

Laboratoire Géographie-cités  
Atelier d'ElementR


## R au long cours : retours sur trois années de thèse avec R en géographie.


Maxime Guinepain  
Doctorant Géographie-cités (UMR8504) et ESO Nantes (UMR6590)  
ATER Sorbonne Université (Licence Géographie/Master TLTE)  
[maxime.guinepain@posteo.net](mailto:maxime.guinepain@posteo.net)





Géographie-cités  
UMR 8504





 Présentation

 Ressources &  
formation

 Les enquêtes  
déplacements

 Mode  
d'organisation

 Environnement  
logiciel

 Discussions

# Présentation de mon travail

- Étude de la mobilité quotidienne des personnes en emploi en France
- Une approche pensée au départ comme relevant des méthodes mixtes


## La mobilité comme domaine commun ?


Comment les mobilités quotidiennes reflètent-elles le positionnement social des travailleur·ses ?


Comment prolongent-elles les rapports sociaux (Kergoat, 2001 et 2011) à l'œuvre au travail et au sein des ménages ?


## La mobilité comme sujet de conflit ?


Dans le contexte de la valorisation sociale d'un mode de vie fondé sur de longs déplacements, comment négocier la transition écologique dans les classes populaires ?


 Présentation

 Ressources &  
formation

 Les enquêtes  
déplacements

 Mode  
d'organisation

 Environnement  
logiciel

 Discussions

# Présentation de mon travail

- Étude de la mobilité quotidienne des personnes en emploi en France
- Une approche pensée au départ comme relevant des méthodes mixtes
  - Un travail qui s'appuie sur l'exploitation d'une base de donnée « massive » (>600.000 individus), la Base Unique du Cerema, dans le cadre du projet Mobiliscope
  - Des analyses statistiques et spatiales qui sont centrales dans le processus d'administration de la preuve

- Présentation
- 🚊 Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Itinéraire de formation

- Des bases de programmation (comme « hobby ») en Basic et en C#
  - *Des langages qui, par certains aspects, ressemblent à R, et des « tics de langage »...*
- Formation à l'ENS de Lyon en 2017
- Formation à Paris-Cité dans le cadre du master Géoprisme en 2019
- Autoformation en mars 2020...



- Présentation
- 🚊 Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Ressources de formation

- Les ateliers pour débiter
- Un travail sur des cas d'étude dans le cadre de ma formation en Master
- Un manuel générique (tel qu'*R pour les géographes*)

## *Mais aussi...*

- La documentation, souvent très utile, détaillée et riche d'exemples dans R
- La « paradocumentation » = les « cheatsheet », essentielles, notamment au début
- StackOverflow, à la vie à la mort

○ Présentation

🚊 Ressources &  
formation

○ Les enquêtes  
déplacements

○ Mode  
d'organisation

○ Environnement  
logiciel

○ Discussions

# Ressources de formation

- Les ateliers pour débiter
- Un travail sur des cas d'étude dans le cadre de ma formation en Master
- Un manuel générique (tel qu'*R pour les géographes*)

## *Mais aussi...*

- La documentation, souvent très utile, détaillée et riche d'exemples dans R
- La « paradocumentation » = les « cheatsheet », essentielles, notamment au début
- StackOverflow, à la vie à la mort (= ChatGPT, en version originale)

# Ressources de formation

Présentation

Ressources & formation

Les enquêtes déplacements

Mode d'organisation

Environnement logiciel

Discussions

The screenshot shows the RStudio interface. The main editor contains R code for a spatial analysis script. The right-hand pane shows the help documentation for the `st_join` function, with a search bar and a list of related functions. A white callout box with a black border is overlaid on the help pane, containing the text: "La documentation, accessible facilement dans l'IDE". The console at the bottom shows the R prompt and some output.

```
1 # FACTOINDIV V6FF
2 # Un script développé de juillet à septembre 2023
3 # en vu de la publication d'un article
4 # Maxime Guinepain
5
6 source("START.R", print.eval = T)
7
8 # seuilSignifiant = 20
9 # pasDeTemps = 60
10 # vite = T
11 # garder = c("initMémoire", "seuilSignifiant", "pasDeTemps", "vite", "garder")
12
13 if (!dir.exists("Sorties/Figures/FactoIndiv v6FF")) { dir.create("Sorties/Figures/FactoIndiv v6FF") }
14
15
16 # Et si on refaisait une analyse factorielle à l'intérieur d'un pool simplifié de journées ?
17 # viz_enTete("Analyse factorielle FactoIndiv v6f", soustitre = "Juillet 2023")
18
19 -# Calcul de la densité des ZF =====
20 rm(list = ls()[!ls() %in% garder], pos = globalenv())
21 initMémoire(BasesCharger = c("shp_ZF", "shp_COM"))
22
23 if(!"densite" %in% colnames(shp_ZF))
24 {
25   grille = read_sf("Sources/Maillles/carreaux_200m_met.shp")
26
27   # Approche 1 : avec changement de maillage complet
28   grille$surf1 = as.double(st_area(grille))
29   # grille = st_intersection(grille, shp_ZF)
30   grille$surf2 = as.double(st_area(grille))
31   # grille$surf1 = grille$surf1 / 10^6
32
```

La documentation,  
accessible facilement dans  
l'IDE

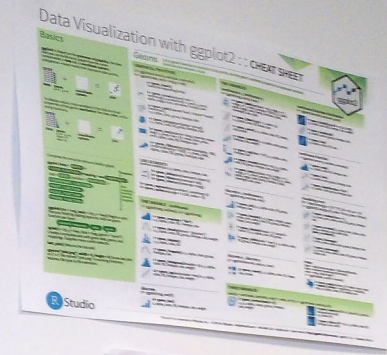
Novembre 2023

Atelier ElementR  
R au long cours

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Ressources de formation

Les cheatsheets, à toujours garder sous la main





Présentation

Ressources &  
formation

Les enquêtes  
déplacements

Mode  
d'organisation

Environnement  
logiciel

Discussions

# Ressources de formation

st\_voronoi and basic plot with R

Asked 3 months ago Modified 3 months ago Viewed 52 times Part of R Language Collective

