

Résultats : Une méthode commune

Ces bibliothèques permettent de partager un même

 **Lire les données : sans élever la question de l'usage**

 **Partager des requêtes : pour partager les usages**

• Trouver des atypies à corriger

Validiser l'activité en ne-PMS

Alimentez des entreprises et lancez des idées (décision et recherche)

• Ancrer la donnée pour finaliser les résultats (réproductibilité)

 **Partager des applications réutilisant ces données**

- Utiliser cette approche permet de mieux s'appropriier les données en local (autonomie, compréhension, technicité, règles d'exceptions)

2024-03-26 - Saint-Jacut-de-la-Mer

Résultats : Une méthode commune

- Ces bibliothèques permettent de partager une même approche pour :
 - **Lire les données : sans en questionner l'usage**
 - **Partager des requêtes : pour partager les usages**
 - Trouver des atypies à corriger
 - Valoriser l'activité comme e-PMSI
 - **Alimenter des entrepôts et lacs de données (décisionnel et recherche)**
 - **Ancrer la donnée pour fiabiliser les résultats (reproductibilité)**
 - **Partager des applications réutilisant ces données**
- Utiliser cette approche permet de mieux s'approprier les données en local (autonomie, compréhension, technicité, règles d'exceptions)

Résultats : Une méthode commune, deux exemples en image

```
R : Lire et requêter les données

library(pmeasyr)
library(dplyr, warn.conflicts = FALSE)
library(nomensland)

p ← noyau_pmeasyr(finess = '750712184',
                 annee = 2022,
                 mois = 12,
                 path = '~/Documents/data/mco/',
                 tolower_names = TRUE,
                 lib = FALSE)

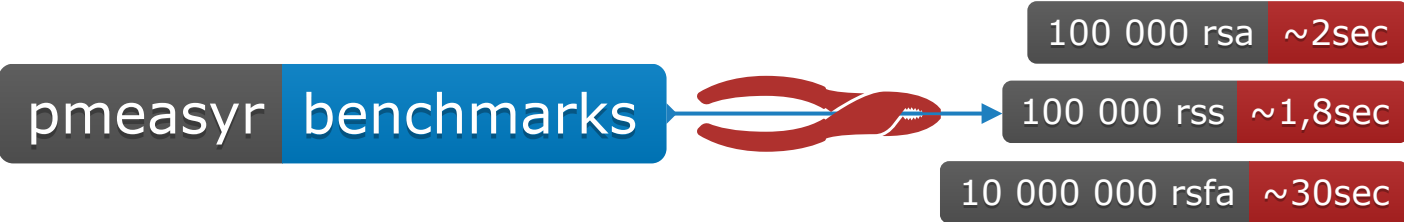
rsa ← irsa(p) %>% prepare_rsa

# chirurgie de l'obésité
chir_obesite ← get_all_listes("Chirurgie bariatrique")
sejours_chir_obesite ← lancer_requete(rsa, chir_obesite)

# Recours exceptionnel
recours_exc ← get_all_listes("Recours Exceptionnel")
sejours_recours_exc ← lancer_requete(rsa, recours_exc)

# requête ad hoc
exemple_requete ← list(
  actes = c('QEFA003', 'QEFA005', 'QEFA010',
            'QEFA013', 'QEFA015', 'QEFA019',
            'QEFA020')
)

sejours_qefa ← requete(rsa, exemple_requete)
```



```
python : Lire les données et les convertir en fichier parquet

import polars as pl
import pypmsi as pm

p = pm.noyau_pmsi(
    finess = '290000017',
    annee = 2021,
    mois = 12,
    path = '~/Documents/data/mco')

# Lire les données 2022
rsa = p.irsa(annee = 2022, typi = 4)
# rsa['rsa'] : Shape(141_901, 89)

#
# | nofiness | novrsa | cle_rsa | novrss | ... | ndas | na | ghm |
# |-----|-----|-----|-----|---|-----|---|-----|
# | str      | str    | str     | str    |   | i32  | i32 | str  |
# |-----|-----|-----|-----|---|-----|---|-----|
# | 290000017 | 227    | ABCDEFGHI | 121    | ... | 10   | 2   | 23Z02Z |
# | 290000017 | 227    | ABCDEFGHI | 121    | ... | 113  | 90  | 10C024 |
# | ...      | ...    | ...     | ...    | ... | ...  | ... | ...   |
# | 290000017 | 227    | ABCDEFGHI | 121    | ... | 0    | 2   | 23M20T |
# | 290000017 | 227    | ABCDEFGHI | 121    | ... | 2    | 0   | 28Z07Z |
# | 290000017 | 227    | ABCDEFGHI | 121    | ... | 27   | 25  | 03M074 |
#
# écrire les données en fichier parquet
for key, tab in rsa.items():
    tab.write_parquet("~/Documents/data/parquet/rsa/" +
                     key + ".parquet")
```

