Bienvenue dans le module : Administration de bases de données PostgreSQL

Bienvenue à toutes et à tous !

Dans un monde où chaque application, site web ou service numérique repose sur la donnée, comprendre et maîtriser les bases de données est devenu une compétence essentielle pour tout informaticien. Les bases de données sont aujourd'hui le cœur du système d'information : elles centralisent, sécurisent et structurent les données pour permettre leur exploitation, leur analyse et leur valorisation. Qu'il s'agisse d'un réseau social, d'une plateforme de e-commerce, d'un service bancaire ou d'une application scientifique, aucune solution numérique ne peut fonctionner sans une base de données fiable et bien administrée.

L'administrateur de bases de données (ou DBA) joue donc un rôle clé : il garantit la disponibilité, la performance, la sécurité et la cohérence de ces systèmes. Avec PostgreSQL, l'un des SGBD open source les plus puissants et les plus utilisés, vous allez explorer concrètement les principes fondamentaux à connaitre avec des travaux pratiques.

OBJECTIFS PEDAGOGIQUES

Ce module vous guidera à travers les grands principes de l'administration PostgreSQL, depuis la conception jusqu'à la sécurisation des systèmes. Vous apprendrez à :

- Modéliser et concevoir des architectures optimales à l'aide des méthodologies UML et Merise.
- Utiliser le DDL avancé pour concevoir des bases performantes : partitions, index complexes, contraintes.
- Automatiser les tâches d'administration avec des scripts SQL ou des outils dédiés.
- Contrôler les accès et gérer les rôles (DCL) pour garantir la confidentialité et l'intégrité des données.
- Adapter les architectures selon les charges de travail : OLTP, OLAP, ou bases dénormalisées.
- Mettre en œuvre des stratégies de sécurisation contre les menaces (injections SQL, escalade de privilèges).
- Gérer les sauvegardes et restaurations pour assurer la résilience et la continuité du service.

1 - Prérequis

Les travaux pratiques du module s'appuieront sur un environnement virtualisé avec une machine virtuelle déjà configurée.

Pour ce module, vous devrez disposer d'un ordinateur ayant **au moins** les ressources suivantes :

- Plus de 4 Go de RAM
- Plus de 4 cœurs
- Plus de 5 Go d'espace disque libre

En termes de privilèges, vous devrez disposer des droits d'administrateur sur votre machine. Pour faciliter le déploiement de votre VM, nous nous appuierons sur la solution Vagrant. L'ensemble des outils nécessaires à ce module sont décrits dans ce document.

Toutes les commandes précédées du mot 'bash' représenteront des commandes à exécuter dans un client shell ou PowerShell selon votre système d'exploitation. Par exemple, la cellule suivante indique que la commande ls devra être exécutée dans un client shell :

```
bash
ls
```

2 - Environnement virtuel

Pour déployer et configurer votre machine virtuelle (VM), je s'occupe de tout tu s'occupe de rien 😉.

Afin d'assurer la compatibilité de la formation avec les différentes architectures de processeurs, y compris pour les étudiants utilisant des Mac équipés de puces ARM de nouvelle génération, nous adopterons une solution combinée basée sur Vagrant et d'une solution de virtualisation VMware ou VirtualBox. Cela nous permettra de contourner les problèmes d'environnement et de nous focaliser sur l'essentiel de la formation.

3 - Logiciel de virtualisation : VMware ou VirtualBox

Cette section décrit les étapes d'installation selon votre préférence ou votre environnement existant.

3.1 VirtualBox

Télécharger et installer VirtualBox selon que vous soyez sur Windows ou MAC OSX.

https://www.virtualbox.org/wiki/Downloads

3.2 VMware

Si vous ne disposez pas de VMware, il est recommandé de priviliger VirtualBox.

Télécharger et installer VMware Workstation ou VMWare Fusion selon que vous soyez sur Windows ou MAC OSX. Le téléchargement n'est possible qu'après avoir créer un compte.

https://knowledge.broadcom.com/external/article?articleNumber=309355

Notez qu'il est recommandé d'utiliser VMware Workstation ou VMWare Fusion dans le cas ou vous ne disposez pas des licences mais il est tout à fait possible d'utiliser VMware player (gratuit).

