



Objectif :

Acquérir une pratique élémentaire d'outils permettant de traiter des données en Python, en vue de faciliter la pratique des TP en cours de data-sciences, pour le développement des projets industriels qui en ont besoin, pour le stage, et au delà...

Moyens :

- ➡ Prendre en main des outils (Git, JupyterLab, RosettaHUB/AWS,...) et concepts indispensables tout au long des enseignements du MS autour de la data ;
- ➡ Travailler sur trois jeux de données : des nombres, du texte, des images ;



Sommaire :

1. Gestion de code source, **GIT**
2. Cycle de développement de code pour les data-sciences,
JupyterLab
3. Le minimum vitale sur les "APIs"
4. Accès au Cloud via une API, **boto3** & co.
5. Jeux de données ...
6. Description du mini projet (validation IPv4 & FRG)



Pourquoi a t-on besoin d'un gestionnaire de versions ?

```
$ echo "https://gitlab-student.centralesupelec.fr/" > gitlab.txt  
$ cp gitlab.txt gitlab2.txt  
$ echo "https://gitlab-research.centralesupelec.fr/" >> gitlab2.txt
```

```
$ cat gitlab.txt  
https://gitlab-student.centralesupelec.fr/  
$ cat gitlab2.txt  
https://gitlab-student.centralesupelec.fr/  
https://gitlab-research.centralesupelec.fr/  
$
```

Différences entre
l'ancien et le nouveau

```
$ diff gitlab.txt gitlab2.txt  
1a2  
> https://gitlab-research.centralesupelec.fr/  
$ diff gitlab2.txt gitlab.txt  
2d1  
< https://gitlab-research.centralesupelec.fr/  
$
```

Différences entre le
nouveau et l'ancien



Upgrade de l'ancienne vers la nouvelle ?

Fabrication d'un
"patch"

```
$ diff gitlab.txt gitlab2.txt > patch_old2new.txt
$ cat patch_old2new.txt
1a2
> https://gitlab-research.centralesupelec.fr/
$ █
```

Application de ce patch
et vérifications

```
$ patch gitlab.txt patch_old2new.txt
patching file gitlab.txt
$ cat gitlab.txt
https://gitlab-student.centralesupelec.fr/
https://gitlab-research.centralesupelec.fr/
$ diff gitlab.txt gitlab2.txt
$ █
```

Version
initiale !!

Revenir à la version précédente est également possible :

```
[$ diff gitlab2.txt gitlab.txt > patch_new2old.txt
[$ patch gitlab2.txt patch_new2old.txt
patching file gitlab2.txt
[$ cat gitlab2.txt
https://gitlab-student.centralesupelec.fr/
$ █
```



Ces opérations basiques sont à considérer pour l'ensemble d'un projet :

- Des dizaines, centaines, milliers de fichiers ;
- De multiples évolutions du contenu de chaque fichiers ;
- De nombreux développeurs (et autres) qui interviennent sur le code ;
- Plusieurs (beaucoup...) étapes majeures dans le produit :
 - ★ Des changements d'architectures, de technologies utilisées...
 - ★ Des marqueurs d'étapes (versions...)
 - ★ nécessité maintenir simultanément le code des version les plus récentes...

➔ Nécessité d'un outil (manuellement c'est vite ingérable) !



Mastère SIO - Octobre 2019 - Projet Fil Rouge GIT

Famille des gestionnaires (OpenSource) / modèles de fonctionnement

Modèles centralisés (serveur obligatoire) : RCS, CVS, SVN...

Modèles distribués (serveur optionnel) : GIT, BAZAAR, MERCURIAL,...

Les gestionnaires de versions quand ils sont adoptés ont généralement une très longue durée de vie (bien plus que les logiciels qu'ils permettent de gérer). Plusieurs décennies n'est pas rare (CVS, SVN - Subversion...)

« **Git** (2005) is not an acronym. Linus Torvalds developed Linux and he called it so. Then he developed version control system for it and named it "**Git**", which is an offensive English slang meaning "a stupid person". When asked about it, he said, "I'm an egotistical bastard, and I name all my projects after myself." »

Cf : <https://en.wikipedia.org/wiki/Git>

Site de référence pour GIT : **<https://git-scm.com/>** et sa référence texte et vidéo :

<https://git-scm.com/doc/>, **<https://git-scm.com/book/en/v2/>** et **<https://git-scm.com/videos/>**



GIT peut fonctionner en local et/ou utiliser un serveur centralisé

En pratique pour nous au SIO :

github.com : Site collaboratif, sorte de réseau social pour développeurs - Tout le monde voit tout le monde sur les projets open-sources (« read-only » sauf pour le(s) propriétaire(s)) - Version payante pour rendre un projet privé ;

gitlab-student.centralesupelec.fr (avec votre identifiant @student...)

Site collaboratif mais seulement à l'échelle du Campus de CS (c'est encouragé pour vos projets...).



En tant que gestionnaire de versions, GIT répond à quatre questions concernant les fichiers gérés :

1. Qui ?

➡ La(les) personnes qui effectue(nt) la(les) modification(s) ;

2. Quand ?

➡ Tous les événements sont datés et identifiables ;

3. Quoi ?

➡ Ce qui change est tracé et identifiable ;

4. Pourquoi ?

➡ Un commentaire (optionnel) délivre une explication



Git fonctionne en mode distribué, c'est à dire :

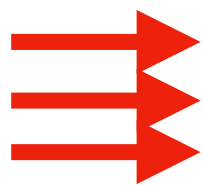
- Fonctionne en mode local pour un seul développeur - « local repository » vs dépôt local ;
- Fonctionne en mode distant pour autant de développeurs que souhaité (1 mini) - « remote repository » vs dépôt distant

Les deux modes sont souvent mixés, à savoir un dépôt local est téléchargé depuis le dépôt distant par tous les intervenants et des opérations de mise à jour, vers et depuis le dépôt distant, sont régulièrement (ou pas) effectuées.



Deux approches pour démarrer :

1. « From scratch » c'est à dire à partir de rien (mkdir, git config, git init, etc..)
2. On part d'un projet existant pour le consulter vs le modifier. Là encore deux possibilités :
 1. Dépôt anonyme (c'est le cas le plus souvent sur **github**) en lecture seule
 2. Dépôt authentifié (c'est le cas sur gitlab-student.centralesupelec.fr)



```
$ git config --global user.name "Laissus Francois"
$ git config --global user.email "francois.laissus@centralesupelec.fr"
$ git config --global core.pager "more -R"
$ cat ~/.gitconfig
[user]
    name = Laissus Francois
    email = francois.laissus@centralesupelec.fr
[core]
    pager = more -R
$
```

Interprétation correcte des
séquences d'échappement ANSI (couleurs
des caractères...)









Mastère SIO - Octobre 2019 - Projet Fil Rouge GIT

Accès
via l'interface web authentifiée -
Un seul fichier disponible...

Accès en
ligne de commandes, le même
fichier est visible

Laissus Francois > Demo-Git > Details


Demo-Git  Project ID: 1256






 Add license  1 Commit  1 Branch  0 Tags  133 KB Files


Démo pour le MS SIO


master demo-git / +

History Find file Web IDE

 Update README.md
Laissus Francois authored 11 minutes ago aa9790fc

 README  Auto DevOps enabled  Add CHANGELOG  Add CONTRIBUTING  Add Kubernetes cluster

Name	Last commit	Last update
 README.md	Update README.md	11 minutes ago

 README.md

Juste pour les besoins du cours "fil rouge" au Mastère SIO

```
[ $ git clone https://gitlab-student.centralesupelec.fr/2008laissusf/demo-git.git
Cloning into 'demo-git'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 6 (delta 1), reused 0 (delta 0)
Unpacking objects: 100% (6/6), done.
[ $ cd demo-git/
/Volumes/Home/fla/Desktop/Head/cours/Outils-DataSc/demo-git
[ $ ls -l
total 4
-rw-r--r-- 1 fla staff 59 16 oct 11:49 README.md
[ $ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean
[ $ git log --oneline
aa9790f Update README.md
2d2ded3 Add README.md
```

