

Tutoriel SDK 4DV-Editor

Créer et éditer son propre système



Exemple d'étude : création d'une voiture

Avant d'aller plus loin...

➤ Objectif

Créer et éditer son propre système, par l'exemple de création d'une voiture.

➤ Prérequis

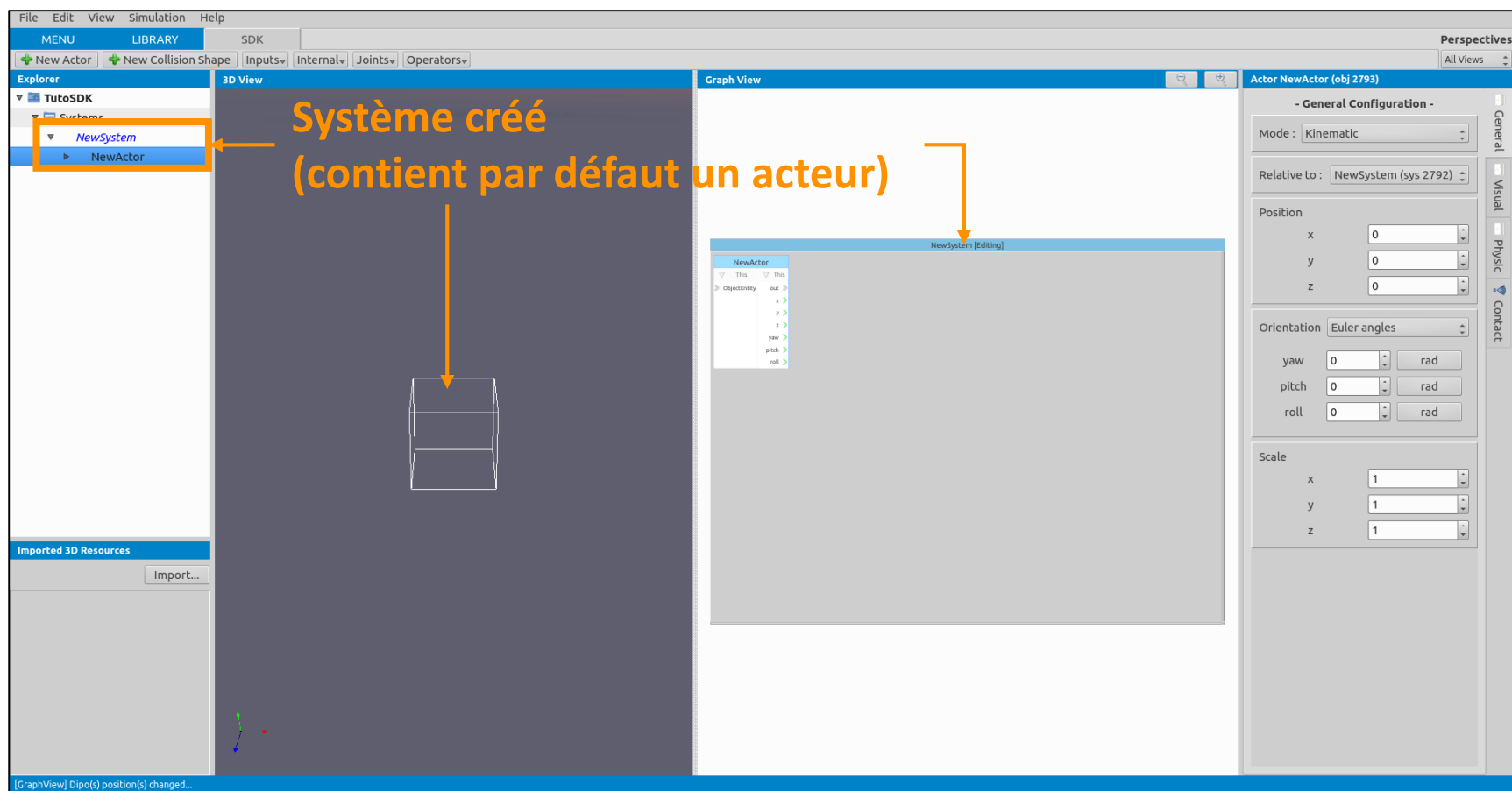
- ✓ Utilisation simple de 4DV-Editor (cf. documentation utilisateur)
- ✓ Disposer de ressources 3D : fichiers Ogre3D (.mesh)
 - Une ressource pour le châssis de la voiture
 - Une ressource pour une roue

Etape 1 :

