format = "%Y-%m-%d")))  
site list <- c(site_count[duplicated(sp_data$SITE)], c("SITE"))  
  
all_day_site <- data.frame(SPECIES = sp_data$SPECIES[i], SITE =  
SITE = sp_data$SITE[j], MONTH = month, WEEK = week, DAY = d,  
COUNT = Nj)  
  
count_index <- paste(match(site_data$SITE, site_list)$DAYNO, sep =  
sep = "\n")  
all_day_site$count[count_index] <- sp_data$COUNT  
count_index <- aggregate(sp_data$COUNT, by = list(sp_data.  
names(site_count_length$as.character(site_count_length$SITE).  
site_count_index) ~ SITE | stack(site_count_length$SITE).  
site_count_index), FUN = sum)  
count_index <- count_index[count_index] <- site_count$values + add  
= 0  
# add zero to close observation season two weeks before and after  
first_obs <- min(all_day_site$MONTH[is.na(all_day_site$COUNT)]  
+ last_obs <- max(all_day_site$MONTH[is.na(all_day_site$COUNT)])  
closing_season <- c((first_obs - 1):(first_obs - 7)), (last_obs  
- 7):(last_obs - 1))  
if (in closing_season is before day 1 of year 365, simply set the  
r(nfa(closing_season)))  
if (closing_season[1] == c(1))  
if (max(closing_season) == c(31))
```



CodeToTool

- Formation à l'outil via des tutoriels
- Formation à l'analyse de données écologiques contextualisée
- Création de supports interactifs

Au sein de Galaxy

- Données
- Outils
- Workflow
- Résultats
- Rapports

Plateforme d'échange

- Forum
- Chat

- Analyse collaborative

Merci