L'état
du dépôt local et l'historique des
modifications



Mastère SIO - Octobre 2019 - Projet Fil Rouge

GIT

édition d'un nouveau
fichier

il n'est pas (encore)
indexé (staged)

'add' le prend en
compte

Il n'est pas
enregistré dans le repository

commit
l'ajoute à l'historique et dans le
repository

```
[$ vi SIO_P2020_Etudiants.csv
[$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        SIO_P2020_Etudiants.csv

nothing added to commit but untracked files present (use "git add" to track)
[$ git add SIO_P2020_Etudiants.csv
[$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   SIO_P2020_Etudiants.csv

[$ git log --oneline
aa9790f Update README.md
2d2ded3 Add README.md
[$ git commit -m "Passage sous GIT" SIO_P2020_Etudiants.csv
[master 7927006] Passage sous GIT
 1 file changed, 16 insertions(+)
 create mode 100644 SIO_P2020_Etudiants.csv
[$ git log --oneline
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
[$
```




Mastère SIO - Octobre 2019 - Projet Fil Rouge

GIT



CentraleSupélec

```
[ $ vi SIO_P2020_Etudiants.csv
[ $ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   SIO_P2020_Etudiants.csv

no changes added to commit (use "git add" and/or "git commit -a")
[ $ git diff
diff --git a/SIO_P2020_Etudiants.csv b/SIO_P2020_Etudiants.csv
index d200988..1c5cc13 100644
--- a/SIO_P2020_Etudiants.csv
+++ b/SIO_P2020_Etudiants.csv
@@ -1,6 +1,5 @@
 SIO - P2020 - Etudiants

-Nom;Prénom;e-mail local
DESMARCHELIER;Pierre
DJOUKENG KENFACK;Roméo
FABRE;Jérémie
[ $ git log --oneline
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
[ $ git diff 7927006
diff --git a/SIO_P2020_Etudiants.csv b/SIO_P2020_Etudiants.csv
index d200988..1c5cc13 100644
--- a/SIO_P2020_Etudiants.csv
+++ b/SIO_P2020_Etudiants.csv
@@ -1,6 +1,5 @@
 SIO - P2020 - Etudiants

-Nom;Prénom;e-mail local
DESMARCHELIER;Pierre
DJOUKENG KENFACK;Roméo
FABRE;Jérémie
[ $ git checkout -- SIO_P2020_Etudiants.csv
[ $ git diff 7927006
[ $
```

Modification du fichier

'status' confirme

Différences avec
la dernière version indexée

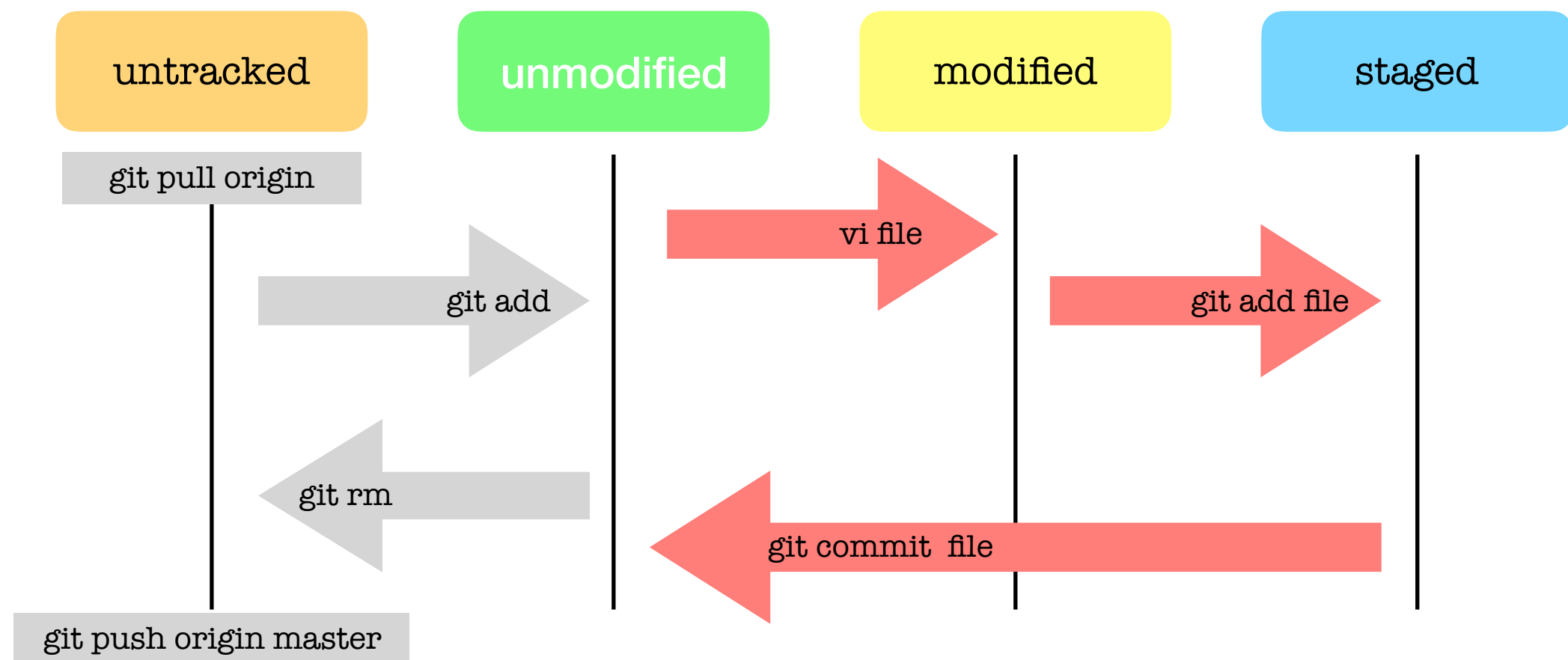
Une ligne en
moins

Même
résultat

On
récupère la dernière version
indexée pour annuler la
modification



En résumé le cycle de vie d'un fichier (*) (`git status/add/commit/...`)

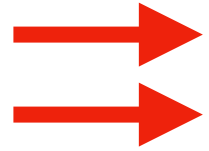


(*) Figure utilisée un peu partout sans auteur initial bien défini...



Mastère SIO - Octobre 2019 - Projet Fil Rouge GIT


Et le "push" final pour mettre à jour de dépôt distant :



```
$ push -u origin master
-bash: push: command not found
$ git push -u origin master
Counting objects: 3, done.
Delta compression using up to 12 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 545 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://gitlab-student.centralesupelec.fr/2008laissusf/demo-git.git
aa9790f..7927006 master -> master
Branch master set up to track remote branch master from origin.
$
```

