

# Projet : robot NAO

K. Boudjelaba

## Objectifs :

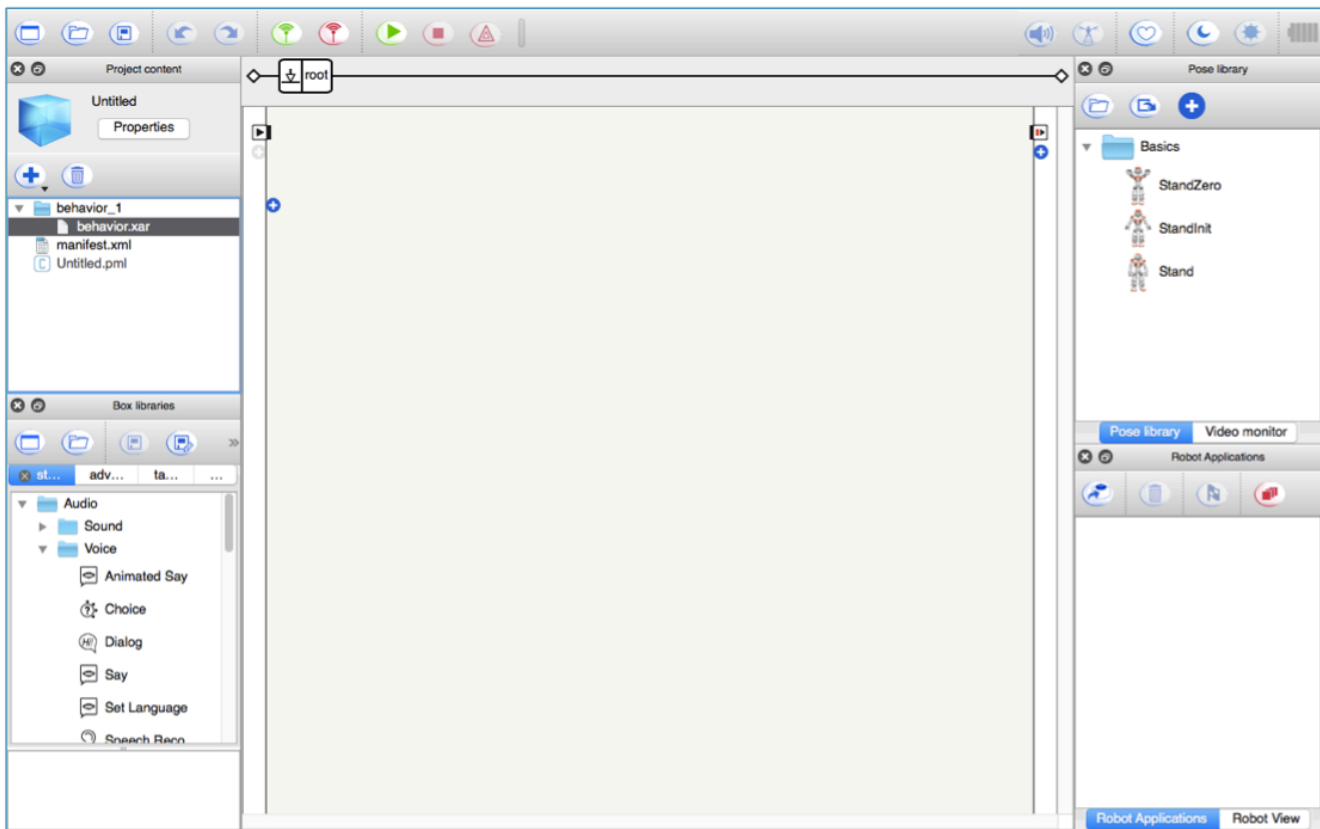
- Apprentissage du code grâce au robot humanoïde NAO
- Piloter, manœuvrer, faire parler le robot
- ...

## Table des Matières

<b>1</b>	<b>Présentation du logiciel Choregraphe</b>	<b>2</b>
<b>2</b>	<b>Exemple : Faire marcher NAO en utilisant la boîte Move To</b>	<b>6</b>
<b>3</b>	<b>Exercices</b>	<b>9</b>
3.1	Exercice 1 .....	9
3.2	Exercice 2 .....	9
3.3	Exercice 3 .....	9
<b>4</b>	<b>TP robot NAO</b>	<b>11</b>
4.1	TP1 : .....	11
4.2	TP2 : .....	11

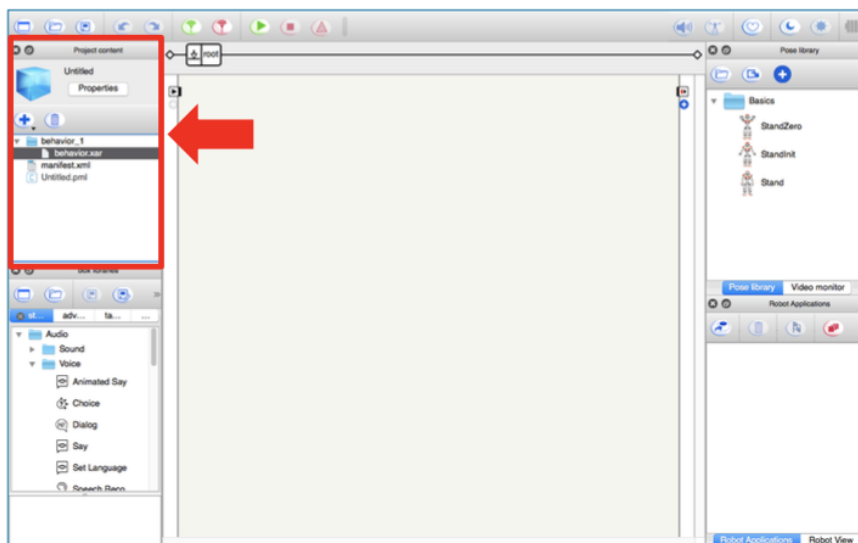
## 1. Présentation du logiciel Choregraphe

La figure ci-dessous présente l'interface du logiciel Choregraphe.