Everything you need to capture and integrate AI knowledge into your workflows. Get started for free →

I would like to execute an `st_voronoi()` with a basic plot in R

text.txt=

name	long	lat	water_level	elevation	depth
EM_01	18.553392	-34.07027	14.4	20.358	63.0
EM_27	18.574777	-34.068709	16.196	19.966	48.0
EM_29	18.613985	-34.053271	18.766	25.477	39.0
EM_20	18.654089	-34.045177	20.102	36.502	45.0

with R code:

```
library(sf) # 'simple features' representations of spatial objects

file = 'text.txt'
#-- import
cfaq <- read.csv(file, header = 1, sep = ',', dec = '.')
#- set as a spatial feature with xy coords an existing projection
cfaq.sf <- st_as_sf(cfaq, coords=c("long", "lat"), crs = 4326) #wgs83
#- transform to local crs
cfaq.sf <- st_transform(cfaq.sf, crs = 32734) #utm 34s
# Voronoi tessellation
voronoi_grid <- st_voronoi(cfaq.sf)

# basic plot with points on top of the Voronoi
plot(voronoi_grid, col = "lightblue", border = "black", lwd = 1.5)
plot(cfaq.sf, col = "red", pch = 16, cex = 2, add = TRUE)
```

StackOverflow, la réponse  
(à peu près) à tout

○ Présentation

○ Ressources &  
formation

🚊 Les enquêtes  
déplacements

○ Mode  
d'organisation

○ Environnement  
logiciel

○ Discussions

## R dans le cadre d'une thèse

- Des données particulièrement nombreuses, avec une architecture particulière
- Des « aménagements » de ces données qui produisent de nombreux fichiers intermédiaires
- De nombreuses analyses qui partent parfois dans des directions différentes
- Un retour permanent sur des « vieux » bouts de code
- Cependant, une cohérence à maintenir

→ *À discuter* : sûrement des points communs avec d'autres types de projets...

Cependant, une autre particularité : travail individuel.

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Le Mobiliscope

- La thèse s'inscrit dans le cadre du projet Mobiliscope, dont elle exploite une partie de la base de données.
- Par cet objet commun, par ma formation et par des questionnements convergents, la thèse reprend une architecture propre au projet et s'inspire de la façon dont il est programmé.



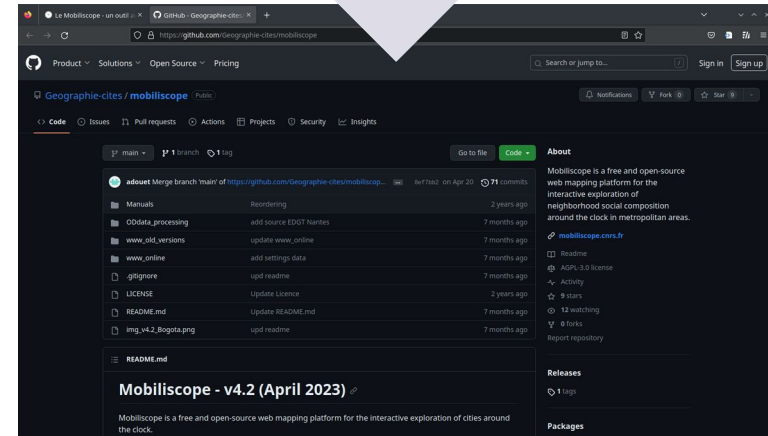
- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Le Mobiliscope

- La thèse s'inscrit dans le cadre du projet Mobiliscope, dont elle exploite une partie de la base de données.
- Par cet objet commun, par ma formation et par des questionnements convergents, la thèse reprend une architecture propre au projet et s'inspire de la façon dont il est programmé.



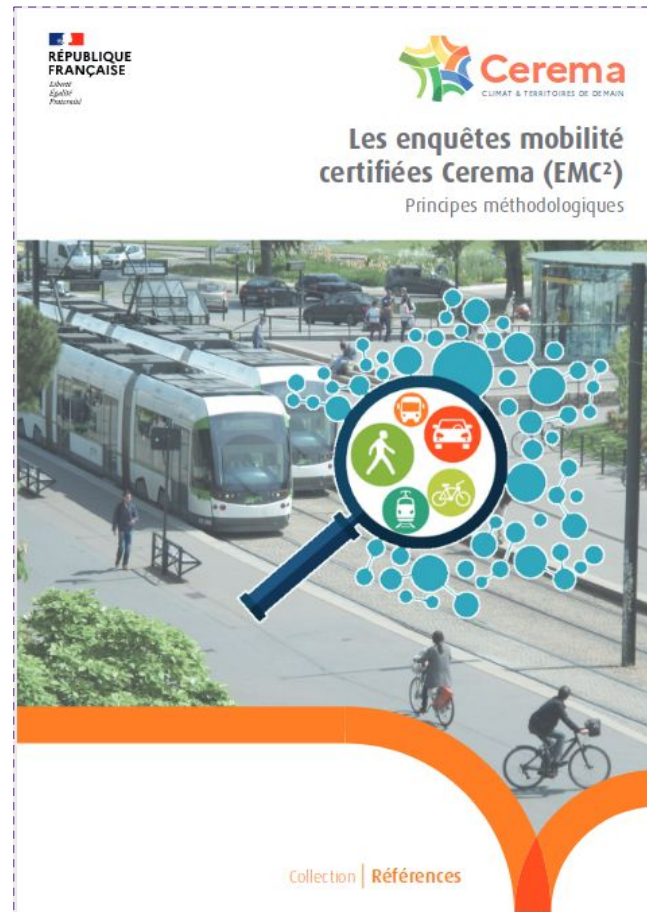
## LES POPULATIONS PRÉSENTES ET LEURS MOBILITÉS AU COURS DE LA JOURNÉE



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# La Base unique du Cerema

- Une base issue des enquêtes de déplacement certifiées Cerema
- Des données standardisées, issues de nombreuses enquêtes (= quelques soucis méthodologiques...)
- Non pas « une » mais cinq bases en réalité