Immédiatement visible via
l'interface web

Laissus Francois > Demo-Git > Details


 **Demo-Git**
Project ID: 1256

🔗 Add license → 3 Commits 1 Branch 0 Tags 195 KB Files

Démo pour le MS SIO

master demo-git / +


History Find file Web IDE

 **Passage sous GIT**
Laissus Francois authored 24 minutes ago 7927006

README Add CHANGELOG Add CONTRIBUTING Enable Auto DevOps Add Kubernetes cluster

Set up CI/CD

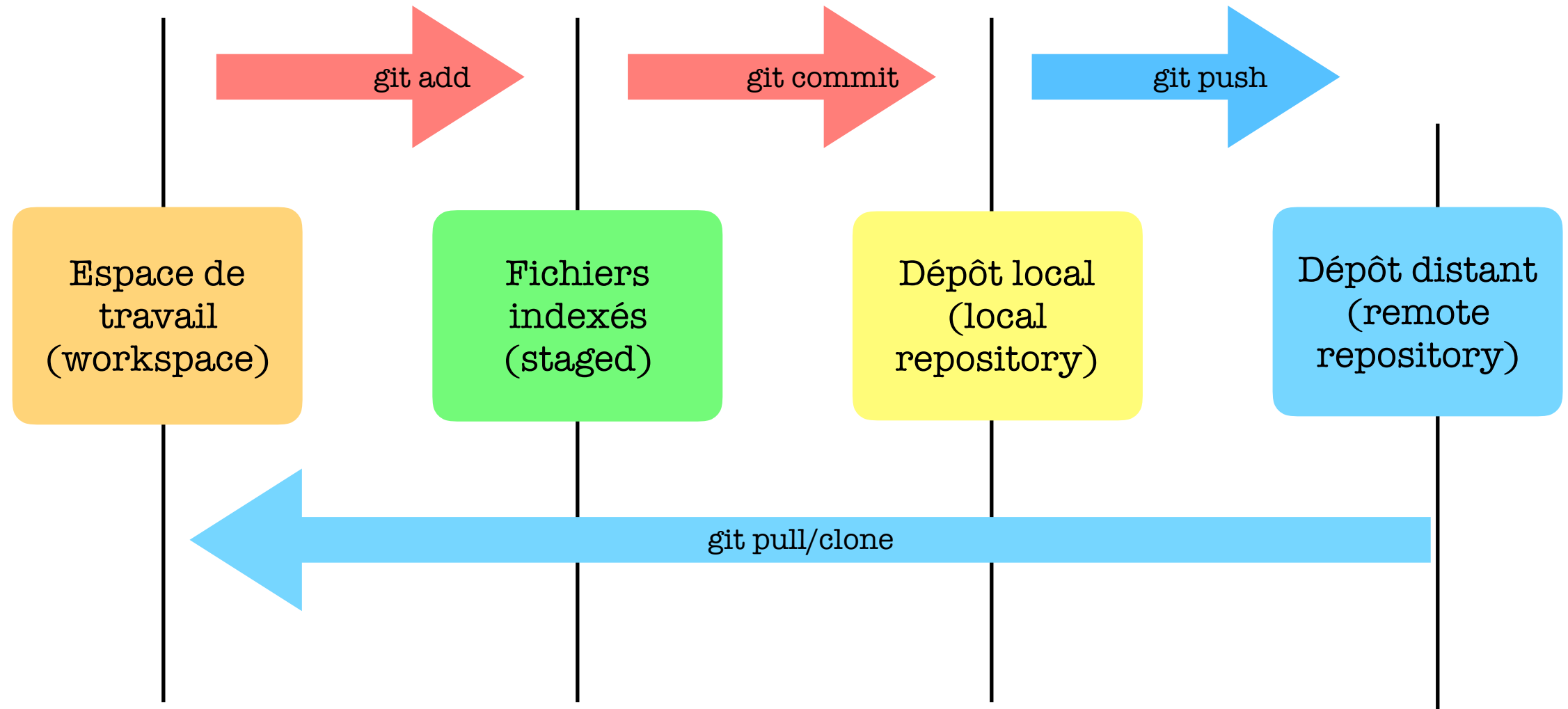
Name	Last commit	Last update
README.md	Update README.md	2 hours ago
SIO_P2020_Etudiants.csv	Passage sous GIT	24 minutes ago

 **README.md**

Juste pour les besoins du cours "fil rouge" au Mastère SIO



En résumé(*), avec un dépôt distant (comme sur la diapo précédente) :




(*) Figure utilisée un peu partout, sans auteur initial bien défini...



Les branches

Dans un dépôt GIT on travaille toujours sur une branche, par défaut elle se nomme « **master** » (par défaut)



```
$ git branch
* master
```

Le cycle de développement d'un code est rarement linéaire :

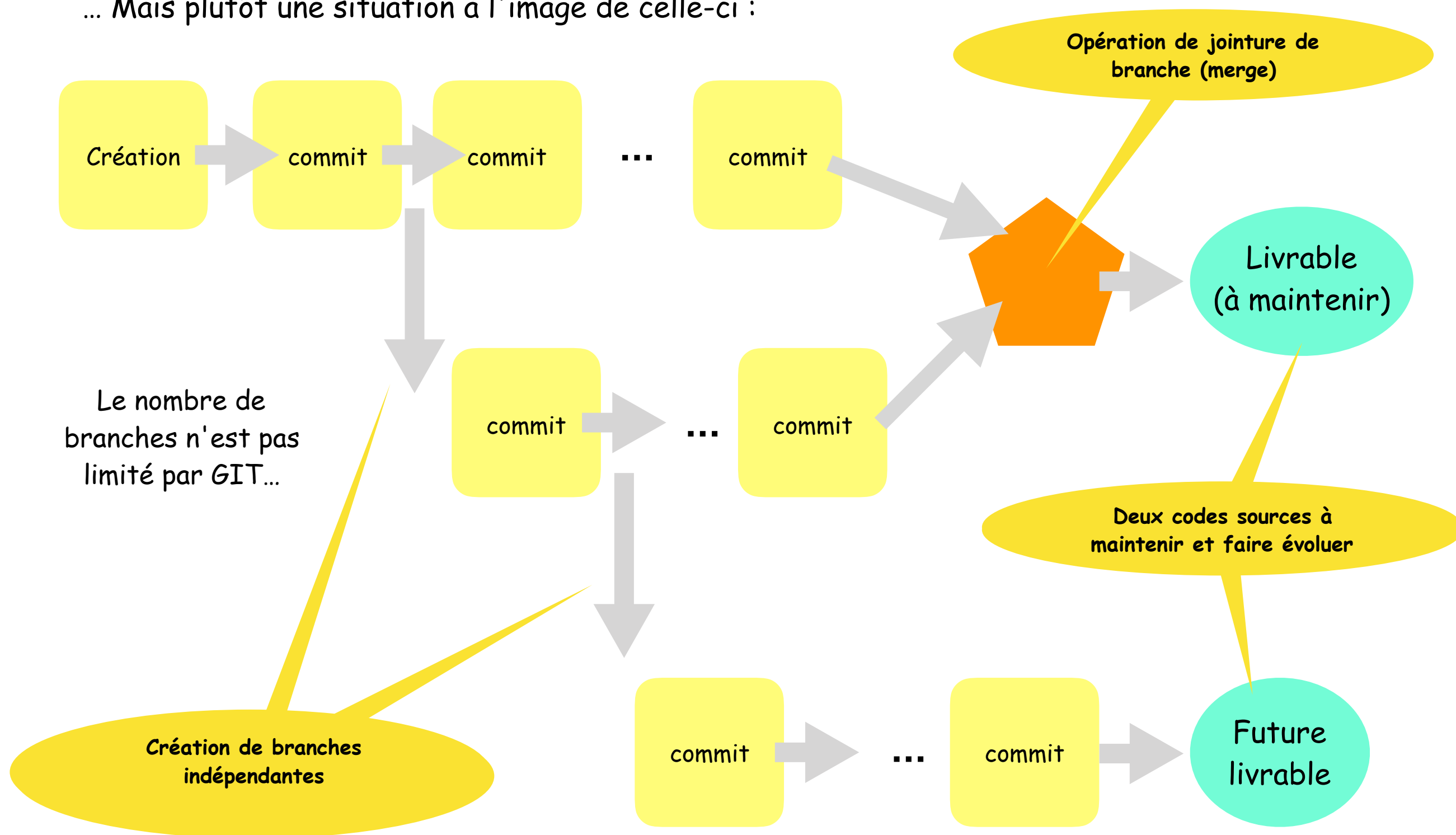
- Besoin d'expérimenter un point, de corriger un bug, (...) sans remettre le fonctionnement /architecture de l'ensemble en question ;
- Besoin de maintenir l'ensemble du projet selon plusieurs versions, par exemple une version spécifique par client, une version de test, une version en développement, etc...
-

On a donc rarement cette démarche linéaire, surtout en mode « agile » (...) :





... Mais plutôt une situation à l'image de celle-ci :