4 Vagrant

Vagrant est un logiciel anciennement libre et open-source pour la création et la configuration des environnements de développement virtuels.

4.1 Installation de Vagrant

Étape 1 : Télécharger le fichier d'installation de Vagrant depuis le site officiel.

Étape 2 : Installez Vagrant à partir du fichier téléchargé en fonction de votre système d'exploitation.

L'exécutable de Vagrant sera automatiquement ajouté à votre chemin système, vous permettant ainsi d'utiliser la commande vagrant à partir d'un terminal ssh ou powershell si vous êtes sous Windows.

Étape 3 : Installer le plugin Vagrant pour VMware

Uniquement pour les étudiants sous VMware, veuillez installer les utilitaires :

Vagrant VMware Utility

Pour permettre à Vagrant de communiquer avec VMware vous devrez installer le plugin vagrant-vmware-desktop en exécutant la commande suivante :

```
bash
vagrant plugin install vagrant-vmware-desktop
```

Vous pouvez vérifier l'installation du plugin avec la commande suivante :

```
bash
vagrant plugin list
```

Votre environnement de virtualisation est à présent prêt.

Étape 4 : Une fois l'installation terminée, redémarrez votre ordinateur pour vous assurer que les variables d'environnement ont bien été mises à jour.

Étape 5 : Pour vérifier l'installation de Vagrant, exécutez la commande suivante dans un terminal ssh ou powershell pour obtenir la version de votre vagrant :

```
bash
vagrant --version
```