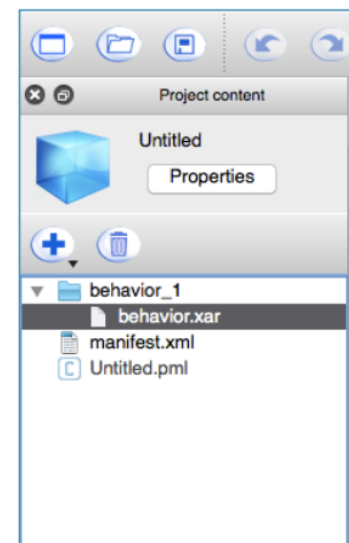


**Figure 1.** Interface du logiciel Choregraphe

Dans la colonne de gauche, on trouve la fenêtre Project Content (Figure 2) et la fenêtre Box Libraries (Figure 3) qui contient toutes les boîtes (actions) à ajouter dans l'espace de codage.



**Figure 2.** Fenêtre Project Content



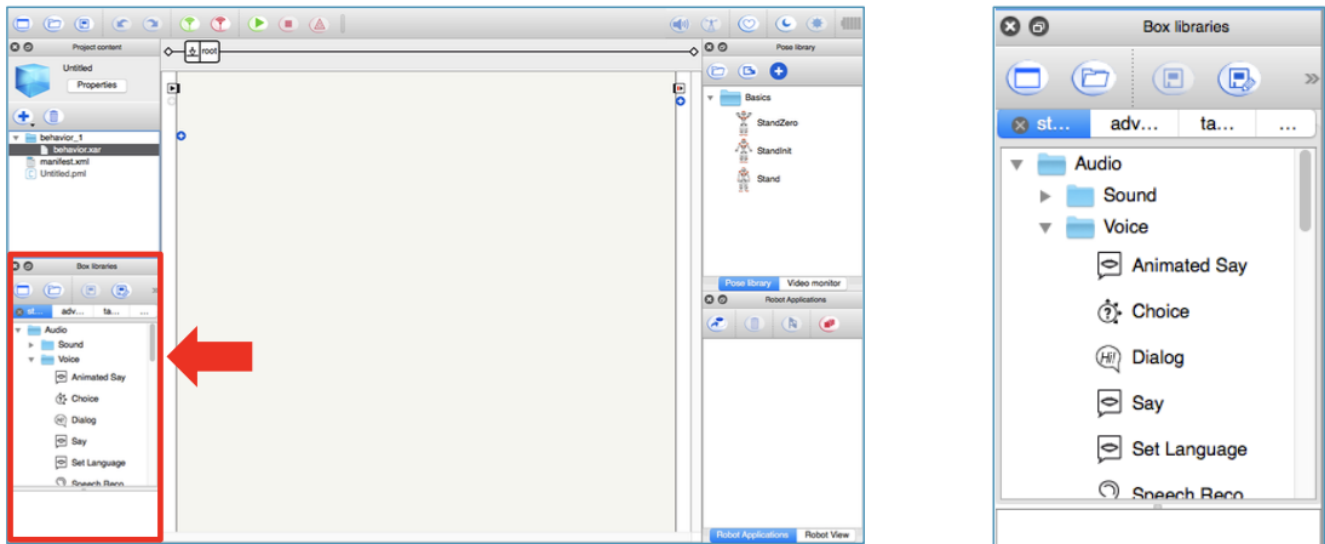


Figure 3. Fenêtre Box libraries

La fenêtre centrale est l'espace de travail où on peut déposer et relier les différentes boîtes (Figure 4).