Créer un système vide

Créer un système vide

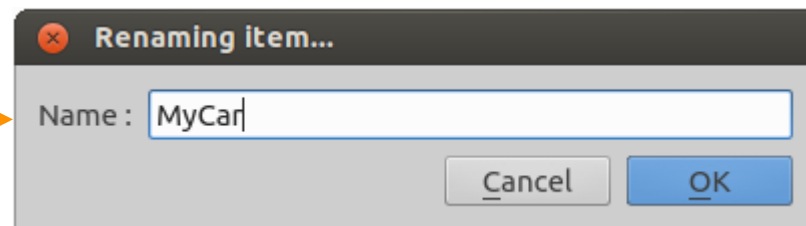
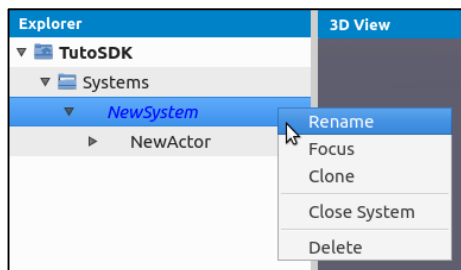
➤ Onglet LIBRARY > New System



Quelques manipulations

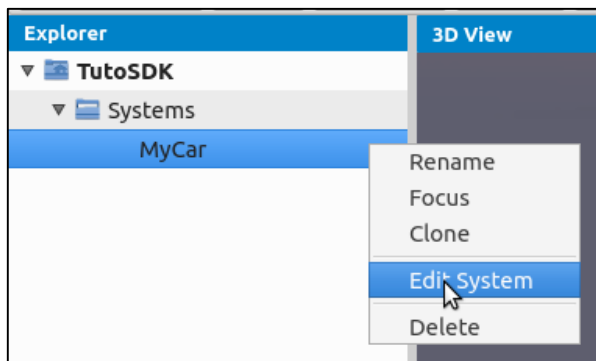
➤ Renommer le système

✓ Clic droit sur le système dans la partie *Explorer* > **Rename**

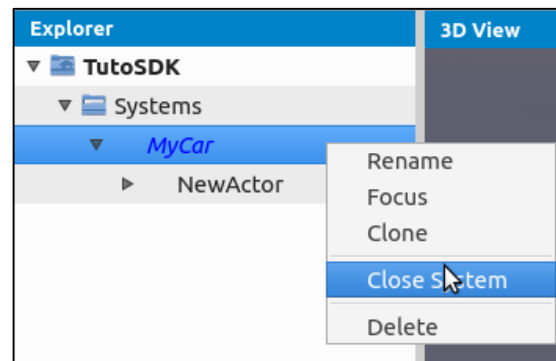


➤ Fermer/ouvrir le système

✓ Clic droit sur le système dans la partie *Explorer* > **Close system** ou **Edit system**