5 Support de cours et TP

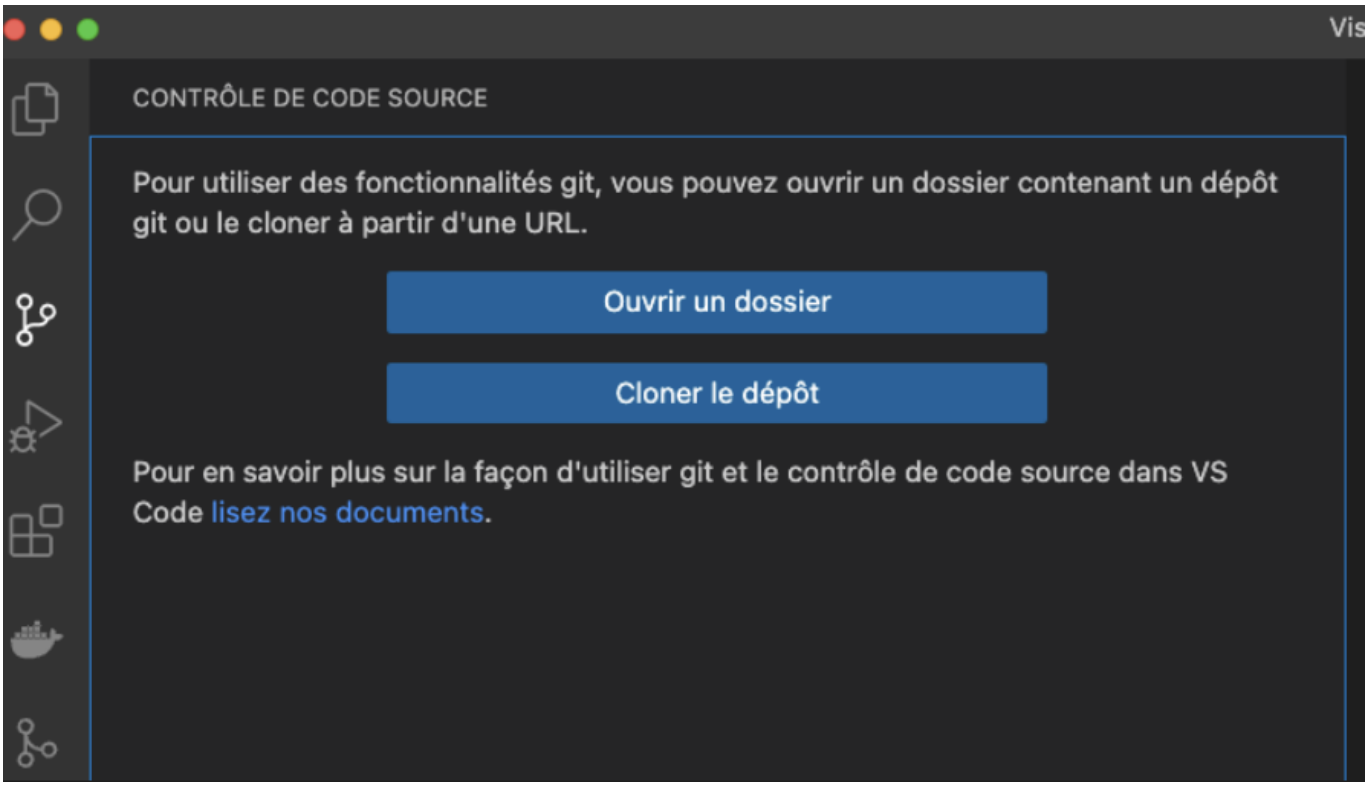
Introduction

Les TPs et les cours sont hébergés dans un projet gitlab. Pour accéder au cours et TP, vous devrez cloner le projet gitlab en local sur votre ordinateur avec un client git. Le clonage du projet git peut être réalisé à partir d'un client git quelconque comme votre IDE.

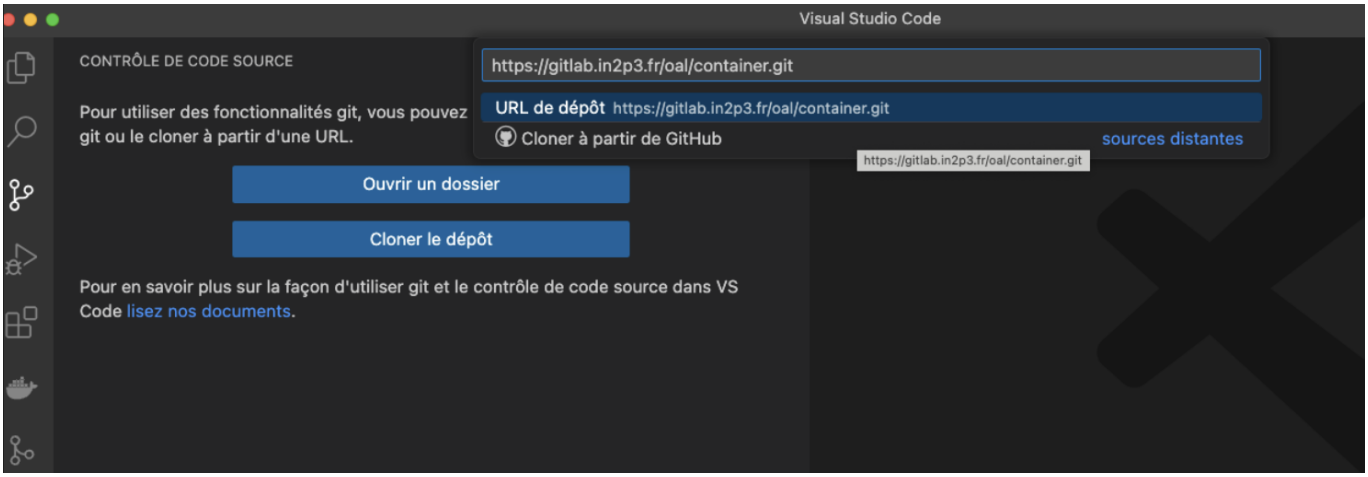
5.1 Clonage du repo gitlab

A partir de votre client `git` cloner le repo gitlab de la formation. La procédure décrite ci-dessous s'applique uniquement pour VS Code.