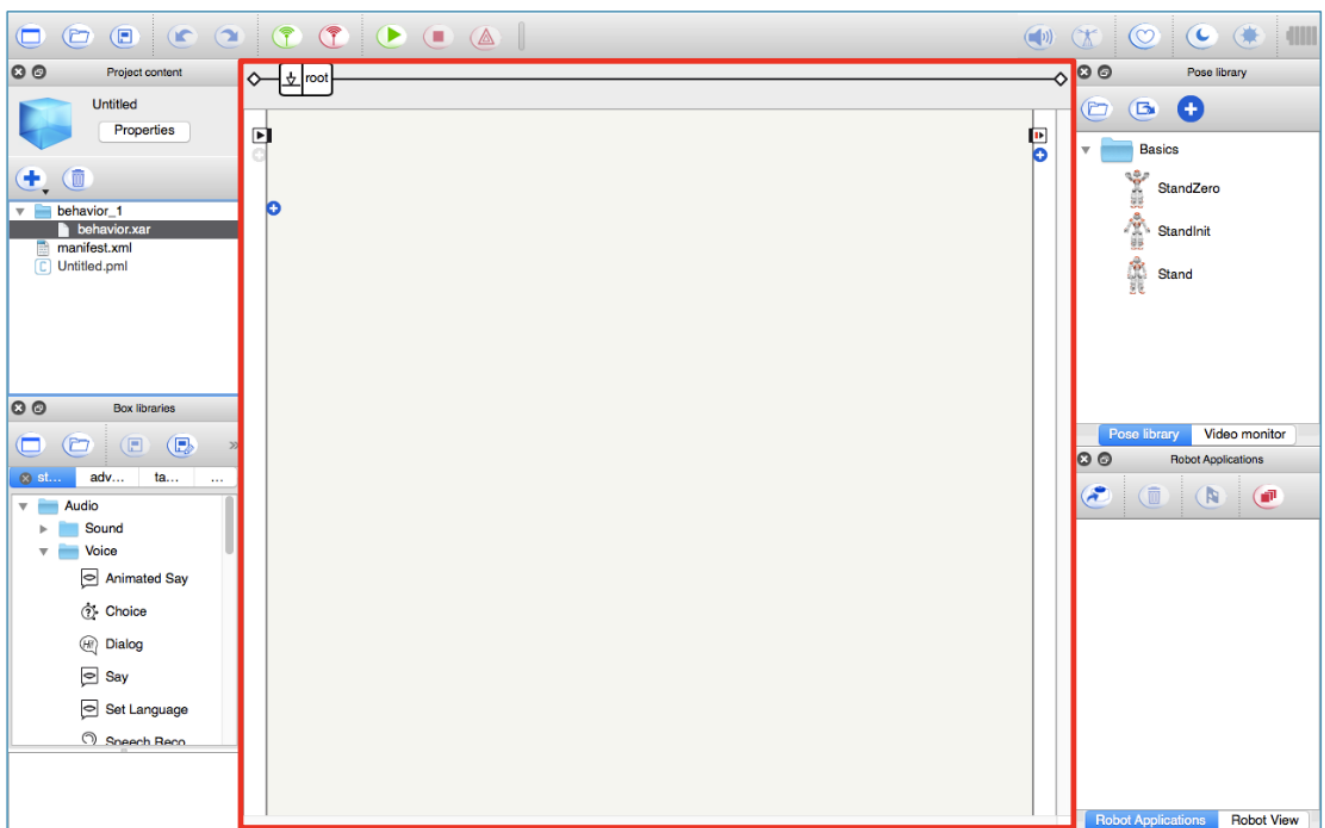


Figure 4. Fenêtre Espace de travail

Les boîtes peuvent être sélectionnées dans les colonnes de gauche et de droite, afin de les déposer dans l'espace de travail (Figures 5 et 6).