Ouvrir (éditer) le système



Fermer le système

Etape 2 : Créer/éditer des acteurs

Créer un acteur

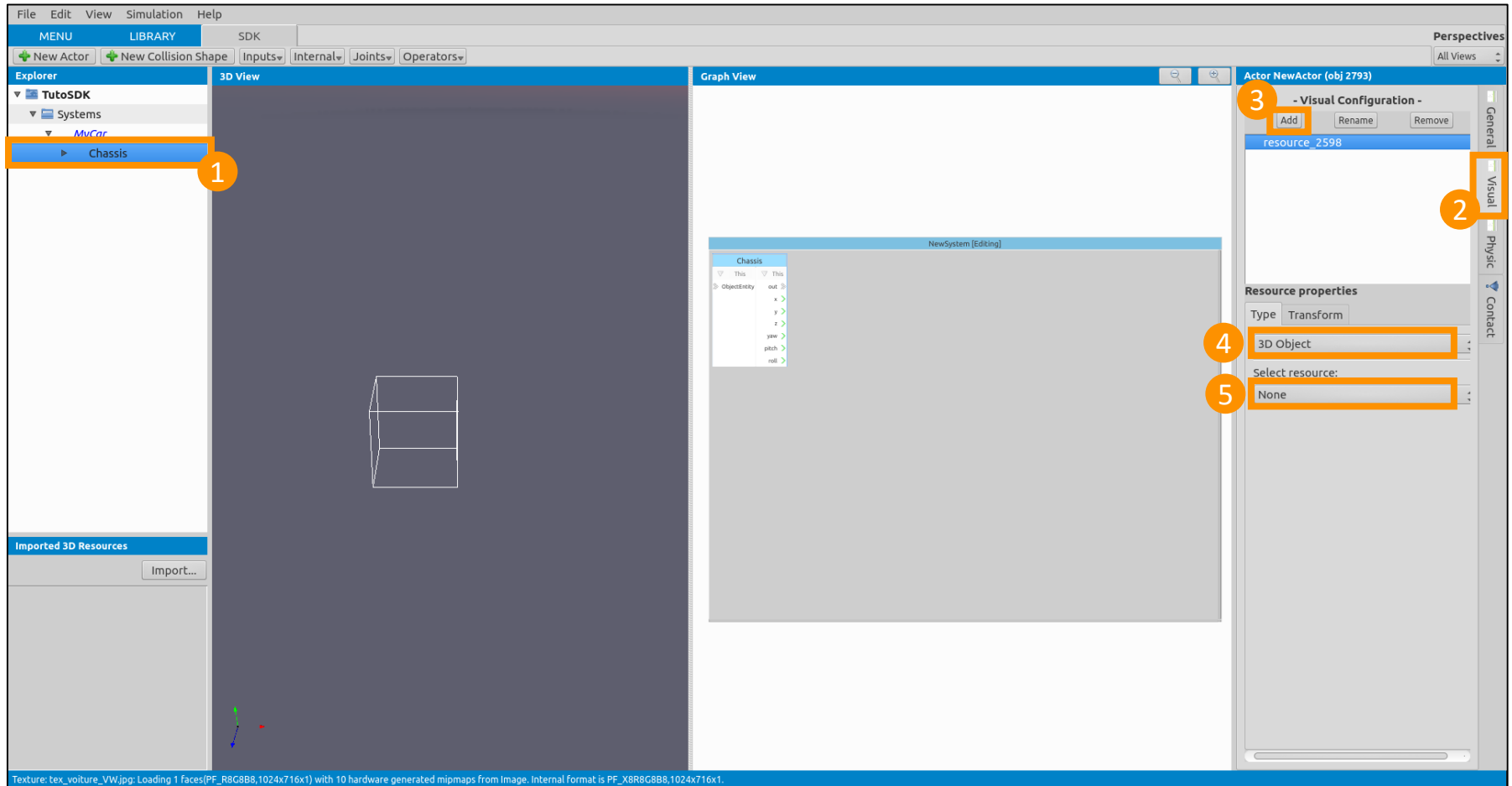
- **Le premier acteur existe déjà**
 - ✓ Il a été créé automatiquement avec le système
- **Pour créer les acteurs suivants**
 - ✓ Onglet SDK > **New Actor**
- **Renommer un acteur**
 - ✓ Procéder comme pour le système, en cliquant sur l'acteur dans la partie *Explorer*



Penser à renommer systématiquement les acteurs pour les reconnaître facilement par la suite.

Création de l'acteur « Châssis »