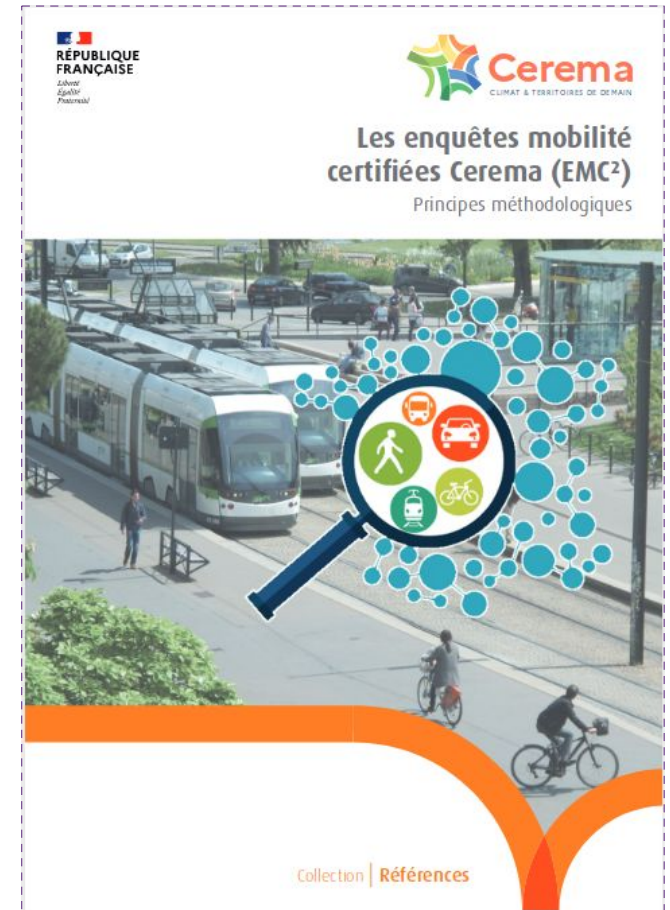
○ Présentation

○ Ressources &  
formation○ Les enquêtes  
déplacements○ Mode  
d'organisation○ Environnement  
logiciel

○ Discussions

# Les enquêtes origine-destination du Cerema

- Une méthodologie standardisée, héritée des enquêtes de mobilité des années 1960 (ancien « standard CERTU »)
- Questions générales sur les mobilités + relevé horodaté des déplacements de la veille (jour ouvrable) sur 24h, par ménage (2 ou tous les individus), localisé sur un maillage *ad hoc*