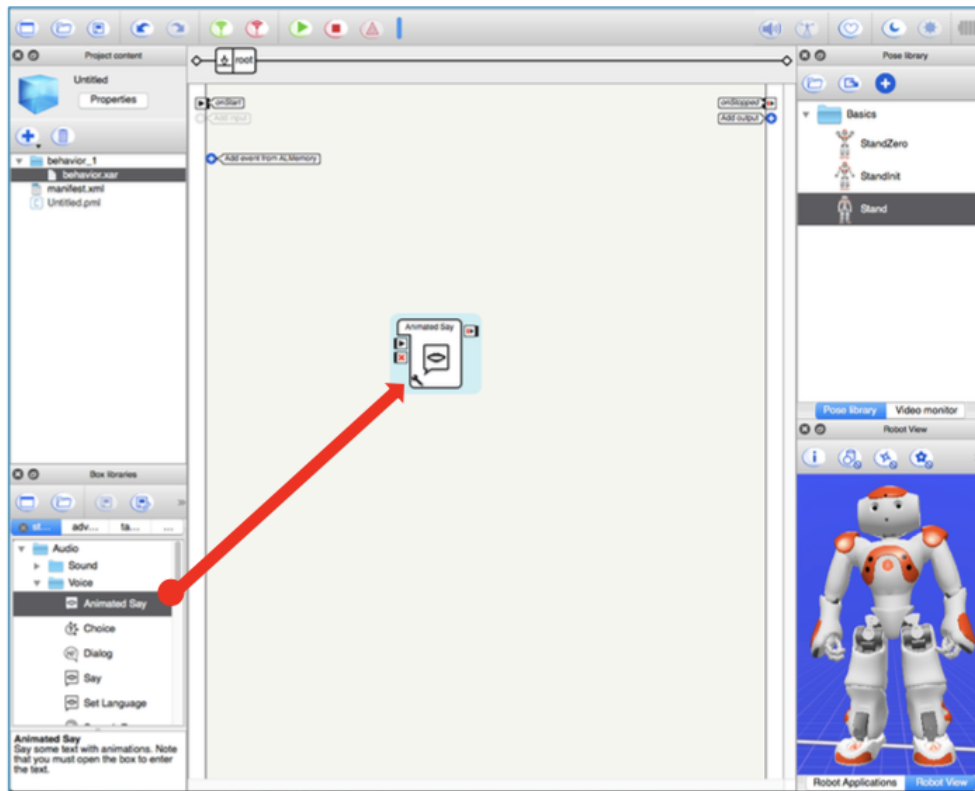


Figure 5. Sélection des boîtes à partir de Box libraries

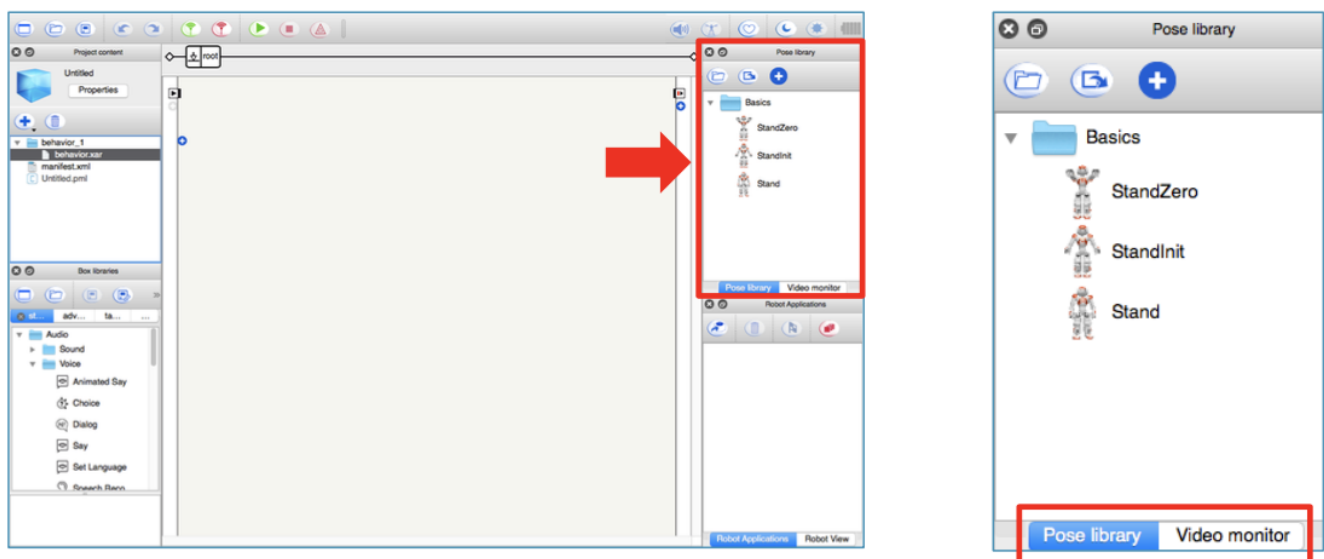


Figure 6. Sélection des boîtes à partir de Pose library

- En cliquant sur l'onglet Video monitor (Figure 6), on peut voir le retour vidéo de la caméra de NAO.
- En cliquant sur l'onglet Robot view (Figure 7), on peut voir le robot exécuter les instructions programmées dans l'espace de travail (en mode simulé et (ou) réel).

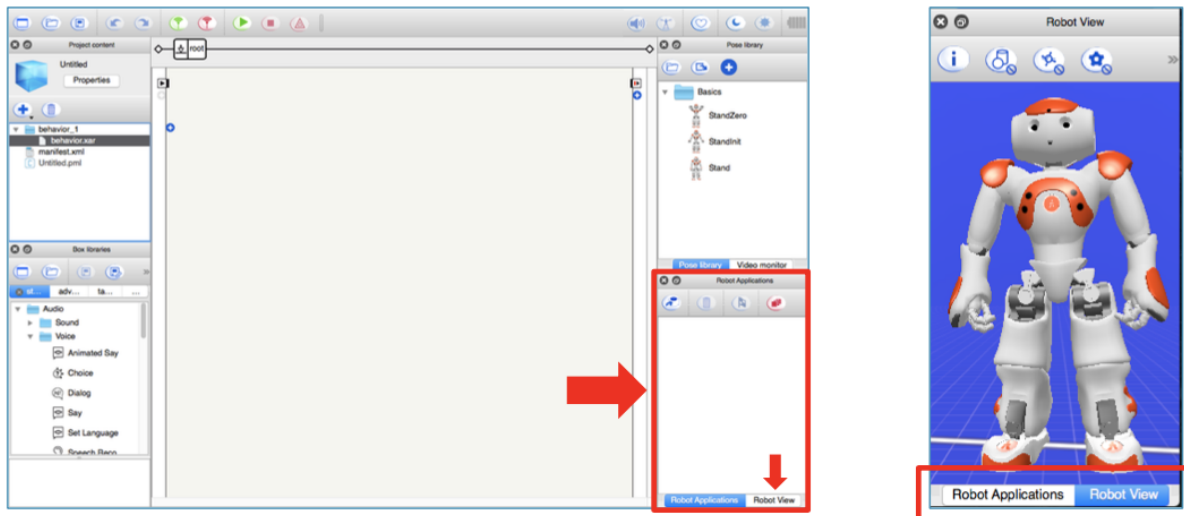


Figure 7. Fenêtre Robot view

Les différents nouveaux éléments de la barre d'outils sont donnés dans la figure 8.