Mastère SIO - Octobre 2019 - Projet Fil Rouge

GIT



CentraleSupélec

```
[ $ git branch
* master
[ $ git branch private
[ $ git branch
* master
  private
[ $ git checkout private
Switched to branch 'private'
[ $ git branch
  master
* private
[ $ ls -l
total 8
-rw-r--r-- 1 fla staff  59 16 oct 11:49 README.md
-rw-r--r-- 1 fla staff 287 16 oct 13:26 SIO_P2020_Etudiants.csv
[ $ vi SIO_P2020_Etudiants.csv
[ $ git status
On branch private
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   SIO_P2020_Etudiants.csv

no changes added to commit (use "git add" and/or "git commit -a")
[ $ git diff
diff --git a/SIO_P2020_Etudiants.csv b/SIO_P2020_Etudiants.csv
index d200988..78a5b75 100644
--- a/SIO_P2020_Etudiants.csv
+++ b/SIO_P2020_Etudiants.csv
@@ -1,16 +1,16 @@
 SIO - P2020 - Etudiants

  Nom;Prénom;e-mail local
-DESMARCHELIER;Pierre
-DJOUKENG KENFACK;Roméo
-FABRE;Jérémie
-JOUBIOUX;Alan
-LAMOTHE;François
-LY;Stéphane
-NGBANGO;Soulémanou
-RHERMINI;Acheq
-SCAÉROU;Nicolas
-TCHAPTCHET NOUDJET;Christian
-PIRBAY;Djan - Laetiscia
-TIA;David
-VINCENT;Etienne
+DESMARCHELIER;Pierre;pierre.desmarchelier@student-cs.fr
+DJOUKENG KENFACK;Roméo;romeo.djoukeng@student-cs.fr
+FABRE;Jérémie;jeremie.fabre@student-cs.fr
+JOUBIOUX;Alan;alan.joubioux@student-cs.fr
+LAMOTHE;François;francois.lamothe@student-cs.fr
+LY;Stéphane;stephane.ly@student-cs.fr
+NGBANGO;Soulémanou;soulemanou.ngbango@student-cs.fr
+RHERMINI;Acheq;acheq.rhermini@student-cs.fr
+SCAÉROU;Nicolas;nicolas.scaerou@student-cs.fr
+TCHAPTCHET NOUDJET;Christian;christian.noudjet@student-cs.fr
+PIRBAY;Djan - Laetiscia;djan.pirbay@student-cs.fr
+TIA;David;david.tia@student-cs.fr
+VINCENT;Etienne;etienne.vincent@student-cs.fr
```

Lister les branches
existantes

Créer la branche
"private"

Rendre la
branche "private" active (HEAD)

Visualisation
des modifications dans la branche

Ajout des
adresses e-mails dans le domaine
@student-cs.fr



Mastère SIO - Octobre 2019 - Projet Fil Rouge

GIT

```
→ [$ git commit -a -m "Ajout des adresses e-mail @student-cs.fr"
[private ab16f6c] Ajout des adresses e-mail @student-cs.fr
1 file changed, 16 insertions(+), 16 deletions(-)
rewrite SIO_P2020_Etudiants.csv (82%)
→ [$ git log --oneline
ab16f6c Ajout des adresses e-mail @student-cs.fr
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
```

**Ce commit
n'est visible que dans la branche "private"**

```
→ $ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.
→ $ git log --oneline
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
```

**Modification
du fichier dans la branche "master"**

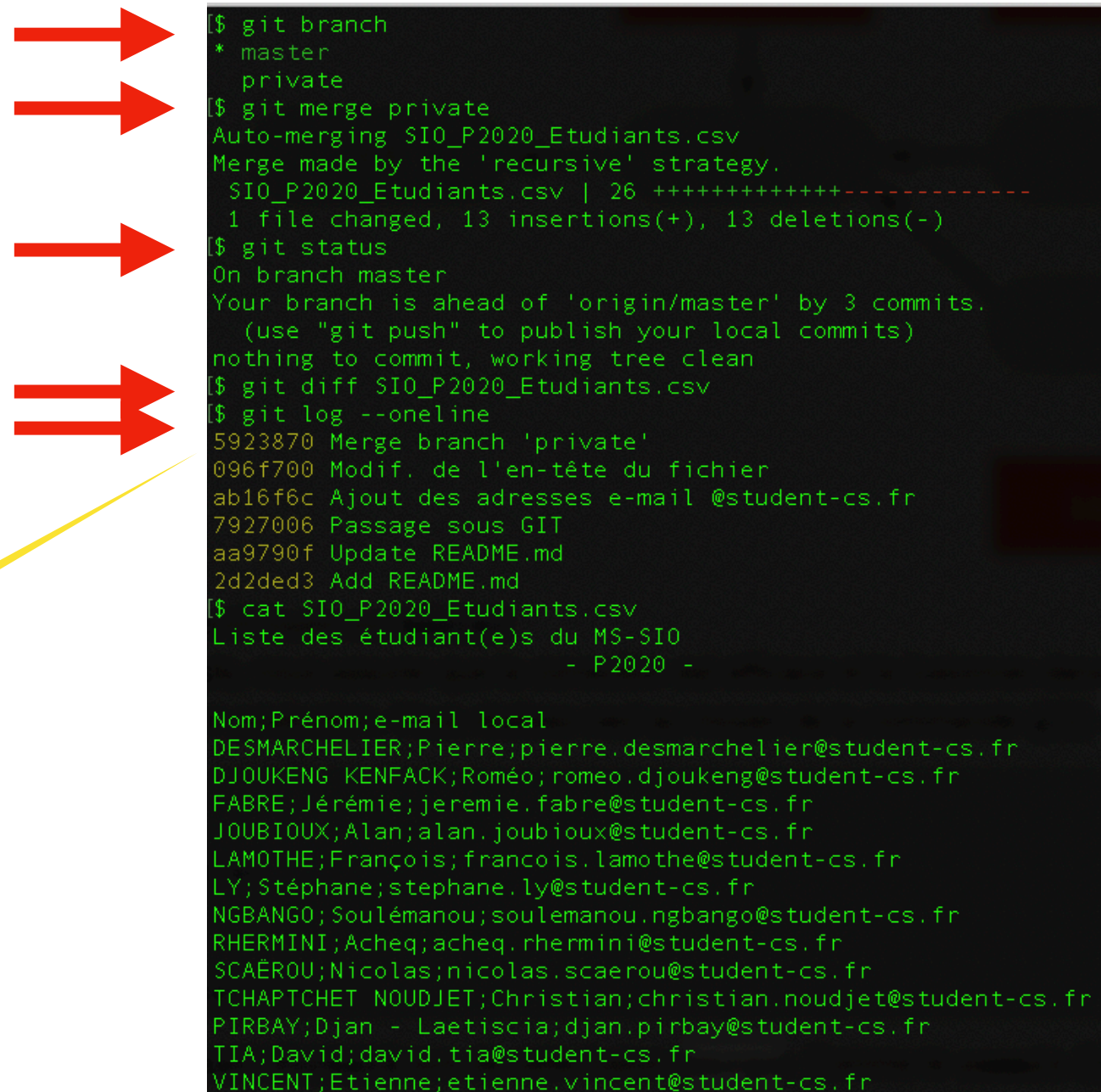
```
→ $ vi SIO_P2020_Etudiants.csv
$ git commit -a -m "Modif. de l'en-tête du fichier"
[master 096f700] Modif. de l'en-tête du fichier
1 file changed, 2 insertions(+), 1 deletion(-)
```

➡ Le fichier a subi des modifications dans les deux branches ...



Besoin de pouvoir opérer l'opération symétrique à la création de branche : la fusion de branches. Il s'agit de fusionner (merge) tous les fichiers avec leurs différences (*).

Dans le cas présent l'opération est simple :



```
[ $ git branch
* master
  private
[ $ git merge private
Auto-merging SIO_P2020_Etudiants.csv
Merge made by the 'recursive' strategy.
  SIO_P2020_Etudiants.csv | 26 ++++++-----
  1 file changed, 13 insertions(+), 13 deletions(-)
[ $ git status
On branch master
Your branch is ahead of 'origin/master' by 3 commits.
  (use "git push" to publish your local commits)
nothing to commit, working tree clean
[ $ git diff SIO_P2020_Etudiants.csv
[ $ git log --oneline
5923870 Merge branch 'private'
096f700 Modif. de l'en-tête du fichier
ab16f6c Ajout des adresses e-mail @student-cs.fr
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
[ $ cat SIO_P2020_Etudiants.csv
Liste des étudiant(e)s du MS-SIO
      - P2020 -

Nom;Prénom;e-mail local
DESMARCHELIER;Pierre;pierre.desmarchelier@student-cs.fr
DJOUKENG KENFACK;Roméo;romeo.djoukeng@student-cs.fr
FABRE;Jérémie;jeremie.fabre@student-cs.fr
JOUBIOUX;Alan;alan.joubioux@student-cs.fr
LAMOTHE;François;francois.lamothe@student-cs.fr
LY;Stéphane;stephane.ly@student-cs.fr
NGBANGO;Soulemanou;soulemanou.ngbango@student-cs.fr
RHERMINI;Acheq;acheq.rhermini@student-cs.fr
SCAËROU;Nicolas;nicolas.scaerou@student-cs.fr
TCHAPTCHET NOUDJET;Christian;christian.noudjet@student-cs.fr
PIRBAY;Djan - Laetiscia;djan.pirbay@student-cs.fr
TIA;David;david.tia@student-cs.fr
VINCENT;Etienne;etienne.vincent@student-cs.fr
```

La fusion
s'est accompagnée d'un commit du
résultat (commentaire par défaut)

(*) Cf le fonctionnement de la commande **"patch"** présentée dans les premiers slides.



