

# Présentation M.O.R.S.E.

El Jawad Alaa

09/12/2016

Qu'est-ce que M.O.R.S.E ?

Utilisation

Extension de MORSE

Merci

Qu'est-ce que M.O.R.S.E ?

# Definition

## The Modular OpenRobots Simulation Engine






Figure 1: MORSE

# Histoire

Créé par le **LASS-CNRS** (*Laboratoire d'Analyse et d'Architecture des Systèmes*).

Puis repris par d'autres universités (Georgia Tech, Dassault Aviation, INRIA, ISAE SUPAERO ...)

# En quelques mots (ou images)

- ▶ simulateur d'environnement 3D
- ▶ focus sur le rendering (  blender )
- ▶ Physique géré par  BULLET PHYSICS LIBRARY
- ▶ entierement écrit en  python powered  
`print("Hello, world!")`

# Installation

Sous linux ou OSX.

[http://www.openrobots.org/morse/doc/1.4/user/  
installation.html](http://www.openrobots.org/morse/doc/1.4/user/installation.html)

Utilisation



# Utilisation ultra-simple

## Ligne de commande

```
# Verification de l'installation  
$ morse check  
# Creation d'un 'workspace'  
$ morse create mysim  
# Lancement d'une simulation  
$ morse run mysim
```

# Utilisation ultra-simple

## Ligne de commande

```
# Verification de l'installation
$ morse check
# Creation d'un 'workspace'
$ morse create mysim
# Lancement d'une simulation
$ morse run mysim
```

## Architecture du 'workspace' créé

- ▶ **scripts**: scripts pour interagir avec vos simulations (contrôle, lecture des capteurs... etc)
- ▶ **src**: définitions et équations d'état des robots, capteurs, et actionneurs
- ▶ *default.py* : script python qui décrit la simulation

# Utilisation simple

## Script basique

```
from morse.builder import *  
# Ajout d'un robot  
robot = ATRV() # a remplacer par le robot voulu  
# Ajout d'un actionneur  
motion = MotionVW()  
robot.append(motion)  
# Ajout d'un capteur  
camera = VideoCamera()  
robot.append(camera)  
# Creation de l'environnement  
env = Environment('sandbox', fastmode=False)
```

# Utilisation simple

## Script basique

```
from morse.builder import *  
# Ajout d'un robot  
robot = ATRV() # a remplacer par le robot voulu  
# Ajout d'un actionneur  
motion = MotionVW()  
robot.append(motion)  
# Ajout d'un capteur  
camera = VideoCamera()  
robot.append(camera)  
# Creation de l'environnement  
env = Environment('sandbox', fastmode=False)
```

- ▶ Environnement peut être édité/créé sous Blender
- ▶ Liste de composants disponibles ici

# Interfacage avec l'exterieur

- ▶ ROS
- ▶ MOOS
- ▶ YARP
- ▶ Socket

# Interfacage avec l'exterieur

- ▶ ROS
- ▶ MOOS
- ▶ YARP
- ▶ Socket

## Utilisation des méthodes:

```
add_stream('') + add_service('') = add_interface('')  
ou robot.add_default_interface('')
```

# Interfacage avec l'exterieur

- ▶ ROS
- ▶ MOOS
- ▶ YARP
- ▶ Socket

## Utilisation des méthodes:

`add_stream('') + add_service('') = add_interface('')`  
ou `robot.add_default_interface('')`

## Exemple avec ROS:

```
camera.add_stream('ros')
```

## Tuto socket:

[Advanced\\_tutorials/request\\_tutorial.html](#)

## Extension de MORSE



# Créer son robot

1. `$ morse add robot <name> mysim`
2. Définir le modèle 3D sous Blender (voir ici)
3. Définir les équations d'états dans la fonction `default_action()`
4. On peut aussi appliquer directement des forces

**Pour plus d'info:** [Adding\\_a\\_robot.html](#)

# Ajouter d'autres composants

Lien

## Actionneurs

```
$ morse add sensor <name> mysim
```

## Capteurs

```
$ morse add actuator <name> mysim
```

Ensuite faut coder !

Merci