

Document Technique : Mise en place de l'environnement de simulation

Le document suivant explique les étapes à suivre pour mettre en place l'environnement de simulation pour le TER atelier flexible.

Attention pour pouvoir faire tourner la simulation, vous devez posséder un ordinateur ou une machine virtuelle tournant sous Ubuntu 14.04.5 ou supérieur en **64-bits**. Vous devez également être sur une session dont le nom est le suivant : **projn7cellule**. Il est cependant possible (mais non recommandé) de travailler sur une session possédant un nom différent, une commande devra cependant être rentrée. Elle sera donnée plus tard.

1. Installation de git

Pour installer git (dans le cas où ce n'est pas déjà fait, allez dans l'invité de commande et tapez la commande suivante :

```
sudo apt-get install git
```

2. Récupération du projet depuis GitHub

Une fois git installé, vous pouvez récupérer le dossier du projet directement sur internet via la commande suivante (une connexion à internet est donc requise).

```
git clone https://github.com/FlorentSilva/PL_2018
```

Votre dossier se situera alors dans le dossier home de votre ordinateur. Avant de pouvoir lancer la simulation il est nécessaire de suivre encore quelques étapes.

3. Installation de ROS indigo

Pour faire tourner la simulation il est nécessaire de posséder ROS indigo. Pour l'installer, veuillez suivre les commandes suivantes :

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'
```

Cette commande permet de configurer l'ordinateur pour qu'il puisse recevoir des packages venant de ROS. Entrez ensuite :

```
sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116
```

Cette commande permet de mettre en place les clés serveur pour récupérer des packages depuis ROS. Entrez ensuite :

```
sudo apt-get update
```

Cette commande permet de vérifier que la liste des packages présents sur votre ordinateur est bien à jour. Ensuite entrez la commande suivante pour installer la version complète de ROS.7

```
sudo apt-get install ros-indigo-desktop-full
```

4. Installation de ModBus

Vous devez ensuite installer ModBus via la commande suivante :

```
sudo apt-get install libmodbus-dev
```

5. Mise en place du fichier bashrc

Ouvrez le fichier bashrc, pour cela entrez la commande suivante

```
cd && gedit .bashrc
```

Attention, si le fichier est vide, c'est que vous vous trouvez dans le mauvais dossier vérifiez d'être bien dans le dossier home pour rentrer la commande gedit (d'où l'utilité du cd juste avant).

Une fois que vous avez le fichier ouvert ajoutez les trois lignes suivantes :

```
source /opt/ros/indigo/setup.bash
export VREP_ROOT_DIR=~/.PL_2018/V-Rep/
source ~/.PL_2018/ros_ws/devel/setup.bash
```

Cela permet de sourcer sur les bons dossiers lors de la compilation des programmes. Une fois le fichier modifié sauvegardez-le et fermez tous les terminaux pour les changements soit pris en compte.

6. Lancement de la simulation

Une fois que tout ceci est fait vous pouvez lancer la simulation, pour cela rendez-vous dans le dossier du TER

```
cd PL_2018/
```

Ensuite lancez la simulation via la commande suivante :

```
./launch.sh
```

Normalement la simulation devrait marcher. Si ce n'est pas le cas, fermez tous les terminaux et relancez les deux commandes précédentes jusqu'à que la simulation fonctionne. En cours de simulation il se peut que les robots crashent, si c'est le cas, relancez simplement la simulation.

7. Recompilation

Une fois que vous aurez codé il faudra recompiler, pour cela entrez la commande suivante :

```
cd && cd PL_2018/ros_ws/ && catkin_make
```

Cette commande permet de se placer dans le dossier de compilation et d'exécuter la compilation, si vous êtes déjà dans le dossier ros_ws lancez simplement la commande suivante :

```
catkin_make
```

Bon Travail !