Mastère SIO - Octobre 2019 - Projet Fil Rouge GIT

Mettre à
jour le dépôt distant

```
[ $ git push
Counting objects: 9, done.
Delta compression using up to 12 threads.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (9/9), 1.33 KiB | 0 bytes/s, done.
Total 9 (delta 3), reused 0 (delta 0)
To https://gitlab-student.centralesupelec.fr/2008laissusf/demo-git.git
7927006..5923870 master -> master
```

Suppression
de la branche devenue (?)
inutile

```
[ $ git log --oneline
5923870 Merge branch 'private'
096f700 Modif. de l'en-tête du fichier
ab16f6c Ajout des adresses e-mail @student-cs.fr
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
[ $ git branch -d private
Deleted branch private (was ab16f6c).
[ $ git log --oneline
5923870 Merge branch 'private'
096f700 Modif. de l'en-tête du fichier
ab16f6c Ajout des adresses e-mail @student-cs.fr
7927006 Passage sous GIT
aa9790f Update README.md
2d2ded3 Add README.md
[ $ git branch
* master
[ $ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean
```



1. Se créer un compte sur gitlab-student.centralesupelec.fr avec l'e-mail @student-cs.fr ;
2. Dans une fenêtre Terminal, configurer son outil git pour qu'il accède au dépôt /2008laissusf/demo-git.git ;
3. Vous avez dû recevoir un mail qui vous accorde le statut de développeur pour ce projet ;
4. Le but de TD est de modifier votre adresse e-mail (remplacer l'adresse par défaut par une autre, qui peut être fausse, peu importe) dans le fichier SIO_P2020_Etudiants.csv et de rendre la modification accessible à tous !
5. Enjoy GIT ;-)



Pourquoi JupyterNotebook/JupyterLab ?

- IDE le plus utilisé actuellement dans la communauté 'data*' (plus de 6 millions de downloads dans le monde) et est devenu "incontournable" ;
- Existe depuis le début des années 2010, il est désormais stable ;
- JupyterNotebook (extension .jpynb) puis son évolution actuelle JupyterLab (cf <https://jupyter.org> et <https://jupyterlab.readthedocs.io/>) ;
- Mode client-serveur en local (ou distant) ;
- L'IHM est écrit en javascript (Framework phosphor.js) et exécutable dans tout browser web moderne ;
- Permet nativement de faire du **Julia**, du **Python** et du **R** ;
- Déjà installé sur vos postes...



- C'est un environnement qui répond aux exigences des flux de travail dans le domaine des data-sciences et de la simulation ;
- Une seule et unique instance permet à l'utilisateur d'écrire du code, de l'accompagner d'explications et de l'exécuter, mais aussi de visualiser des données, d'exécuter des calculs et de vérifier les résultats obtenus ;
- Le code s'écrit par cellules, plus ou moins complexes selon les besoins ;
- Le code peut s'exécuter pas à pas, dans chaque cellule, et permet un prototypage des projets particulièrement puissant, puisque le développeur peut interagir avec son code à chaque ligne/cellule écrite sans remettre en cause tout ce qui a été exécuté (un peu comme sous un débogueur) ;



Au SIO il s'exécute en mode locale à partir d'un "launcher" qui se nomme « Anaconda Navigator »



Celui-ci est une sur-couche d'un gestionnaires de paquets pour Python et R nommé « conda ».



On peut ignorer conda dans un premier temps, le navigateur permet d'effectuer toutes les tâches essentielles (gestion des paquets logiciels, des environnements ...).



Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab

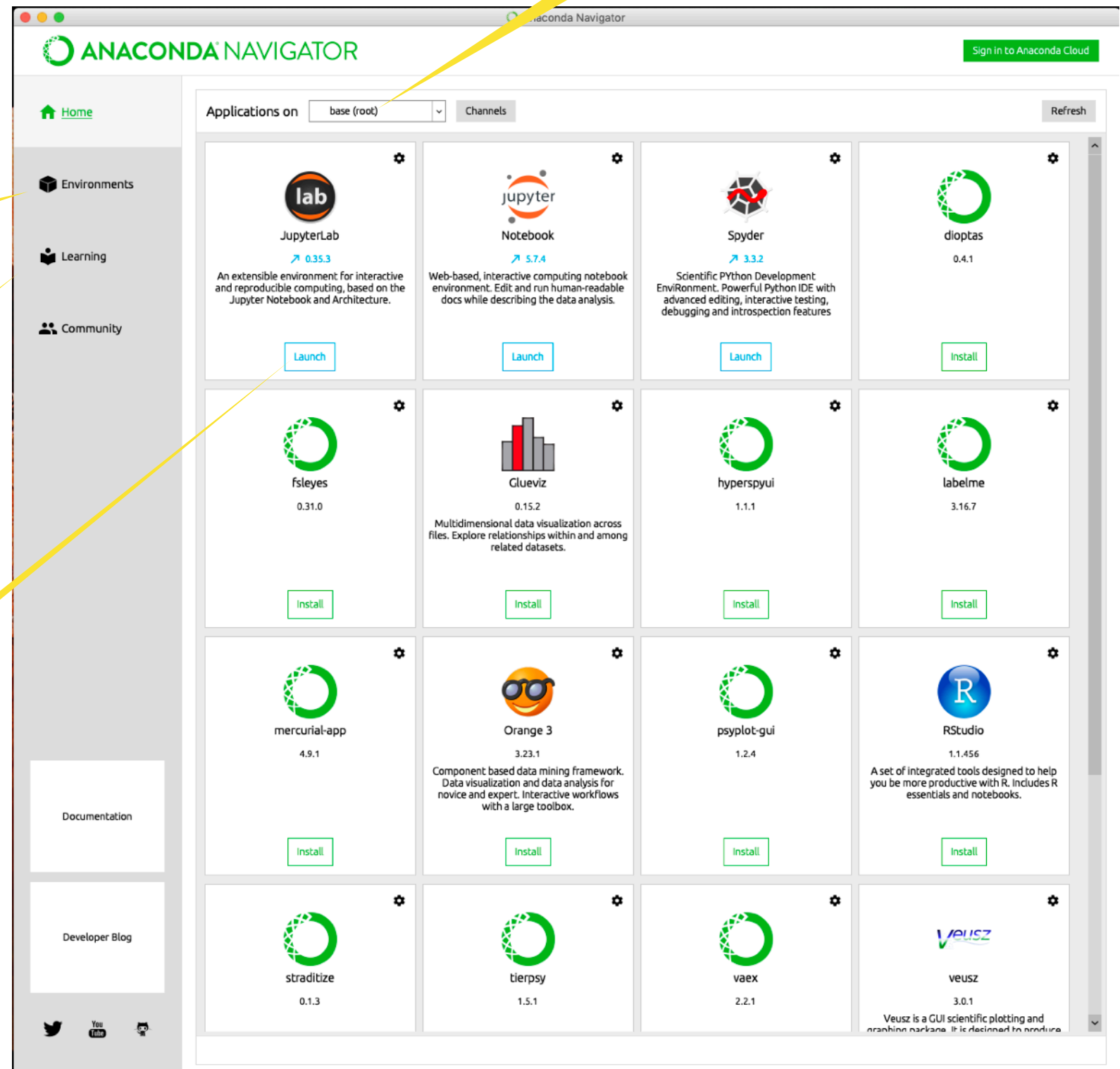
La fenêtre d'accueil

Gestion du contenu
des environnements de travail (ajout/
retrait/modifications)

Divers
documentations

L'outil qui nous
intéresse...