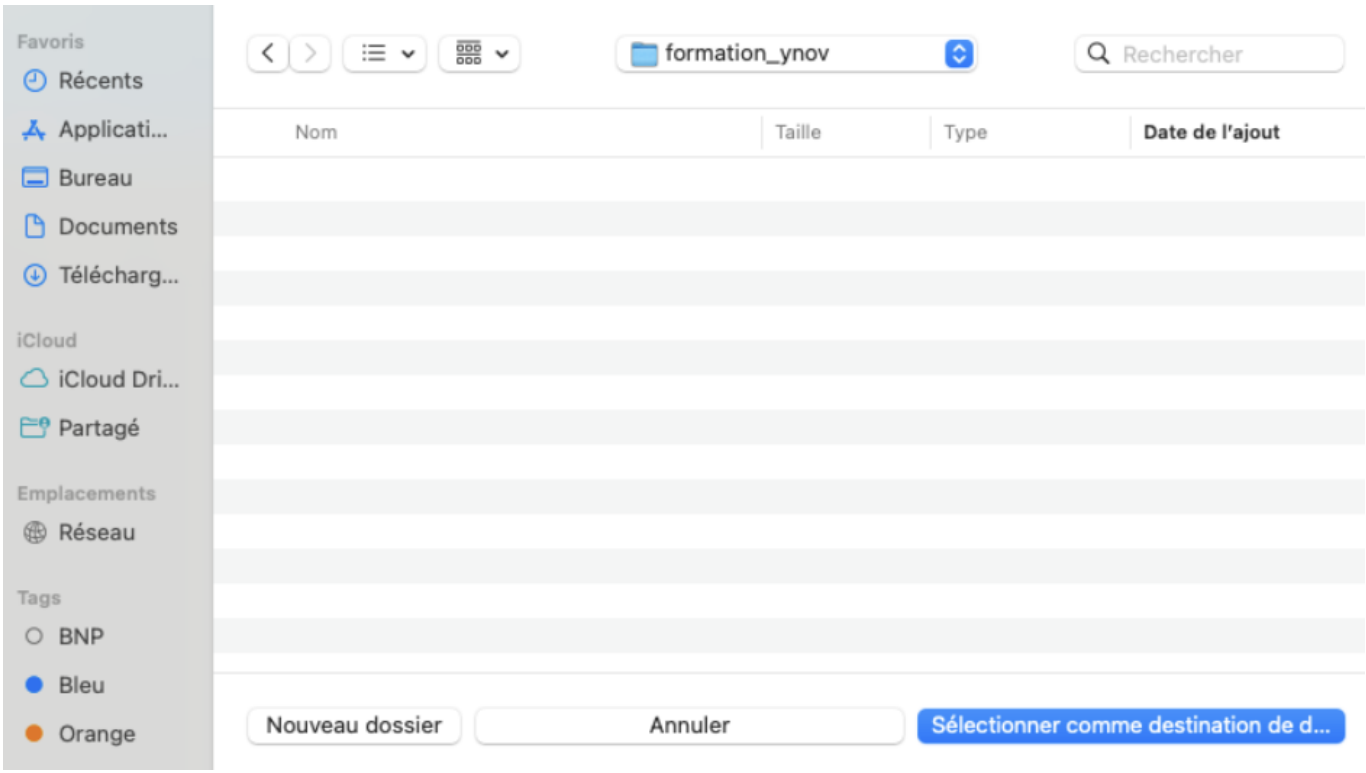
Cliquer dans le menu de gauche sur l'onglet `Contrôle de code source`:



Cliquer sur le bouton `Cloner le dépôt`, indiquer le dépôt `https://ynov2025:glpat-xo_-OEFqUzl_4V_lSJ0DgG86MQp1Omh4ZAk.01.0z03jeqc7@gitlab.in2p3.fr/oal/instance/administration-bdd-2025.git` et cliquer sur `URL de dépôt` :



Une boite de dialogue s'ouvre pour identifier l'emplacement où sera stocké votre repo local :
L'emplacement selectionné sera référencé comme le `GITDIR`.