➤ Ajout d'un élément visuel



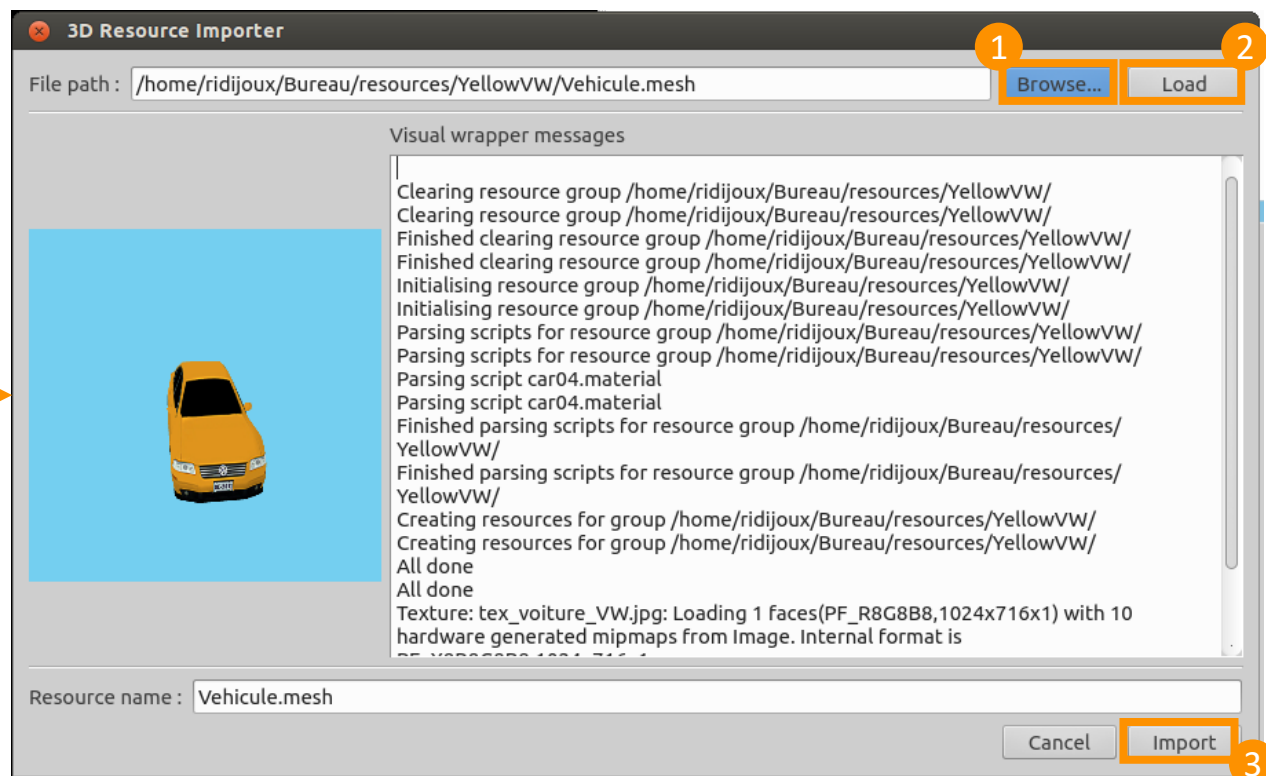
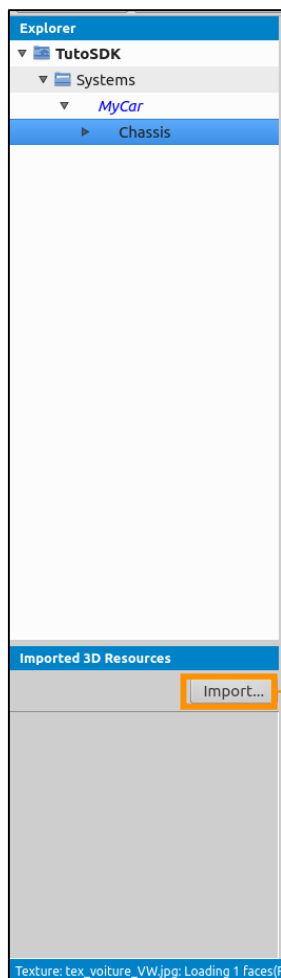
➤ Ajout d'un élément visuel

- 1 Sélectionner l'acteur
- 2 Onglet **Visual**
- 3 Ajouter une ressource visuelle
- 4 Sélectionner le type de la ressource visuelle
 - ✓ Simple : box, sphere etc...
 - ✓ Ressource 3D importée : « 3D Object »
- 5 Sélectionner la ressource importée (cas « 3D Object »)



La première fois, il est nécessaire d'importer une ressource avant de pouvoir l'utiliser.