Choix d'un environnement
de travail





Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab



CentraleSupélec

L'environnement par défaut

Les outils (packages)
installés vs existants et à
télécharger

L'onglet de gestion
des environnements

C'est à dire :

Un ensemble complet

d'outils par exemple

dédiés à Python 2.x

ou à Python 3.x

ou à un client particulier...

Name	Description	Version
anaconda_depends		2019.03
_ipyw_lab_nb_ex...	A configuration metapackage for enabling anaconda-bundled jupyter extensions	0.1.0
alabaster	Configurable, python 2+3 compatible sphinx theme.	0.7.12
anaconda	Simplifies package management and deployment of anaconda	2018.12
anaconda-client	Anaconda.org command line client library	1.7.2
anaconda-project	Tool for encapsulating, running, and reproducing data science projects	0.8.2
appnope	Disable app nap on os x 10.9	0.1.0
appscript	Control applescriptable applications from python	1.0.1
asn1crypto	Python asn.1 library with a focus on performance and a pythonic api	0.24.0
astroid	A abstract syntax tree for python with inference support.	2.1.0
astropy	Community-developed python library for astronomy	3.1
atomicwrites	Atomic file writes.	1.2.1
attrs	Attrs is the python package that will bring back the joy of writing classes by relieving you from the drudgery of implementing object protocols (aka dunder methods).	18.2.0
babel	Utilities to internationalize and localize python applications	2.6.0
backcall	Specifications for callback functions passed in to an api	0.1.0
backports		1.0
backports.os	Backport of new features in python's os module	0.1.1
backports.shutil_g...	A backport of the get_terminal_size function from python 3.3's shutil.	1.0.0
beautifulsoup4	Python library designed for screen-scraping	4.6.3
bitarray	Efficient arrays of booleans -- c extension	0.8.3
bkcharts	High level chart types built on top of bokeh	0.2
blas		1.0
blaze	Numpy and pandas interface to big data	0.11.3
bleach	Easy, whitelist-based html-sanitizing tool	3.0.2
blinker	Fast, simple object-to-object and broadcast signalling	1.4
blosc	A blocking, shuffling and loss-less compression library that can be faster than 'memcpy'	1.14.4
bokeh	Statistical and novel interactive html plots for python	1.0.2

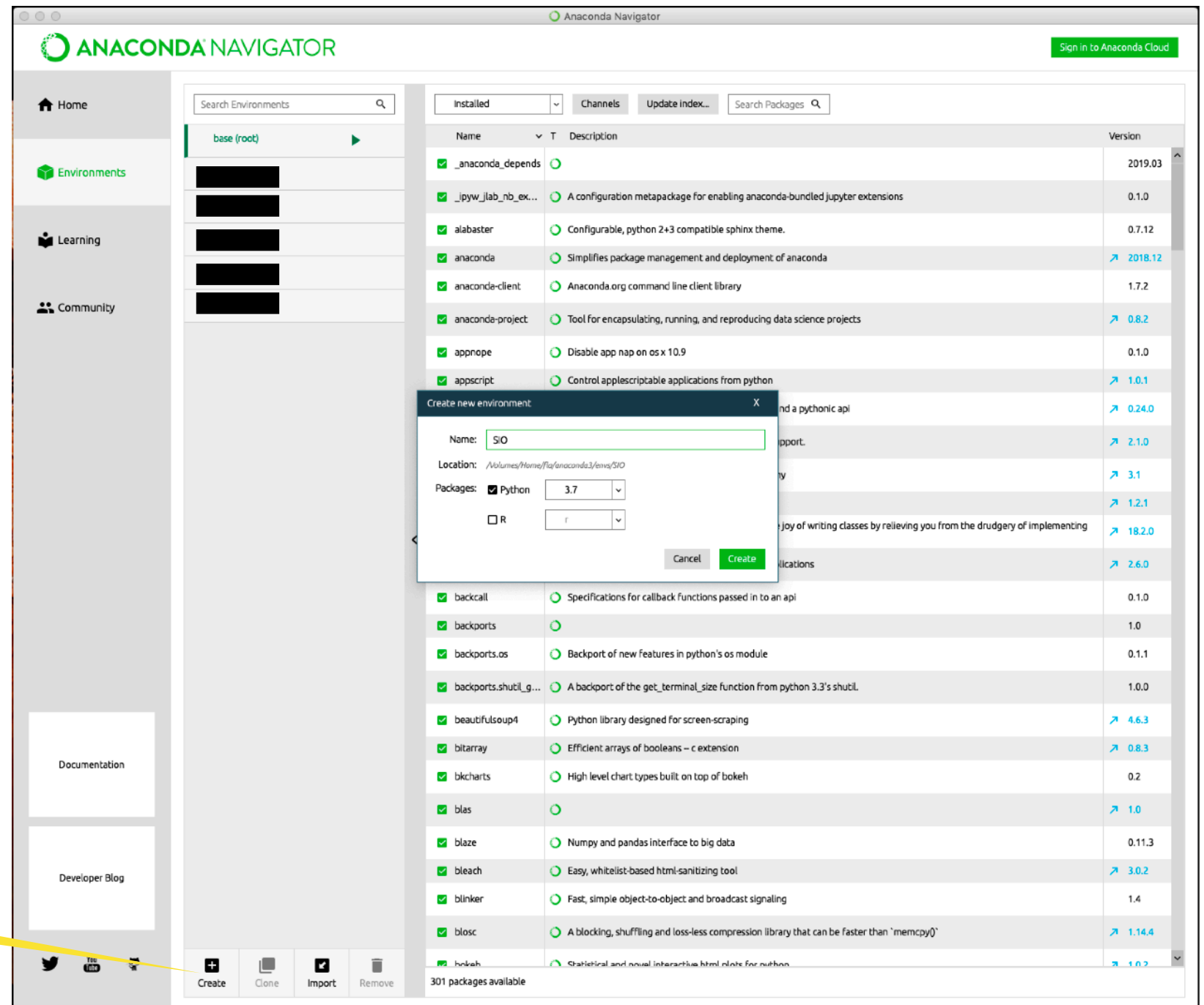


Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab

Il est recommandé
de se créer un
environnement
différent par
projet pour pouvoir
faire évoluer les
outils de manière
indépendante...