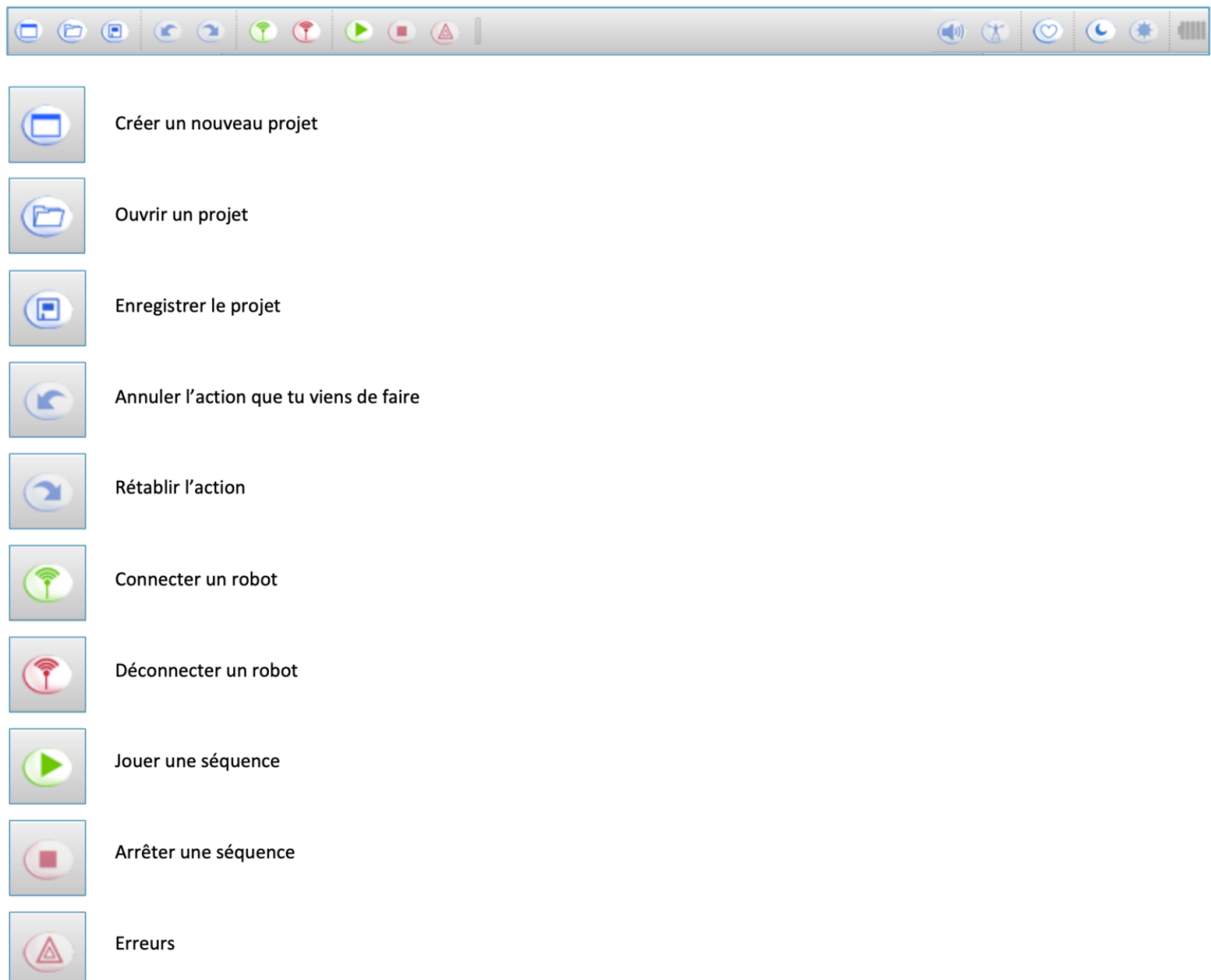



Figure 8. La barre d'outils

## 2. Exemple : Faire marcher NAO en utilisant la boîte Move To

Avant de commencer à programmer, il est recommandé de connecter NAO en local (virtual robot) afin de visualiser les mouvements et les déplacements du robot dès l'exécution de votre programme. Pour cela, cliquer sur l'icône  ou **Connection** puis sur **Connect to virtual robot** (Voir Figure 9).

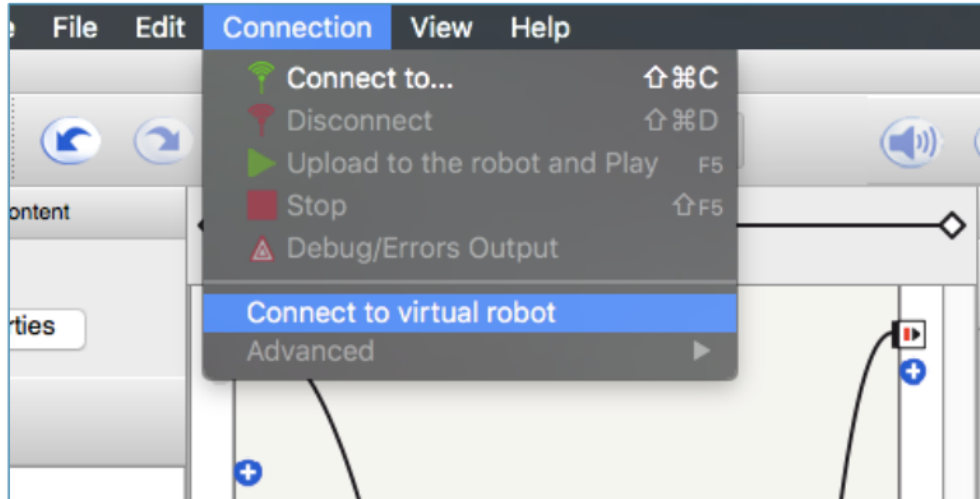


Figure 9. Connexion en local (robot virtuel)

Pour connecter le vrai robot NAO, il faut le connecter au réseau internet de la salle ou à l'ordinateur avec un câble ethernet.