Importer une ressource 3D



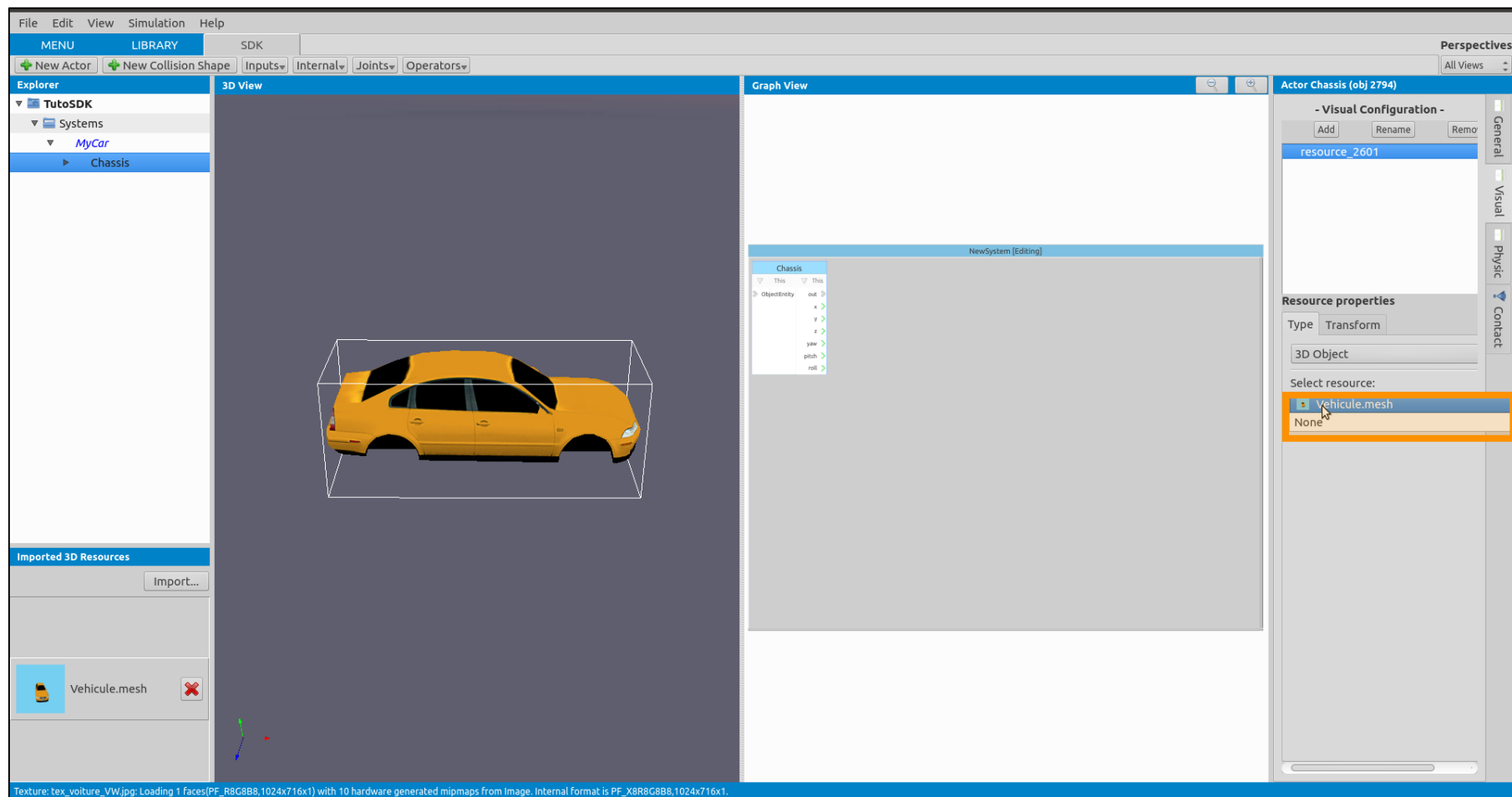
- 1 Spécifier le fichier de ressource 3D (.mesh ou .scene)
- 2 Charger la ressource
- 3 Finaliser l'importation



Cette opération doit être faite une fois par ressource. Par la suite, elle est disponible pour tous les projets.