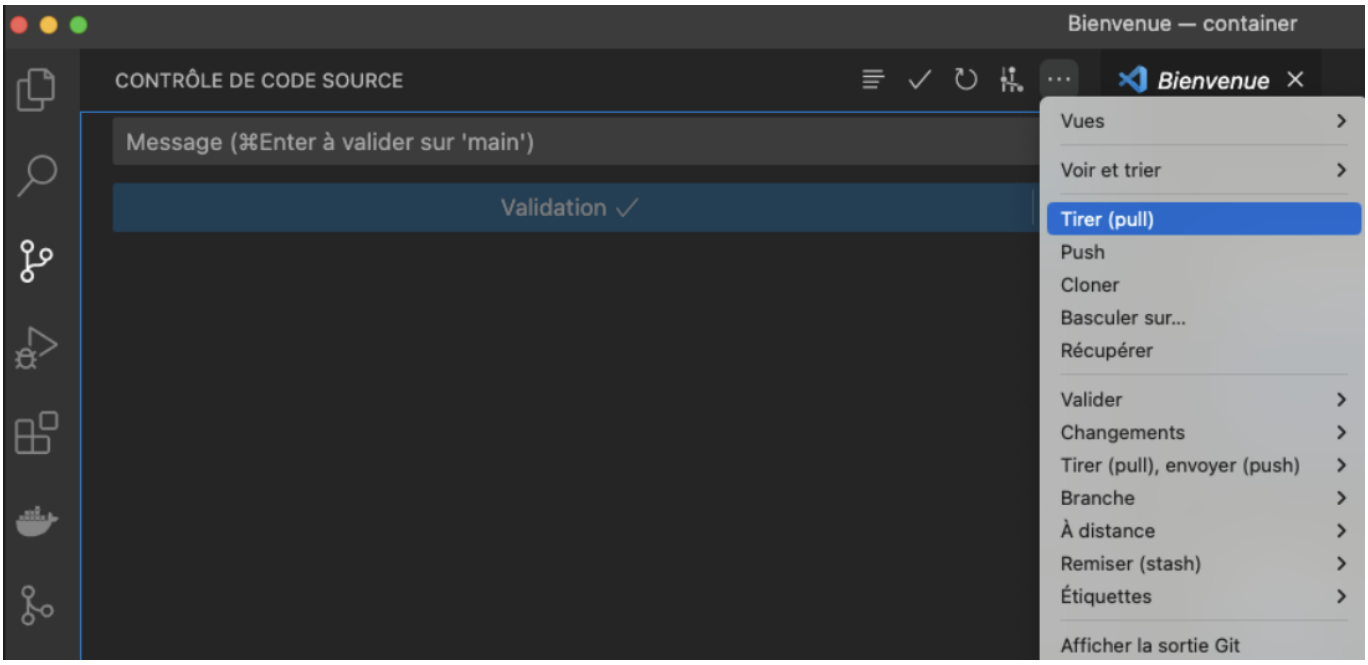
L'ensemble du contenu de la formation sera téléchargé sur votre ordinateur :

- Les supports de cours
- Les TP
- La configuration de votre VM à appliquer par vagrant

5.2 Mise à jour des supports

L'ensemble des éléments du projet sera régulièrement mis à jour tout au long du module, vous serez donc amené à mettre à jour votre repo git local.

Pour mettre à jour le contenu de la formation, il vous suffira de vous rendre dans l'onglet `Contrôle de code source` puis de `tirer` les derniers changements :



Notez que pour ne pas entrer en conflit avec les mises à jour qui seront apportées tout au long du module, il vous est demandé avant de travailler sur un TP de le copier dans le répertoire MES_TPS.

Bravo, vous êtes à présent prêt à suivre le module.

6 Déploiement de la VM

Pour les TP, nous utiliserons une à plusieurs machines virtuelles. la VM s'appuiera sur une image disponible sur vagrantCloud. Pour faciliter le déroulement de la formation, une configuration de la VM est disponible dans votre repo gitlab dans le répertoire vagrant/vmware.

Pour déployer votre VM, il vous suffit d'ouvrir un terminal powershel/ shell puis de lancer la commande suivante :

- Si vous utilistez virtualBox ;

```
cd $GITDIR/vagrant/virtualbox
vagrant up
```

- Si vous utilistez VMware ;

```
cd $GITDIR/vagrant/vmware
vagrant up
```

Cette commande s'occupe de téléchager l'image de la VM, de la configurer puis de la démarrer.