Présentation

Ressources & formation

Les enquêtes déplacements

Mode d'organisation

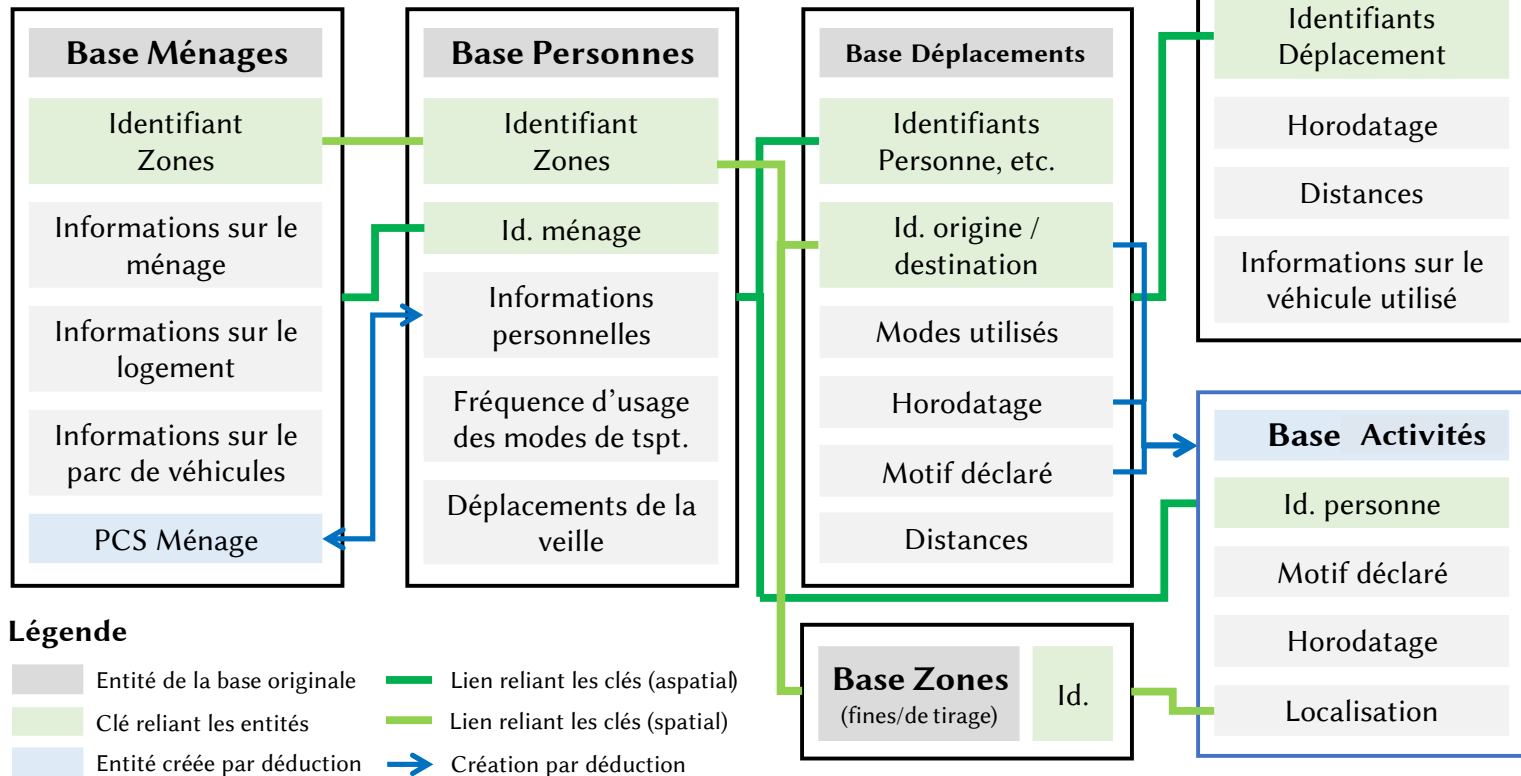
Environnement logiciel

Discussions

# La Base unique du Cerema

## Structure des bases de l'Enquête Ménages Déplacements

M. Guinepain, 2020



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# La Base unique du Cerema

- Certaines bases sont accessibles via Progedo (avec des caractéristiques spécifiques)
- Certaines ont été mises en ligne par leurs maîtres d'œuvre sur des portails d'OpenData
  - *C'est le cas de l'EDGT Loire-Atlantique 2015, exemple traité lors de cette séance.*



**D'autres enquêtes en accès libre :**

Royan (2014), Lille (2016),  
Grenoble (jusqu'à 2016), Poitiers  
(2018), Brest (2018)...

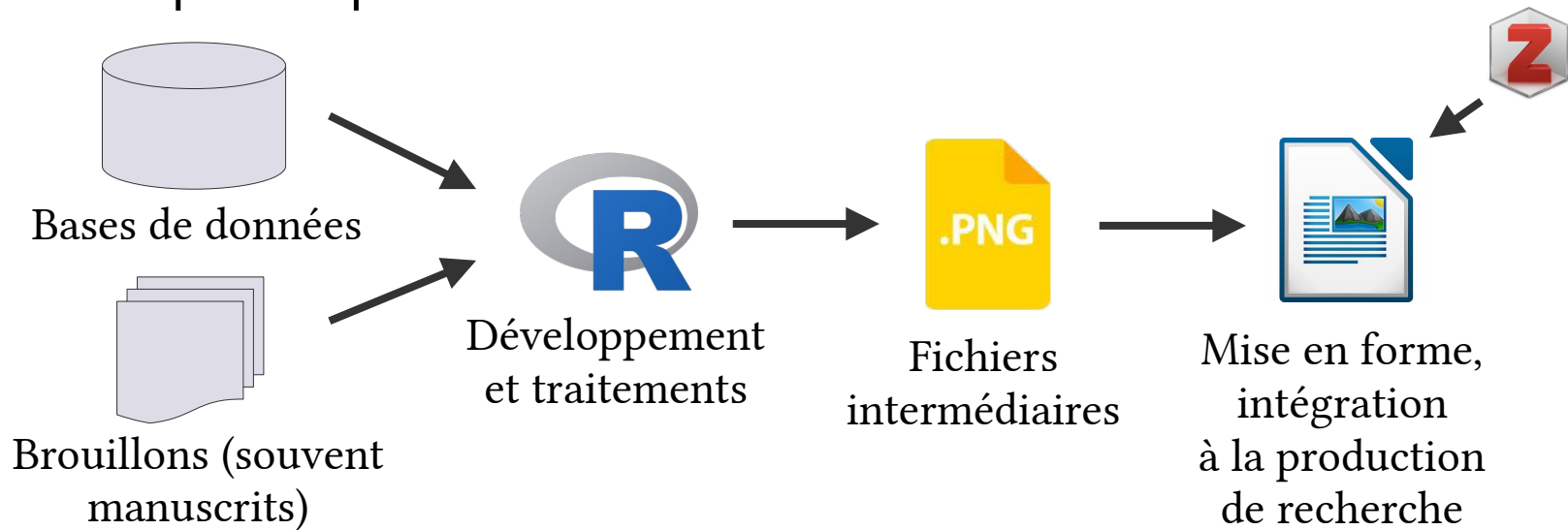
**Mais des cas d'enquêtes qui  
ont « disparu »**

Calvados, Nord Pas-de-Calais



# Un « workflow » à l'ancienne

- Le « workflow » = l'organisation du travail dans ses différentes étapes, dont R n'est qu'un élément
- Des possibilités d'intégration, voire d'automatisation, offertes par les outils Rmarkdown
- Cependant, je préfère travailler « à l'ancienne », dans des scripts uniquement consacrés au code



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚆 Mode d'organisation
- Environnement logiciel
- Discussions

# Éviter les redites

- But : ne pas réécrire de nombreuses fois les mêmes bouts de code, et pouvoir les « mettre à jour » (pour faire face à des mises à jour, pour améliorer certains rendus ou à mesure que mon « niveau » augmente...)
- Une « architecture » de scripts qui font référence les uns aux autres, inspirée de comment fonctionnent les projets de programmation logiciels
- Tout est rassemblé dans un même répertoire qui représente l'environnement du projet

```
source("START.R", print.eval = T)
```

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

# Des fonctions partout

- Les fonctions comportent une série d'instructions à appliquer (en général à leurs arguments)
- Elles peuvent effectuer des actions « en silence » ou rendre un résultat (sous la forme d'un objet)
- Elles constituent dans R un objet de l'environnement, comme les variables, les bases...
- Elles permettent d'exécuter de nombreuses fois une même séquence de codes (avec des arguments qui peuvent changer)
  - Dans certains cas, il est nécessaire d'écrire des fonctions pour pouvoir paralléliser ou appliquer à de grands ensembles de données une même séquence de code (type `lapply( )`).