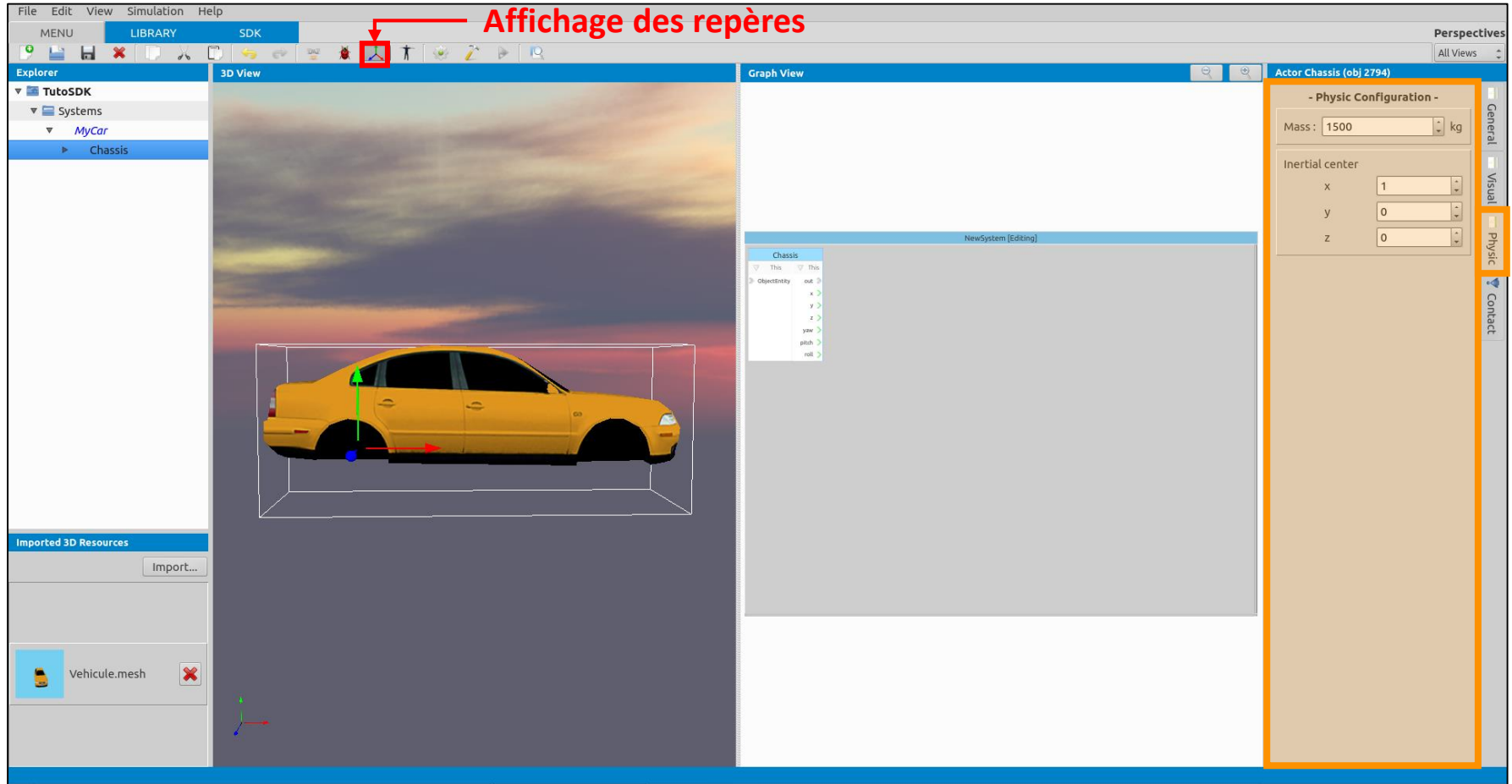
Création de l'acteur « Châssis »


➤ Visuel du châssis



Création de l'acteur « Châssis »

