



## TP 02 : Créer un URDF et le mettre en mouvement

Encadrant : Khadimoullah Ramoth

Dans ce TP vous allez apprendre à créer un URDF (Unified Robot Description Format) qui est la façon dans ROS de décrire un robot.

A la fin de ce TP vous serez capable de modéliser un bras robotisé.

Dans ce TP Nous vous proposons de modéliser une main, et articuler ses doigts.

Il faudra rendre le package sous GitHub, avec un README.md expliquant les différents votre package. Vous pouvez écrire en anglais ou en français.

## Première Partie : Comprendre l'URDF

Dans cette partie nous allons voir comment créer un URDF.

Un URDF est un fichier xml, avec des balises spécifiques qui permettent de décrire un robot. Dans cette première partie nous allons voir les balises robot, link, visual, geometry et material.

Pour rappel une balise en xml est définie comme suit:

```
<balise 1>
  <balise2>
    <balise3 arg="value" arg2="0.4">
      .
      .
      .
    </balise3>
  </balise2>
</balise1>
```

Parfois une balise peut tenir sur une seule ligne <balise4 ... />

Dans un fichier URDF il faut commencer par un shebang xml : `<?xml version="1.0"?>`

Puis une balise robot (avec l'argument name, qui sera le nom du robot) qui contiendra toutes les descriptions du robot (toutes les balises).

La balise link permet de créer un élément dans le robot cette balise link contiendra plusieurs balises dont visual, geometry material ou encore origin

1. Dans votre catkin workspace, créer un package udm\_urdf avec pour dépendance roscpp, urdf, std\_msgs, geometry\_msgs, sensors\_msgs.
2. Dans le package udm\_urdf créer un dossier urdf.
3. Dans le dossier urdf créer les fichiers suivants urdf\_cylinder.urdf, box.urdf, sphere.urdf, mesh.urdf
4. Modifier cylinder.urdf afin de faire afficher un cylindre de hauteur 10cm et de rayon 5cm.
5. Modifier box.urdf pour afficher un pave de 10x15x5cm
6. Modifier sphere.urdf pour afficher une sphere de rayon 10cm
7. On va maintenant modifier mesh.urdf
  - a) Créer dans le package udm\_urdf un dossier mesh
  - b) Aller à l'adresse suivante:  
<https://cad.onshape.com/documents/90c373b34c1f3bd23121eba5/w/84ce1e23bc20e257a735c19b/e/ab48b8ccbbbd88e1fb3d0565>
  - c) Exporter le model 3D en stl (dans Parts clique gauche sur frame puis export)
  - d) Puis modifier mesh.urdf afin d'afficher le model.
  - e) Pour vérifier que les fichiers urdf sont correctes, utiliser l'outil check\_urdf, si cela n'est pas installé, sudo apt-get install liburdfdom-tools
8. Pour visualiser dans rviz, créer un dossier launch dans udm\_urdf
  - a) Créer visualize\_urdf.launch
  - b) Ajouter un arg model qui aura pour valeur par défaut  
`"$(find udm_urdf)/urdf/cylinder.urdf"`
  - c) Ajouter un param robot\_description ajouter l'argument  
`command="$(find xacro)/xacro.py $(arg model)"`

- d) Ajouter le noeud joint\_state\_publisher\_gui qui se trouve dans le package du même nom
- e) Ajouter le noeud robot\_state\_publisher qui se trouve dans le package du même nom
- f) Ajouter le noeud rviz qui se trouve dans le package du même nom avec args qui vaut "-d \$(find urdf\_tutorial)/rviz/urdf.rviz"
- g) Lancer le launch file
- h) `roslaunch udm_urdf visualize_urdf.launch model :='$(find udm_urdf)/urdf/cylinder.urdf'`

## Partie Deux : Description de la main

Dans un premier temps nous allons créer un doigt

9. Créer un fichier main.urdf

10. Créer un box qui fait la taille de la paume d'une main

Nous allons maintenant étudier comment ajouter d'autres formes dans un urdf

11. Ajouter une balise link avec pour nom index\_phalanx\_base

12. Ajouter une balise joint avec le type fixed, la paume sera le parent et l'index\_phalanx\_base sera le child.

13. Ajouter une geometry cylinder qui fait la taille de la première phalange d'un index

14. Ajouter un matériel aux différentes pièces du système afin de le rendre plus lisible.

15. Visualiser votre urdf avec le launch file créé précédemment.

Il faut placer correctement la phalange et le joint entre la phalange et la paume.

16. Dans le joint ajouter la balise origine qui aura deux arguments xyz et rpy le premier étant la position cartésienne de la phalange et le second l'orientation yaw pitch roll. Il faudra indiquer la position du joint

17. Dans le joint ajouter la position du joint relatif au parent.

18. Ajouter le reste des phalanges pour compléter l'index.



## Partie 3 : Articuler le modèle

Nous allons maintenant articuler le modèle.

Pour ce faire il faut modifier les joints et remplacer fixed par continuous, prismatic, revolute, planar ou floating.

Continuous est un joint qui modélise un mouvement de rotation continue, sans limite comme des roues par exemple.

Les joints prismatic représentent les liaisons glissières, avec effort maximum, un upper et lower en mètre et une vitesse max.

Revolute sont les joints qui représentent les liaisons pivots avec des limitations: en effort, en angle (lower et upper) et en vitesse, contrairement à la liaison prismatic, les unités sont en radian.

En plus de modifier le type de joint, il faudra aussi ajouter une balise axis qui indique la direction de ou l'axe du mouvement.

19. Modifier les joints du fichier main.urdf afin de mettre des revolute joint.
20. Lancer le launch file et tester les joints
21. Terminer la main en ajoutant les doigts manquants.
22. Déposer sous Moodle un lien vers le projet GitHub qui contient le package udm\_urdf.