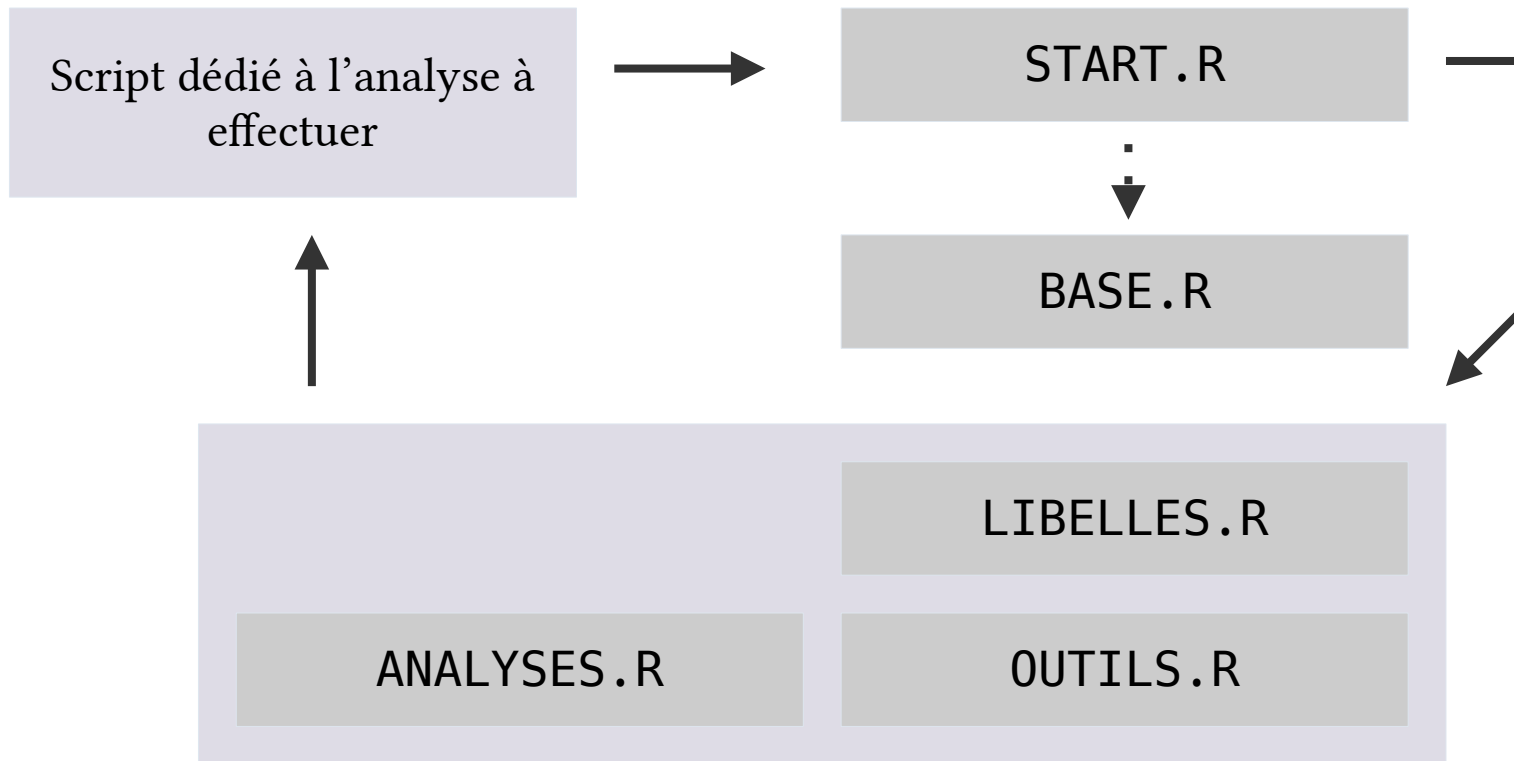
- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

# Des fonctions partout

- Les fonctions de l'environnement local sont à utiliser comme n'importe quel fonction d'un paquet : on les appelle par leur nom, on leur donne des arguments (nommés si on veut) entre parenthèses (qui sont obligatoires pour les appeler), on peut attribuer leur sortie avec l'opérateur `<-` ou `=`.
- Cependant, elles n'exécutent pas le code de la même manière qu'une ligne de code hors fonction :
  - Les sorties consoles doivent être explicitement formulées à l'aide de `cat( )` ou `print( )`, tout comme les sorties graphiques ;
  - Les fonctions opèrent dans leur propre environnement et laissent l'environnement général inchangé.

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Un ensemble de scripts interdépendants



```
source("START.R", print.eval = T)
```

○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

🚗 Mode  
d'organisation


○ Environnement  
logiciel

○ Discussions

# Des fonctions partout

- Un exemple de fonction très bête :


```
# Equivalent de dev.off(),  
# sans sortie intempestive sur le terminal  
off = function()  
{  
  o = dev.off()  
}
```



○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

 Mode  
d'organisation

○ Environnement  
logiciel

○ Discussions

# Des fonctions partout

- Un exemple de fonction très bête

```
# Equivalent de dev.off(),  
# sans sortie intempestive sur le terminal  
off = function()  
{  
  o = dev.off()  
}
```

- Quatre types de fonction :
  - Pour **standardiser** (typiquement, les sorties graphiques)
  - Pour **envelopper** (= se substitue dans mon code à des fonctions natives ou issues de paquets, dont j'adapte le fonctionnement à mes besoins)
  - Pour **des trucs pratiques** (= aller plus vite sur certaines opérations : conversions, tris...)
  - Pour exécuter des analyses complexes en parallèle ou plusieurs fois ou pour clarifier le cheminement du code.