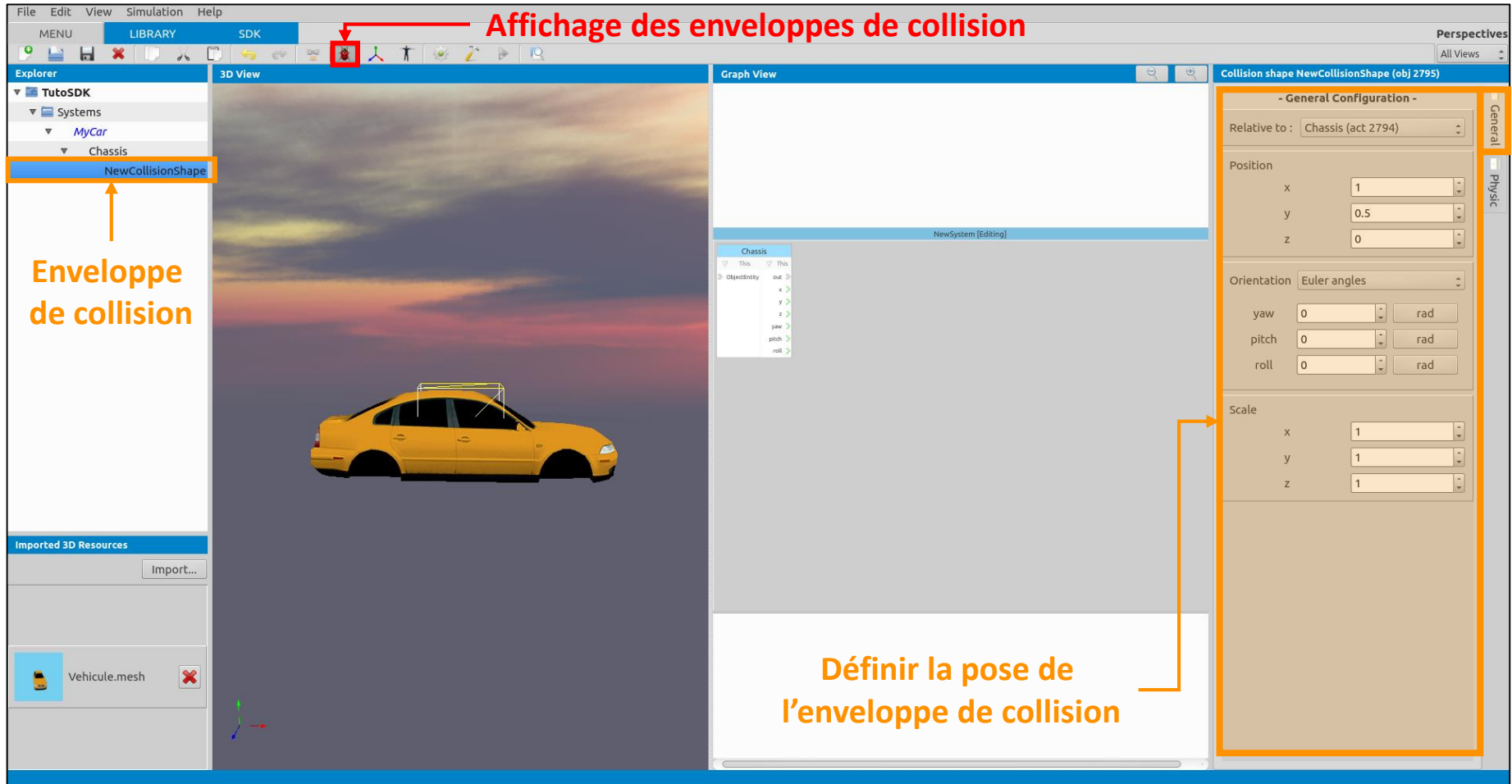
➤ Définition de la physique



 Le centre d'inertie est défini par rapport au repère de l'acteur.

Création de l'acteur « Châssis »

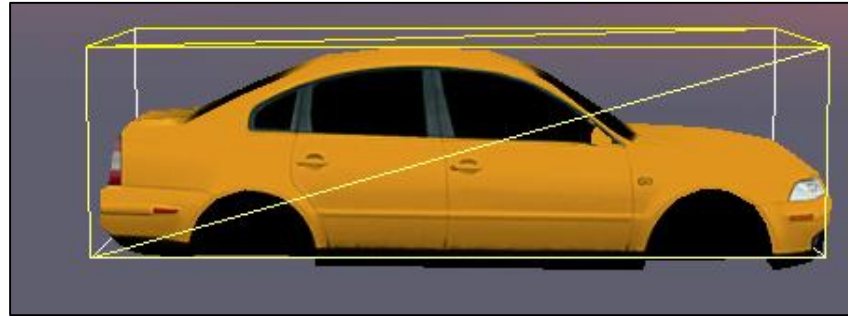
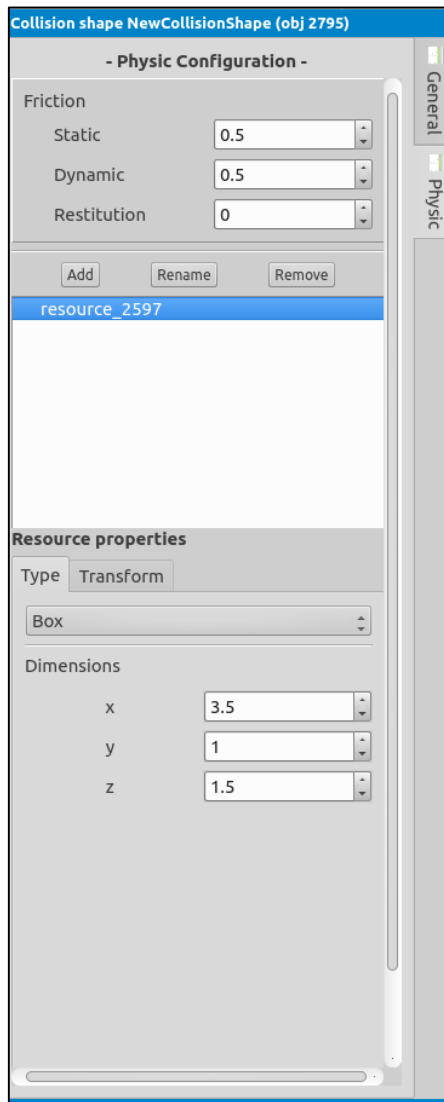
➤ Définition des enveloppes de collision (« Collision shapes »)



Une enveloppe de collisions est automatiquement créée avec l'acteur.
Pour en créer d'autres : Onglet SDK > **New CollisionShape**

Création de l'acteur « Châssis »