Créer un nouvel
environnement

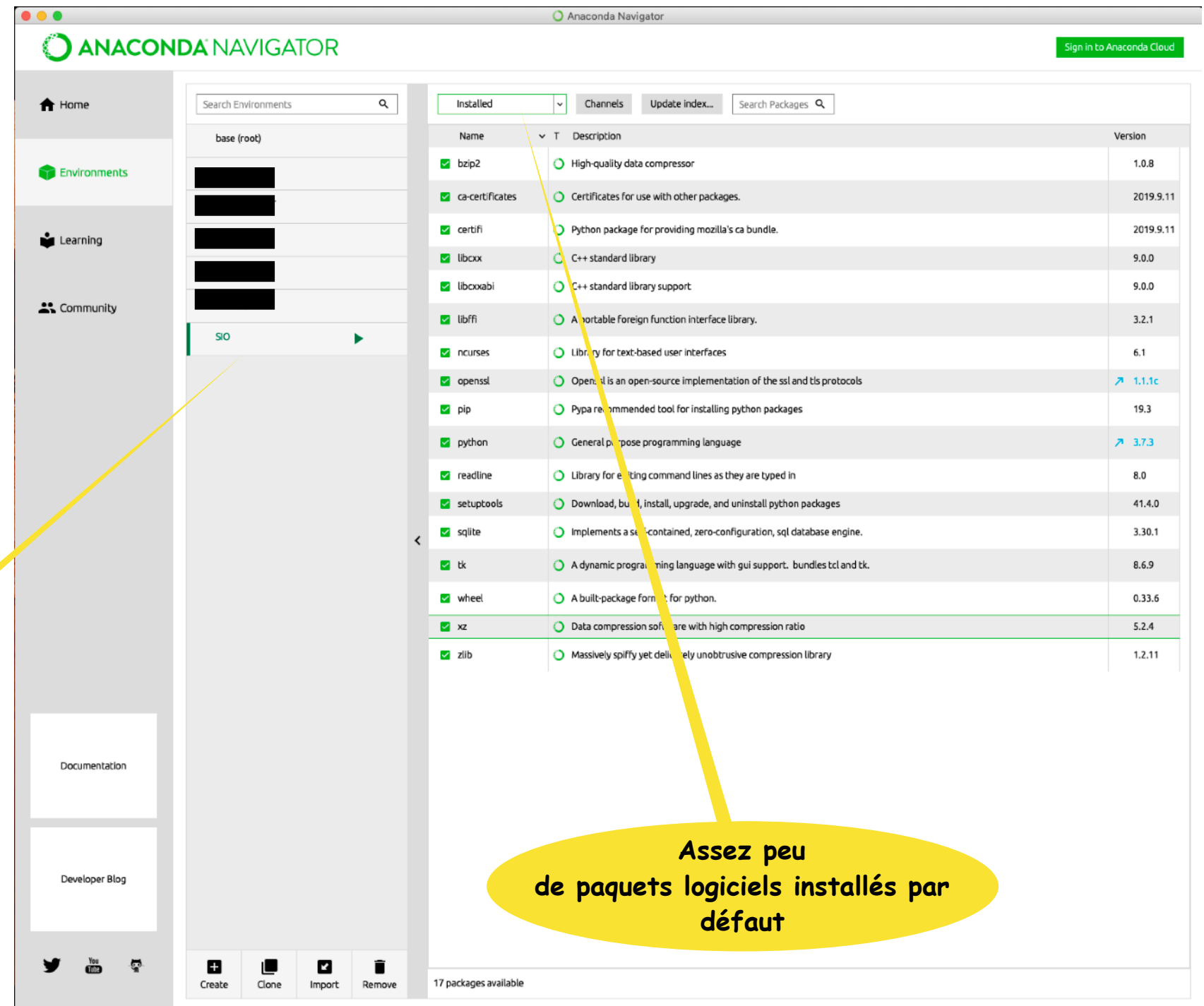




Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab

Le nouvel environnement
SIO juste créé :



The screenshot shows the Anaconda Navigator application window. On the left sidebar, the 'Environments' tab is selected, showing a list of environments. The 'SIO' environment is highlighted at the bottom of the list. A yellow arrow points from the text 'Nouvelle environnement "SIO"' to this entry. The main panel displays the 'base (root)' environment details, showing a list of installed packages. A yellow arrow points from the text 'Assez peu de paquets logiciels installés par défaut' to this list. The list includes packages like bzip2, ca-certificates, certifi, libcxx, libcxxabi, libffi, ncurses, openssl, pip, python, readline, setuptools, sqlite, tk, wheel, xz, and zlib. At the bottom of the main panel, it says '17 packages available'.

Name	Description	Version
bzip2	High-quality data compressor	1.0.8
ca-certificates	Certificates for use with other packages.	2019.9.11
certifi	Python package for providing mozilla's ca bundle.	2019.9.11
libcxx	C++ standard library	9.0.0
libcxxabi	C++ standard library support	9.0.0
libffi	A portable foreign function interface library.	3.2.1
ncurses	Library for text-based user interfaces	6.1
openssl	OpenSSL is an open-source implementation of the ssl and tls protocols	1.1.1c
pip	PyPA recommended tool for installing python packages	19.3
python	General purpose programming language	3.7.3
readline	Library for editing command lines as they are typed in	8.0
setuptools	Download, build, install, upgrade, and uninstall python packages	41.4.0
sqlite	Implements a self-contained, zero-configuration, sql database engine.	3.30.1
tk	A dynamic programming language with gui support. bundles tcl and tk.	8.6.9
wheel	A built-package format for python.	0.33.6
xz	Data compression software with high compression ratio	5.2.4
zlib	Massively spiffy yet delightfully unobtrusive compression library	1.2.11

Nouvelle
environnement "SIO"

Assez peu
de paquets logiciels installés par
défaut

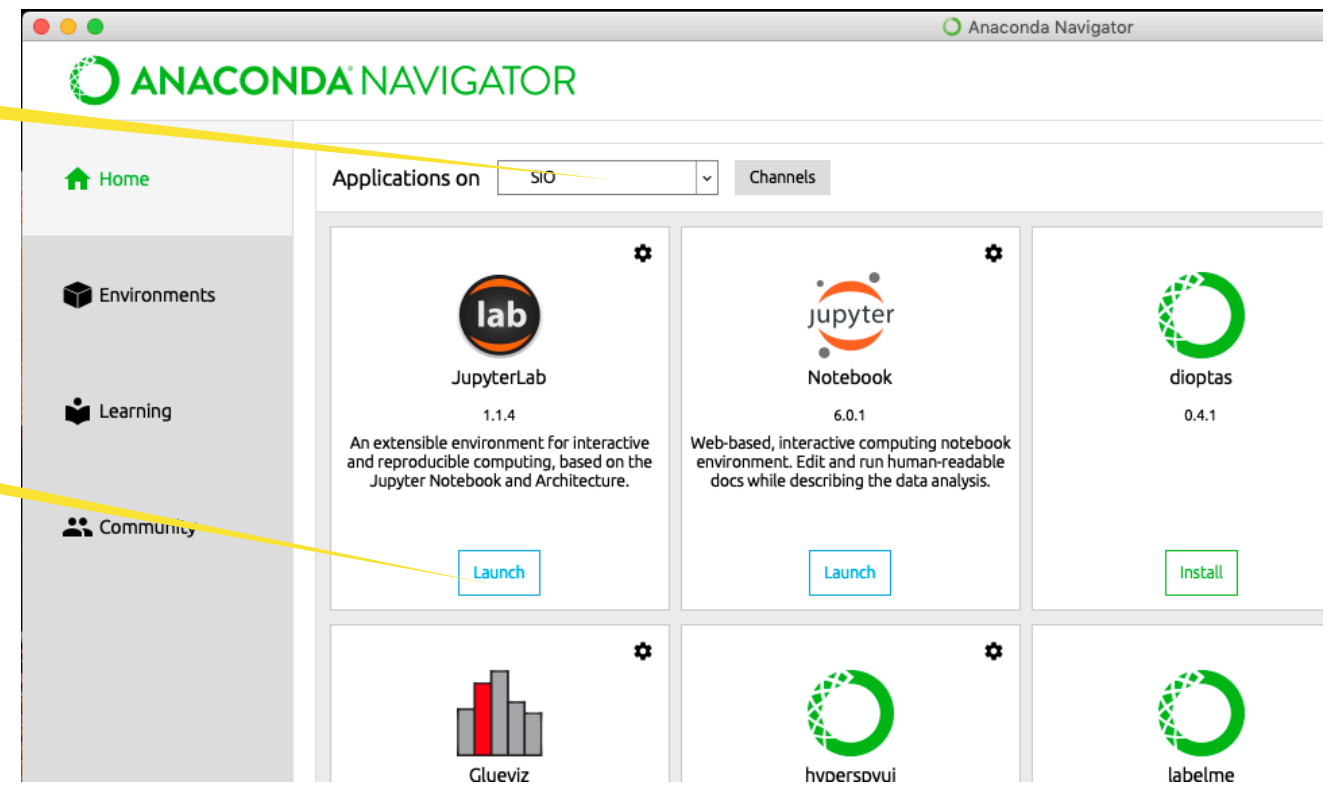


Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab

À
sélectionner plutôt que "Base"