○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

🚊 Mode  
d'organisation

○ Environnement  
logiciel

○ Discussions

## Des fonctions partout → pour standardiser

- Fonction `sortie(nom = "sortie", format = "png", taille = "def", portrait = F, h = 11, l = 17, chemin = NULL, pointsize=8)`
- Son but :
  - Gagner du temps sur les appels des fonctions graphiques dans R (type `pdf( )`, `png( )...`)
  - Utiliser toujours les mêmes paramètres (formats, définition...), y compris pour l'enregistrement des fichiers
  - Gagner en flexibilité (possibilité de changer le format en changeant simplement les arguments)



○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

🚊 Mode  
d'organisation

○ Environnement  
logiciel

○ Discussions

# Des fonctions partout → pour standardiser

- Fonction `src_fig( )`
- Son but :
  - Ne pas répéter dix mille fois les mentions des sources dans le code !
  - Pouvoir mettre à jour d'un seul coup toutes les mentions s'il y a un changement dans les données
  - Adapter le texte en fonction des données utilisées (même si pas dans l'exemple fourni)
  - Gadget : générer automatiquement la date actuelle...

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

# Des fonctions partout → pour « envelopper »

- Cas des fonctions `analyseFacto( )` et `categ_kMeans( )` fournies dans le script `Analyses`.
- But = « envelopper » des fonctions qui existent déjà (les fonctions `PCA( )` et `MCA( )` ou la fonction `kmeans( )`) en mettant les données en forme, en générant des graphiques ou des informations systématiquement utiles, en ajoutant des étapes de traitement.
  - Exemple : tri des données et réattribution des n° des clusters de la fonction `kmeans( )` pour en stabiliser les résultats, car cette fonction de catégorisation repose sur un processus aléatoire.
- Avantage : s'il faut changer de fonction d'analyse (notamment pour des problèmes de mise à jour...), c'est facile !
  - Passage facile du paquet `FactoMineR` à `ade4`.

○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

🚆 Mode  
d'organisation

○ Environnement  
logiciel

○ Discussions

## Des fonctions partout → pour des trucs pratiques

- Exemple typique : fonctions de conversion horaires fournies dans le fichier exemple (HHMM vers un nombre à décimale de minutes...)
- D'autres exemples dans le fichier : une échelle symétrique pour voir des données de sur/sous-représentation /100, une fonction pour centrer-réduire avec des pondérations, une fonction de tri de tableau que je préfère aux fonctions natives...
- Parfois, c'est plus pratique d'écrire sa propre fonction pour faire ces petites choses que d'en chercher une dans les paquets disponibles !

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

# Des fonctions partout → pour des trucs pratiques

- Exemple typique : fonctions de conversion horaires fournies dans le fichier exemple (HHMM vers un nombre à décimale de minutes...)
- D'autres exemples dans le fichier : une échelle symétrique pour voir des données de sur/sous-représentation /100, une fonction pour centrer-réduire avec des pondérations, une fonction de tri de tableau que je préfère aux fonctions natives...
- Parfois, c'est plus pratique d'écrire sa propre fonction pour faire ces petites choses que d'en chercher une dans les paquets disponibles !

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚆 Mode d'organisation
- Environnement logiciel
- Discussions

## Des fonctions partout → pour clarifier

- Choix fait dans le code BASE.R qui contient de nombreuses étapes : utiliser des fonctions pour clarifier le cheminement (et appliquer les mêmes opérations aux différentes bases qui composent la base de l'enquête).
- L'intérêt est ici limité car la Base Unique est une base plus complexe que l'exemple, avec de nombreuses corrections à apporter.
- Inconvénient : peut conduire à perdre de vue les opérations qui sont vraiment effectuées.

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

## Des fonctions partout → pour clarifier

- Un cas où je pense qu'il n'est pas bon d'utiliser des fonctions : pour se dispenser d'apprendre la syntaxe...
  - Au début de ma thèse, j'utilisais une fonction pour tracer les graphiques en barre à l'aide de ggplot pour ne pas avoir à manipuler la syntaxe très compliquée propre à ce paquet.
  - Plutôt une mauvaise idée : vite limitant, vite très encombrée car je devais la complexifier pour répondre à des besoins parfois spécifiques, et a ralenti mon apprentissage de ggplot, qui a fini par s'avérer incontournable.

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

# Le cheminement des données

- Les données sources restent stockées dans un répertoire. Aucune modification ne leur est apportée. Il est toujours possible de tout régénérer à partir de ce répertoire.
- Les données transformées sont stockées dans le format .Rds (commande `save( )`) dans un répertoire à part.
- Contrôle de l'état des données dans le script (est-il nécessaire de générer une colonne ou peut-on se reposer sur la base sauvegardée ?)

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- 🚊 Mode d'organisation
- Environnement logiciel
- Discussions

# La journalisation

- Utilisation d'une petite fonction `rapport()` qui permet d'écrire du texte à la fois sur la console et dans un petit fichier texte, où il est horodaté.
- Idée =
  - pouvoir comparer le résultat d'une nouvelle exécution du code avec l'ancienne,
  - contrôler le déroulement de l'exécution du code (notamment quand il est fait hors connexion sur HumaNum ou quand je laisse l'ordi tourner tout seul)
  - garder une trace des manipulations successives de la base.



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚊 Environnement logiciel
- Discussions

# Rstudio, c'est pratique

- Le système des onglets permet de gérer plusieurs scripts à la fois, même si ce n'est pas toujours si facile.
- Le mode projet est pratique aussi (mais j'ai des soucis avec la sauvegarde des `.Rdata`).
- Accès facilité aux sorties et aux manuels.
- J'en fais une utilisation assez minimale...