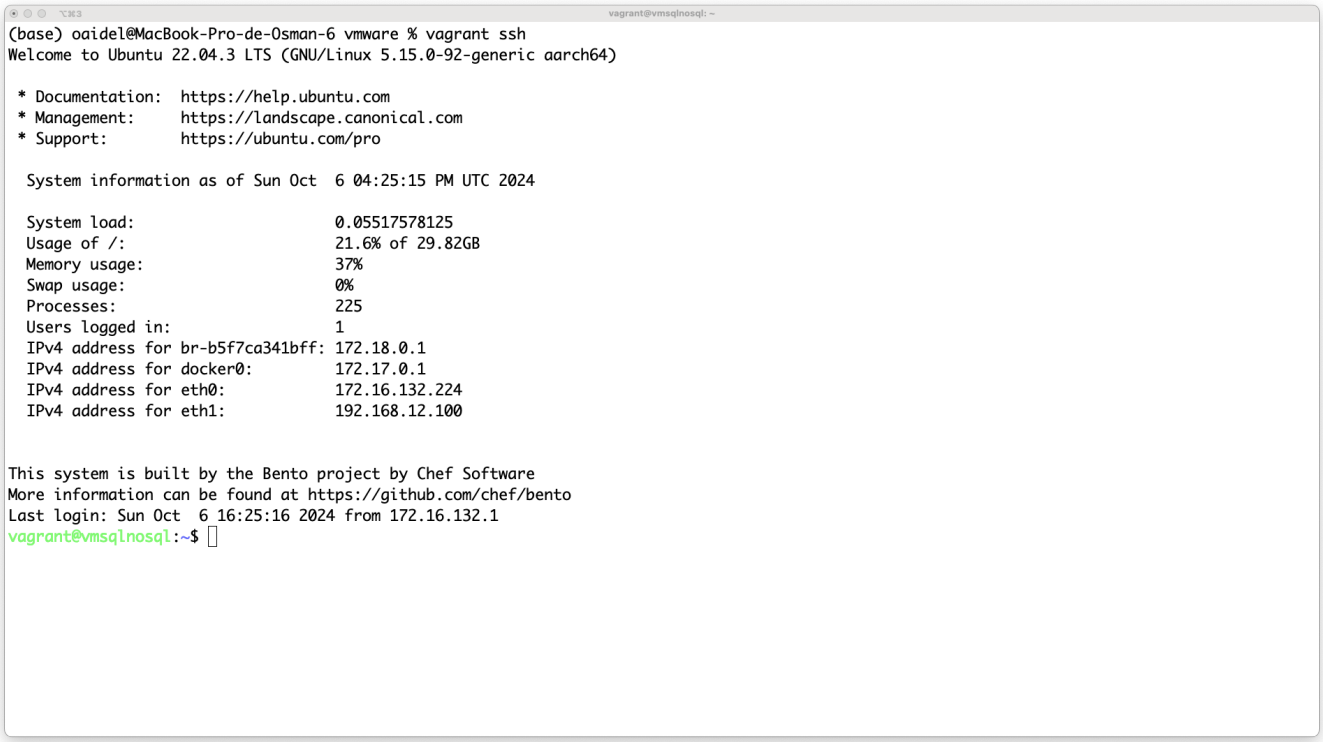
Vérifier l'état de votre VM avec la commande suivante :

```
vagrant global-status
```