### Réalisation du programme

Dans la fenêtre Box libraries, sélectionner le dossier Motions, puis Move To et déplacer la boîte dans l'espace de travail.



Figure 10. Sélection de la boîte Move To

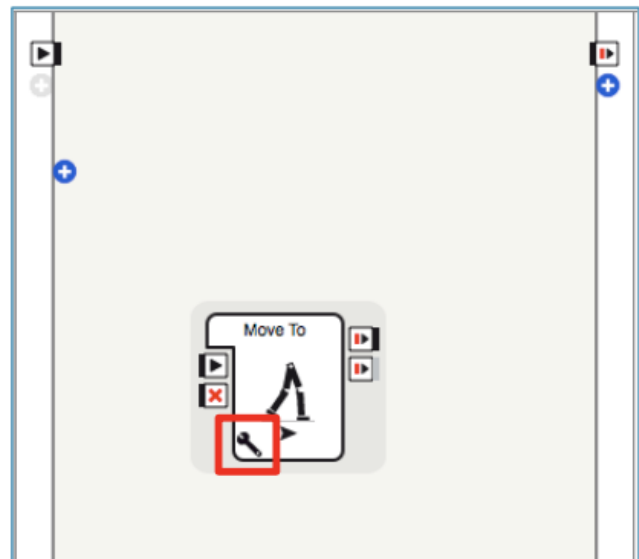

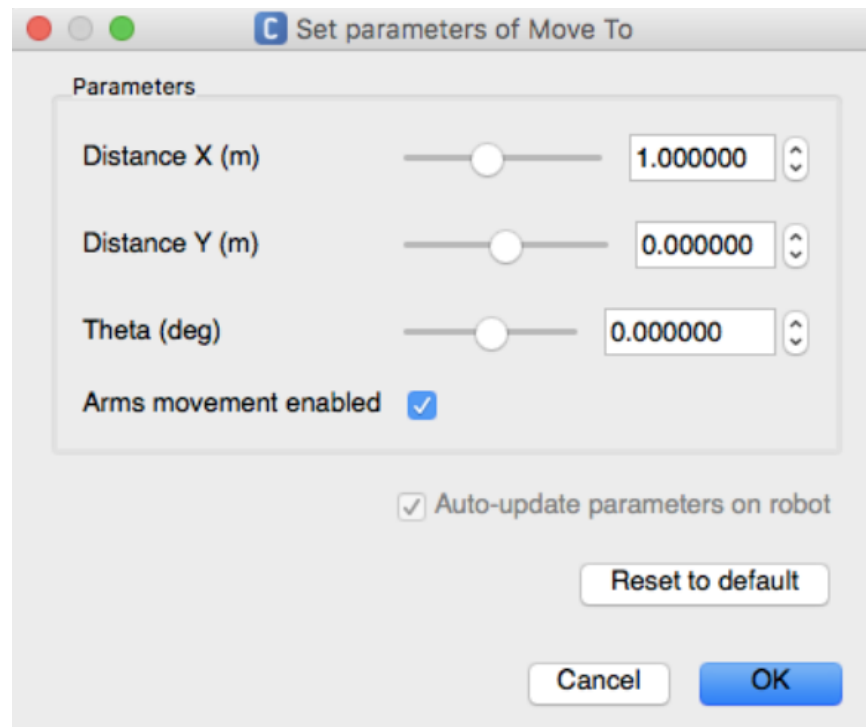
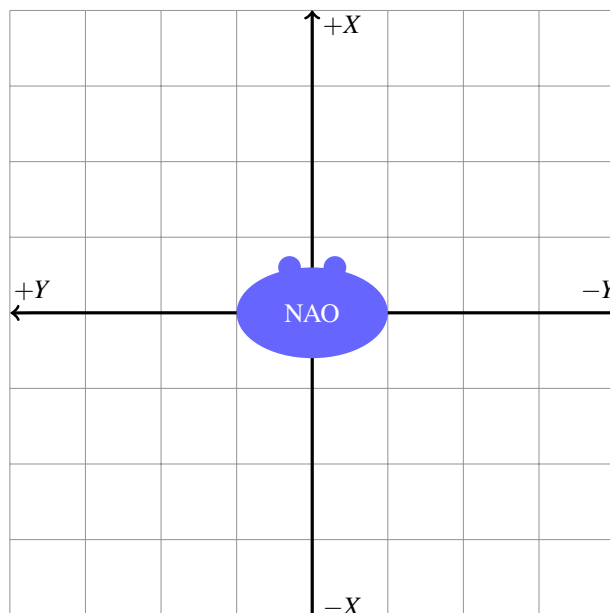


Figure 11. Placement de la boîte dans l'espace de travail

Cliquer sur  pour ouvrir la fenêtre qui permet de modifier les paramètres de l'action.