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚗 Environnement logiciel
- Discussions

# R dans HumaNum

- Très pratique : reprend fidèlement l'interface de Rstudio, si ce n'est qu'elle est dans le navigateur et en ligne.
- Très efficace : possibilité d'effectuer des calculs sur un ordinateur léger en performance (ils sont faits à distance sur le serveur), voire en se déconnectant.
- Une grande capacité de mémoire vive qui permet de manipuler sans efforts de très lourdes bases de données.
- Possibilité de travailler sur plusieurs ordinateurs, sauvegarde des scripts, travail à plusieurs...

# R dans HumaNum

Présentation

Ressources & formation

Les enquêtes déplacements

Mode d'organisation

Environnement logiciel

Discussions

Plusieurs versions de R sont disponibles et basculer de l'une à l'autre est facile

L'environnement est persistant

Petit explorateur de fichiers pour gérer le stockage

The screenshot displays the RStudio Pro interface. The main editor shows a script with the following content:

```
1 # -----  
2 #  
3 #   ENQUÊTES   MÉNAGES - DÉPLACEMENTS  
4 #  
5 #   SCRIPTS DE TRAVAIL M. GUINEPAIN  
6 #   FICHER PRINCIPAL  
7 #  
8 # -----  
9  
10 # Préalables =====  
11  
12 # Pour exécuter ce script lui-même : > source("MAIN.R", print.eval=T)  
13 rm(list = ls())  
14  
15 library(tidyverse) # inclut : ggplot2, tibble, tidyr, readr, purr, stringr, forcats, dplyr  
16 library(readr)    # lecture fichier sous forme de tibble  
17 library(questionr) # fonction freq  
18 library(sf)       # gestion des couches géographiques  
19 library(DescTools) # outils divers, y compris V de Cramer  
20 library(mapsf)    # carto du Riate  
21 library(potential) # carto de potentiel du Riate  
22  
23 # Réglage du thème de tous les ggplots  
24 theme_set(theme_bw(base_size = 9))  
25  
26 # Chargement des fonctions  
27 source("F_MAIN.R", print.eval=T)  
28 source("F_LIBELLES.R", print.eval=T)  
29 source("F_ANALYSE INDIV.R", print.eval=T)  
30 source("F_ANALYSE ESPACE.R", print.eval=T)
```

The environment pane on the right shows a list of objects:

Name	Type	Le...	Size	Value
acm_P...	func...	1	84...	function...
acp_P...	func...	1	11...	function...
A...	data...	9	45...	5932499 ...
analy...	func...	1	26...	function...
bump...	func...	1	4...	function...
bump...	func...	1	58...	function...
bump...	func...	1	10...	function...

The file explorer at the bottom right shows a directory structure:

Name	Size
.RData	4 GB
.Rhistory	19.1 KB
Analyse factorielle du recours a...	215.3 KB
Anciens scripts	
bateau loire-atlantique.pdf	749.2 KB
Data	
EMD.Rproj	205 B
Exemples vélo.pdf	644.3 KB
F_ANALYSE ESPACE.R	37.8 KB
F_ANALYSE INDIV.R	54.5 KB
F_BASES.R	236.9 KB
F_LIBELLES.R	77.4 KB
F_MAIN.R	172.5 KB
Fonctions	
fond d'ecran nerd.png	148.5 KB
LoireAtl Pot.png	211.5 KB
Logues compute entrep.pdf	4.9 MB

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚊 Environnement logiciel
- Discussions

## R dans HumaNum

- Cependant, l'installation de paquets par soi-même est compliquée (même si elle semble possible, dans le répertoire créé pour l'utilisateur·rice)
- J'ai eu dernièrement de gros soucis visiblement liés à un problème de mise à jour...

*mais cela semble lié à la configuration du serveur et l'on peut compter sur l'équipe informatique d'HumaNum (affaire à suivre!)*

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- Discussions

# Le tidyverse

- Réunit un grand nombre de paquets utiles
- Notamment dplyr, tibble, readr, ggplot2...
- Un chargement facile avec un seul appel de library
- Attention aux conflits avec d'autres paquets, souvent plus anciens



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚗 Environnement logiciel
- Discussions

# Tidyverse et ggplot, un langage dans le langage...

- Le paquet dplyr : un travail systématisé sur les bases
- Possibilité de faire des opérations complexes avec des commandes simples, et de systématiser des opérations sur des colonnes (en les appelant toutes, ou selon la nature de la variable, ou selon une partie de leur nom...)
- Utilisation du « pipe » %>%

○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

○ Mode  
d'organisation

🚗 Environnement  
logiciel

○ Discussions

# Tidyverse et ggplot, un langage dans le langage...

- Ggplot : une approche complètement différente par rapport à `plot()`
  - Configuration d'un objet graphique qui n'est tracé qu'à la toute fin
  - Ces objets graphiques peuvent être mis dans l'environnement et on peut agir sur eux à volonté
  - Ils peuvent être imbriqués et combinés à l'aide du paquet `cowplot`
  - Nécessite de disposer la base en format « long » (facile avec `dplyr`)
  - Utilise l'opérateur `+` pour combiner les instructions, ce qui est très bizarre !

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚗 Environnement logiciel
- Discussions

# Tidyverse et ggplot, un langage dans le langage...

- Ggplot : une approche complètement différente par rapport à `plot()`
  - Configuration d'un objet graphique qui n'est tracé qu'à la toute fin
  - Ces objets graphiques peuvent être mis dans l'environnement et on peut agir sur eux à volonté
  - Ils peuvent être imbriqués et combinés à l'aide du paquet `cowplot`
  - Nécessite de disposer la base en format « long » (facile avec `dplyr`)
  - Utilise l'opérateur `+` pour combiner les instructions, ce qui est très bizarre !
- Traçage automatisé de la légende qui force à employer de bonnes pratiques, mais limite les possibilités. Ceci étant, tout est paramétrable au millimètre près...