6.1 Tester votre VM

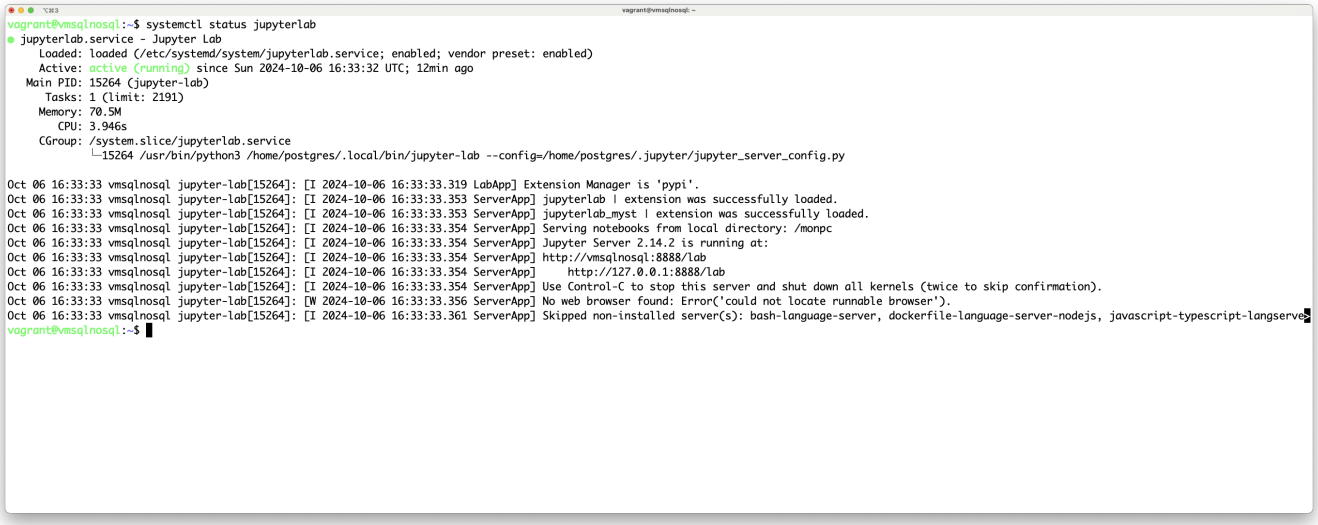
Pour vérifier le bon fonctionnement de votre VM, ouvrez un terminal (vous pouvez utiliser celui utiliser pour le vagrant up) pour vous positionner dans `$GITDIR/vagrant/virtualbox/` et exécuter la commande `vagrant ssh` pour vous connecter en ssh sur votre VM.

```
cd $GITDIR/vagrant/virtualbox/
vagrant ssh
```

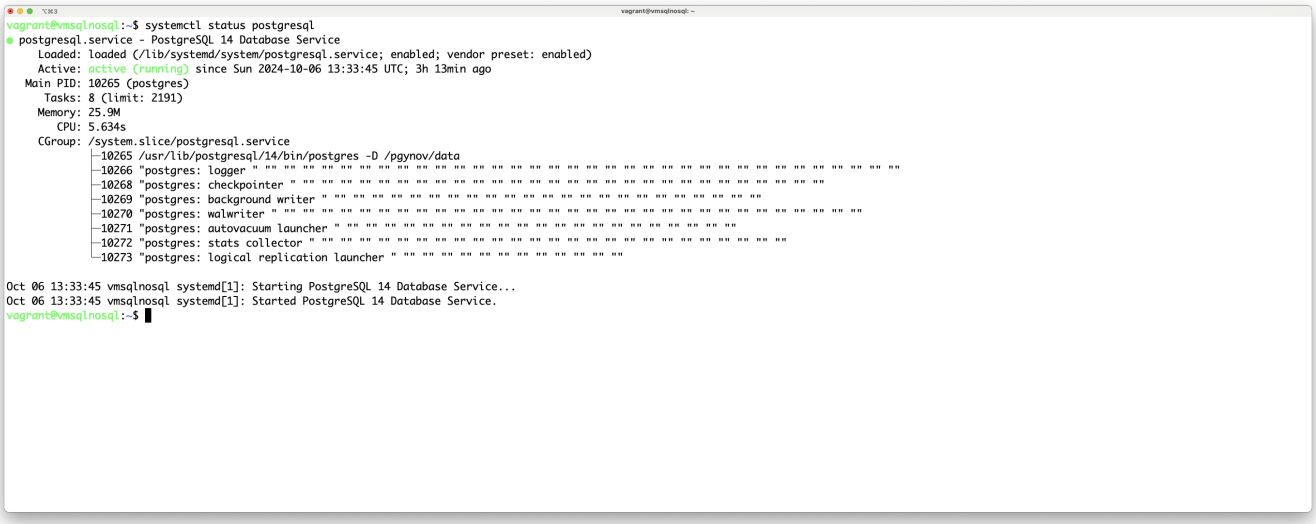


Vérifier que les services Postgresql, Jupyterlab et Docker sont en cours d'exécution :