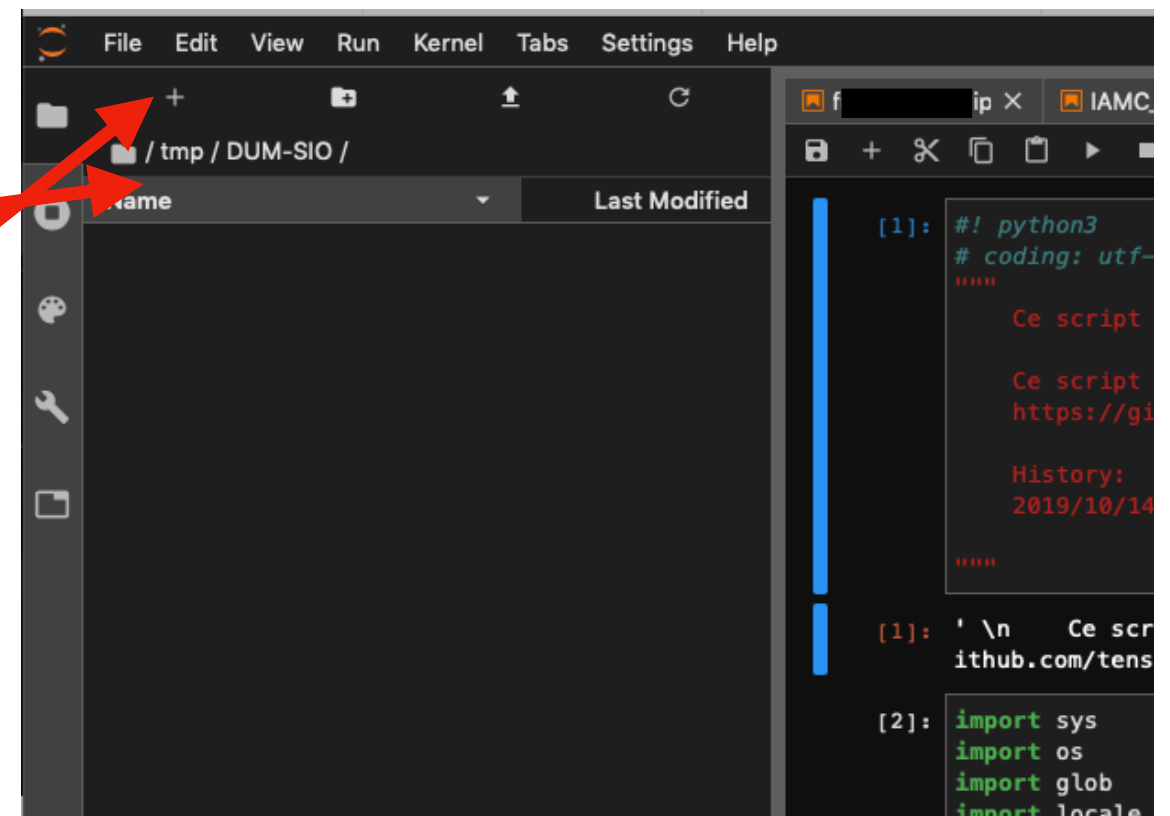
Cliquer
sur "Install" la première fois !



L'IDE s'ouvre dans le browser
par défaut.

Il faut choisir un répertoire pour
son projet

Puis créer un fichier (ici le
répertoire est vide)





Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab

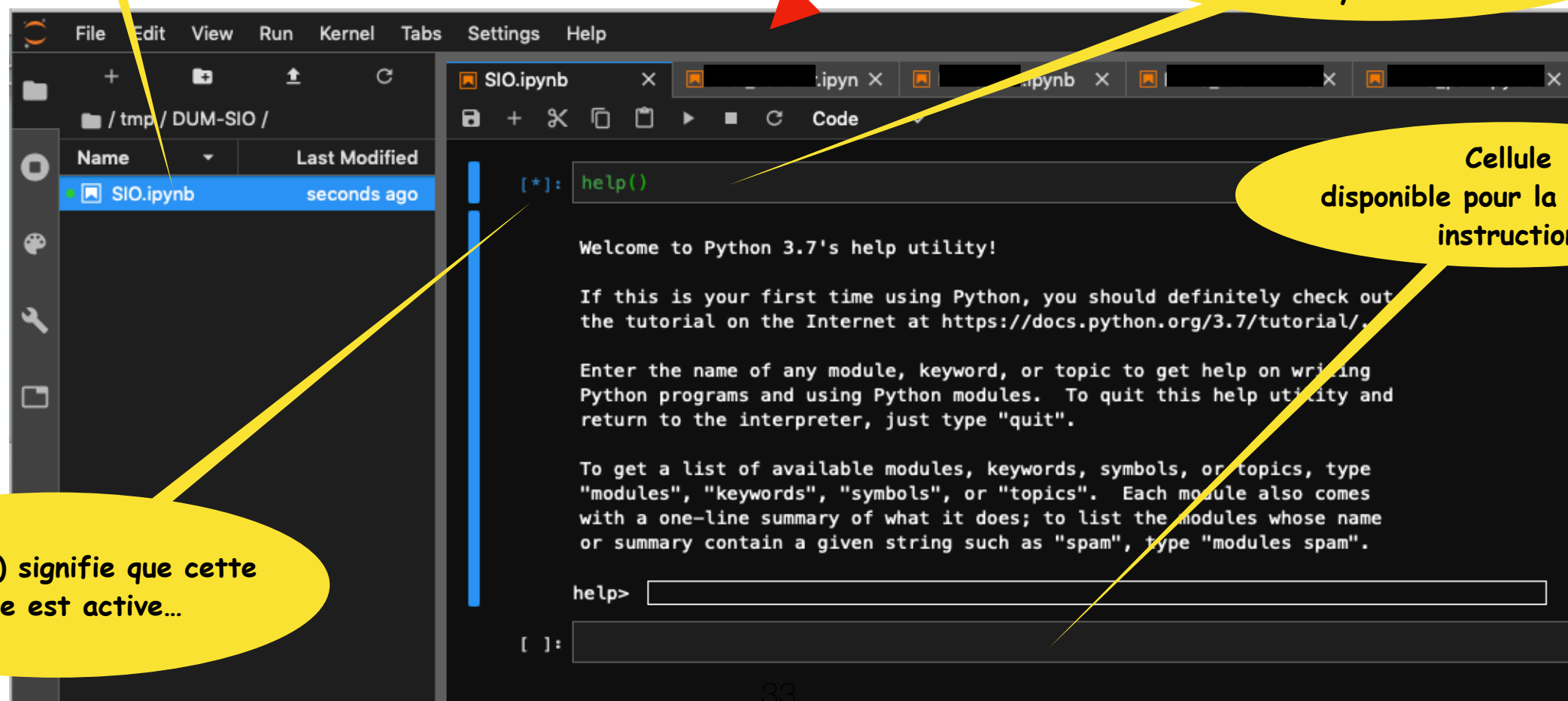
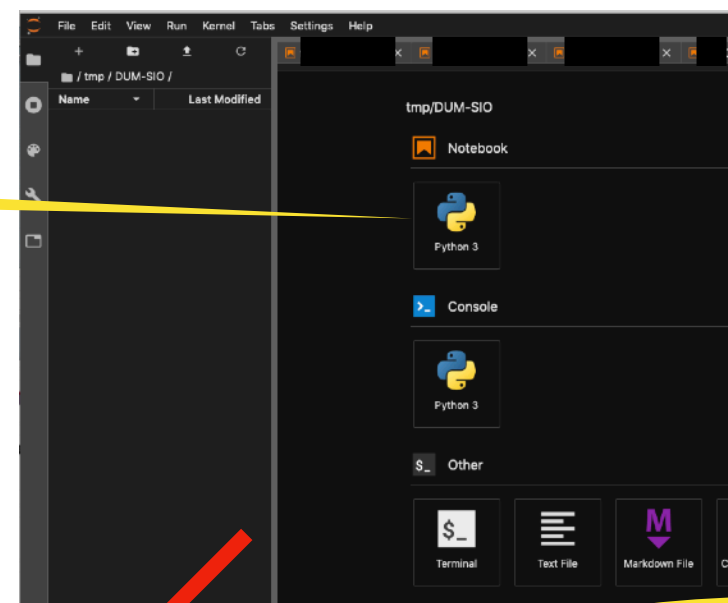
À sélectionner
Python 3

Le premier "notebook" de
l'environnement, après
renommage.

"shift-enter"
pour exécuter cette instruction
Python !

L'étoile (*) signifie que cette
cellule est active...

Cellule
disponible pour la prochaine
instruction





Mastère SIO - Octobre 2019 - Projet Fil Rouge

JupyterLab

Suite avec l'utilisation des *APIs* pour se connecter sur RosettaHUB