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚊 Environnement logiciel
- Discussions

# Les tibble et readr

- Les tibble : des bases de données, en mieux (qu'on peut construire depuis un fichier source avec readr).
- Le gros plus pour moi : la gestion de la nature des variables, qui est un peu moins souple mais évite bien des problèmes.
  - Avec readr, on peut forcer la reconnaissance des colonnes d'un fichier CSV dans les types que l'on souhaite → moins de soucis de chaînes qu'il faut transformer en facteurs, d'identifiants interprétés comme des chiffres...
  - Readr permet aussi d'utiliser un « dictionnaire » pour importer les bases. Très pratique !

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚆 Environnement logiciel
- Discussions

# Les tibble et readr

Exemple : extrait du dictionnaire pour lire la base Ménage de l'EDGT Loire-Atlantique.

L'utilisation des dictionnaires facilite énormément la lecture des fichiers FWF (sans séparateurs !) et le passage d'une base à l'autre.

Variable	Type	Position	Taille	Libelle
MP1	f	1	1	CF
ANNE	c	2	6	EngDate
MTIR	c	8	3	ZT
MP2	c	11	3	ZF
ECH	c	14	4	Ech
M1	f	18	1	LogType
M2	f	19	1	LogOcc
M3D	f	20	1	LogInternet
M5	i	21	1	VehN
M7A1	f	22	1	Veh1_Eng
M13A	i	23	4	Veh1_Ann
M14A	i	27	2	Veh1_Psc
M7B1	f	29	1	Veh2_Eng
M13B	i	30	4	Veh2_Ann
M14B	i	34	2	Veh2_Psc

○ Présentation

○ Ressources &  
formation

○ Les enquêtes  
déplacements

○ Mode  
d'organisation

🚊 Environnement  
logiciel

○ Discussions

## Autres paquets utilisés

- Pour gérer les données spatiales, le paquet `sf`
- Pour la cartographie, `mapsf`
- Quelques paquets supplémentaires selon les besoins (`plyr` pour le recodage des variables, `spatstat`... sans doute pas les plus efficaces)
  - Certains paquets génèrent des conflits. C'est le cas de `plyr` avec le tidyverse.
  - On peut les appeler uniquement quand on a besoin avec l'opérateur `::` → `plyr::revalue()`.

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚂 Environnement logiciel
- Discussions

# Limitations et contraintes techniques

- La **mémoire vive**, en exécution en local, est un gros obstacle
  - R ne gère pas bien la situation quand les bases à charger dans la mémoire excèdent la mémoire disponible (gros crash, voire crash du système sur mon Linux...)
  - Impossible de travailler sur les bases autrement qu'en les chargeant dans la mémoire vive
- De façon générale, R tolère très mal les fonctionnements à base de **boucles** (il faut travailler avec `apply( )` ou `lapply( )`... qui me semblent très peu intuitives).

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- 🚊 Environnement logiciel
- Discussions

# Le problème des versions...

- R est un projet vivant, le logiciel et les paquets sont régulièrement mis à niveau... et cela peut « casser » le code
- Situations rencontrées, auxquelles être vigilant·e
  - Une fonction mise à jour qui n'accepte plus les mêmes arguments : m'est arrivé avec dplyr récemment. Parfois, est précédé d'un avertissement avant la mise à jour (« attention, va prochainement expirer »)...
  - Un paquet qui n'est plus mis à jour par ses créateur·rices et qui cesse de fonctionner sur une version de R mise à jour.

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- 🚋 Discussions

# La portée pédagogique de R

- Un questionnaire sur la substitution de R en cours aux tableurs et logiciels de SIG.
- Est-ce plus pratique pour les étudiant·es ?
  - Rare encore sont celles et ceux qui ont vraiment appris à coder dans les filières de géo. Coût d'entrée qui peut sembler abrupt par rapport à l'utilisation de tableurs et de logiciels de SIG, surtout quand ce sont des étudiant·es avancé·es qui en ont fait avant.
  - Toutefois, on s'arrache tellement les cheveux avec ces logiciels que la question se pose : serait-il vraiment pire d'apprendre quelques bases de code ?
  - La question se pose d'autant plus alors que de nombreux·ses étudiant·es, utilisant des tablettes et smartphones, ne sont plus très familier·es avec les interfaces « clique-bouton »

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- 🚋 Discussions

# La portée pédagogique de R

- Des avantages sûrs
  - Possibilité de dérouler des scripts de façon logique, de corriger des erreurs a posteriori, d'annoter les scripts pour mémoriser les étapes de traitements
  - Pas besoin de se préoccuper des réglages des logiciels et de contourner l'étrange logique qui préside à la conception de leurs interfaces...
  - Une compétence valorisable par la suite
- Mais...
  - Quelle pertinence au regard des projets professionnels ?
  - Quelles injustices causées par le recours au code, qui peut impressionner ?
  - Quels problèmes d'accessibilité pose le recours à des scripts ?

- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- 🚊 Discussions

# Le code ouvert et l'informatique libre

- R (et Rstudio, bien que ce produit soit aussi vendu par une compagnie) s'inscrivent dans le logiciel libre
- Philosophie du libre dans la mise à disposition des paquets, encourage à la mise à disposition du code
- Intégration à réaliser avec les dépôts git, mise à disposition de ressources...



- Présentation
- Ressources & formation
- Les enquêtes déplacements
- Mode d'organisation
- Environnement logiciel
- 🚆 Discussions

# Merci de votre attention

Pour me contacter :  
[maxime.guinepain@posteo.net](mailto:maxime.guinepain@posteo.net)

Remerciements spéciaux :

- L'équipe ElementR : Hugues, Robin, Léa et Marion
- Julie et Aurélie du projet Mobiliscope,
- Julien, Luc, Pierre, Ronan, Romain...  
pour nos discussions « code »