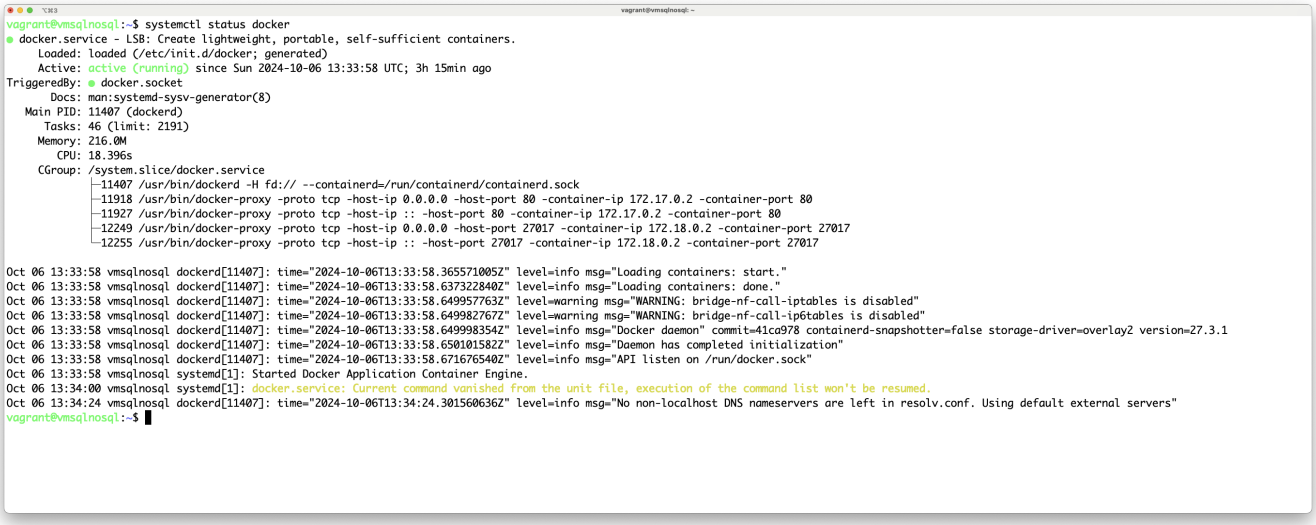
```
systemctl status jupyterlab
```




```
systemctl status postgresql
```



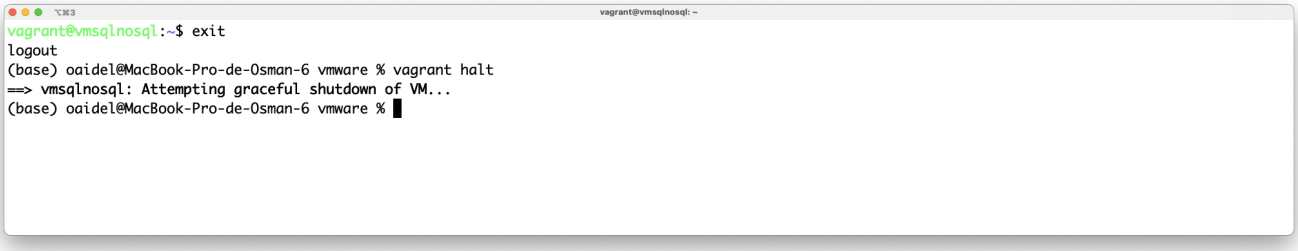

```
systemctl status docker
```



Bravo votre environnement est opérationnel !!!

Vous pouvez à présent arrêter votre VM en utilisant la commande `vagrant halt` :

```
exit
vagrant halt
```



A la fin de la formation, vous pourrez supprimer votre VM avec la commande suivante :

- Si vous utilistez virtualBox ;

```
cd $GITDIR/vagrant/virtualbox
vagrant destroy
```

- Si vous utilistez VMware ;

```
cd $GITDIR/vagrant/vmware
vagrant destroy
```

Vous êtes à présent prêt pour le module !!!