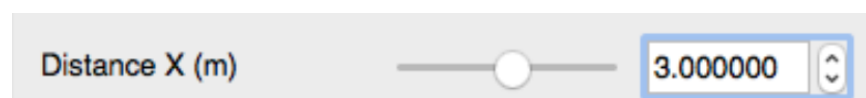


**Figure 12.** Réglage des paramètres de la boîte Move To



**Figure 13.** Correspondance des déplacements

Si on veut déplacer NAO à la position de la figure 16, on doit saisir les valeurs ci-dessous :



**Figure 14.** Réglage du déplacement longitudinal (selon l'axe des X)



Figure 15. Réglage du déplacement transversal (selon l'axe des Y)

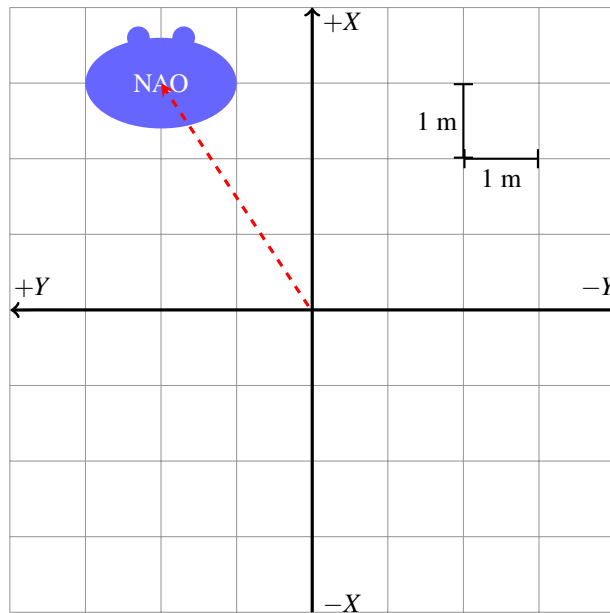
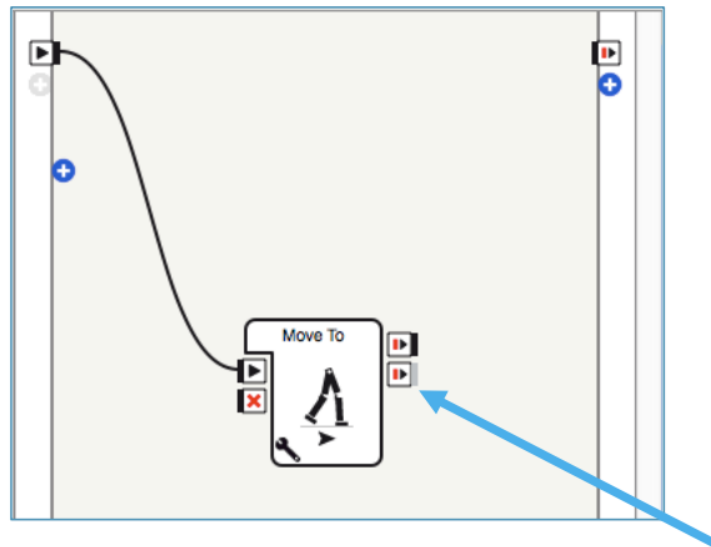


Figure 16. Position finale du robot NAO



Dans le cas où on veut utiliser plusieurs boîtes Move To, on doit utiliser le output gris, et non le noir.



### 3. Exercices



#### Remarque

- Les codes doivent être testés sur le robot virtuel
- Les codes doivent être validés par l'enseignant
- Ne pas utiliser le vrai robot NAO sans l'aval de l'enseignant

#### 3.1 Exercice 1

Réaliser un programme pour que NAO se déplace de la position initiale à la position finale (voir Figure 17). Noter bien l'orientation du buste de NAO au départ et à l'arrivée.

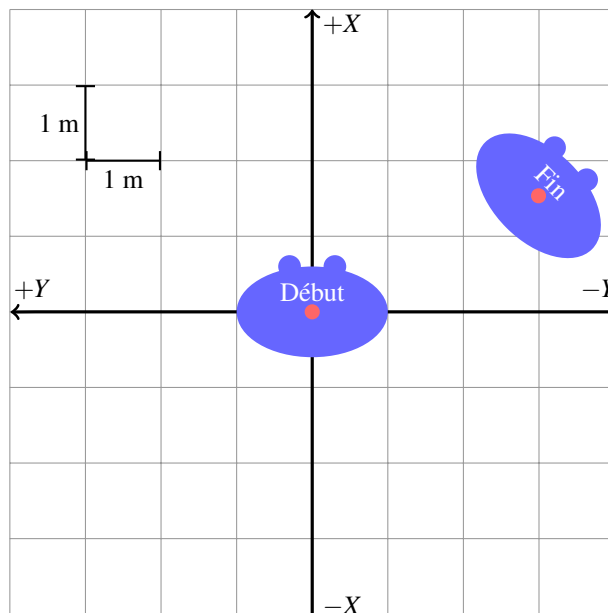


Figure 17. Position initiale et finale du robot NAO

#### 3.2 Exercice 2

Réaliser un programme qui permet à NAO :

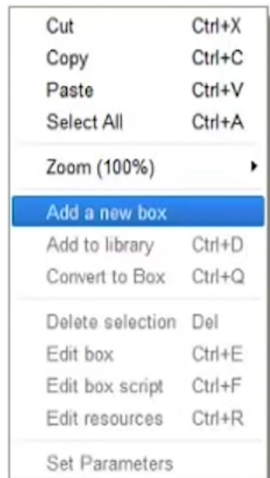
- d'avancer de 50 cm et ces yeux doivent être verts,
- de se déplacer à droite de 30 cm

À chaque fois qu'il arrive à sa position finale, il doit prononcer la phrase **je suis arrivé**.

#### 3.3 Exercice 3

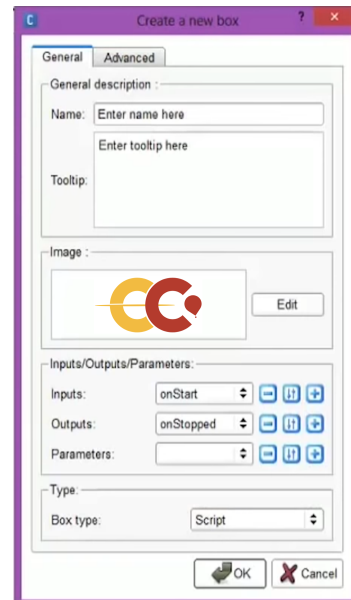
Objectif : Créer et coder en Python une nouvelle boîte pour réaliser une tâche.

1. Avec un clic droit dans la fenêtre centrale (espace de travail) sélectionner **Add a new box**

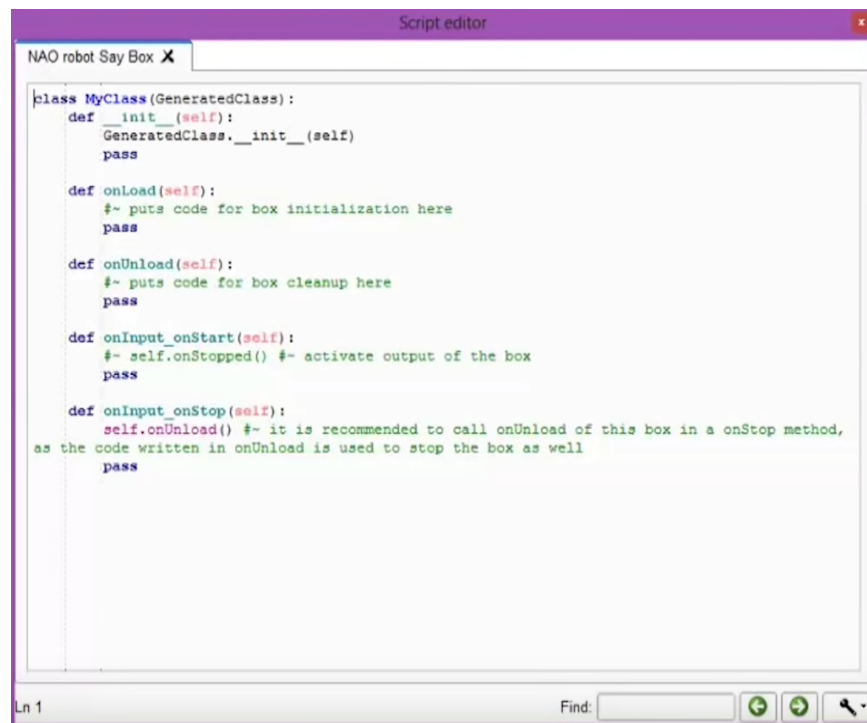


2. Renseigner les différents items de la fenêtre **Create a New Box**

3. Cliquer sur **OK**



Une fenêtre d'édition de script apparaît si on double clic sur la boîte créée.



**Figure 18.** Fenêtre d'édition de script

4. Modifier (compléter) le code de la fonction (méthode) `onInput_onStart (self)`

Exemple 1 : Cette partie du code permet de donner l'instruction à NAO de prononcer la phrase **Bonjour tout le monde.**

```
def onInput_onStart(self):
    ttsProxy = ALProxy("ALTextToSpeech")
    ttsProxy.say("Bonjour tout le monde")
```

Exemple 2 : Cette partie du code permet de donner l’instruction à NAO d’avancer de 30 cm.

```
def onInput_onStart(self):
    motionProxy = ALProxy("ALMotion")
    motionProxy.walkTo(0.3, 0.0, 0.0)
    self.onStopped()
```

4. Réaliser un programme qui permet à NAO de faire demi tour et de prononcer à la fin la phrase *Cette vue est magnifique.*, en utilisant uniquement les boîtes de script Python.

## 4. TP robot NAO

### 4.1 TP1 :

Réaliser un programme qui permet à NAO de réaliser les tâches suivantes de manière chronologique :

1. Prononcer une phrase pour souhaiter la bienvenue aux gens
2. Avancer de 30 cm
3. Saluer le public avec sa main tout en prononçant la phrase "merci de votre visite"

### 4.2 TP2 :

1. Réaliser un programme qui permet à NAO de :
  - Suivre une trajectoire ou un itinéraire
  - Effectuer une tâche dès qu’il entend le mot *action* (ou tout autre mot).
  - Reconnaître un objet et effectuer une tâche dès qu’il aperçoit cet objet

} Par commande vocale et tactile