➤ Définition des enveloppes de collision



➤ Onglet *Physic* de l'enveloppe de collision

- ✓ Définir la forme et la taille de l'enveloppe de collision

➤ Formes

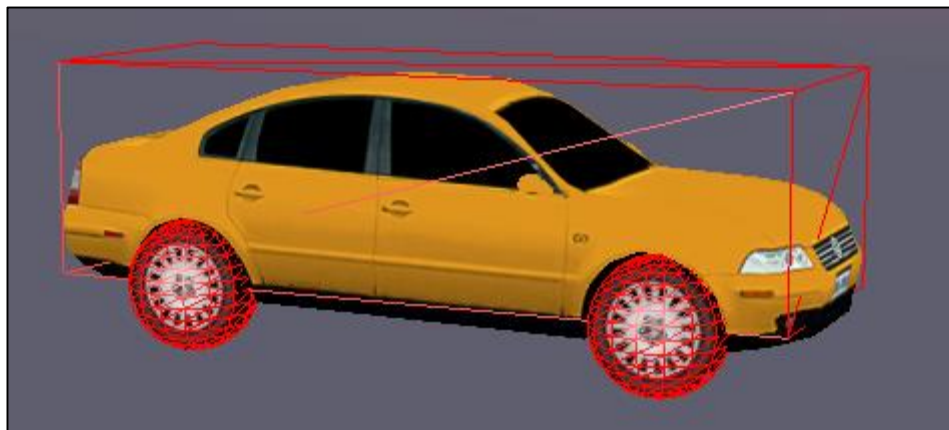
- ✓ Simple : Box, sphere, etc...
- ✓ A partir d'une ressource 3D importée : « 3D Object »



Associer plusieurs formes simples pour affiner la collision

Création des acteurs « Roues »

- Reprendre chacune des étapes précédentes
- Collisions
 - ✓ Pour ce tutoriel, on utilisera des sphères
 - ✓ Redéfinir éventuellement les paramètres « Friction » de l'onglet « Physic » pour régler les coefficients de glissement entre la roue et le sol



Résultat avec vue des « Collision shapes »



Le copier/coller peut être utilisé pour les acteurs d'un système édité

Etape 3 :

Définir des liaisons entre acteurs

- **Créer une liaison entre deux acteurs d'un système**
 - ✓ Onglet SDK > **Joints**

- **Définition de liaisons motrices**
 - ✓ Définir liaisons pivots (« Motor Joint ») entre châssis et roues
 - ✓ Axe de rotation horizontal


- **Définition de liaisons directionnelles**
 - ✓ Nécessité d'une seconde liaison pivot
 - ✓ Axe de rotation vertical
 - ✓ Seulement pour les roues avant

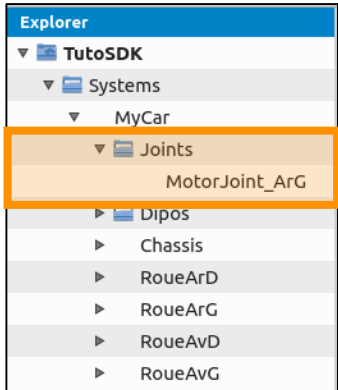
Liaisons des roues arrières

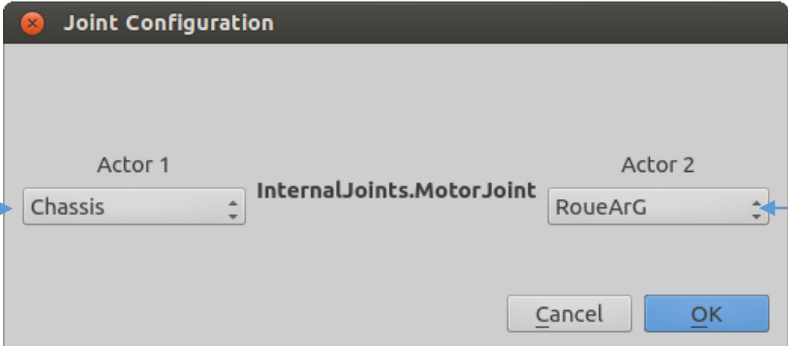
➤ Exemple : Créer la liaison motrice pour la roue arrière gauche

✓ Onglet SDK > **Joints** > **MotorJoint**


Choisir le premier acteur







Choisir le second acteur



1 – La liaison apparaît au niveau de l'explorer dans le dossier « Joints » du système.

2- La définition des acteurs de la liaison n'est pas modifiable : il faut alors créer une nouvelle liaison.



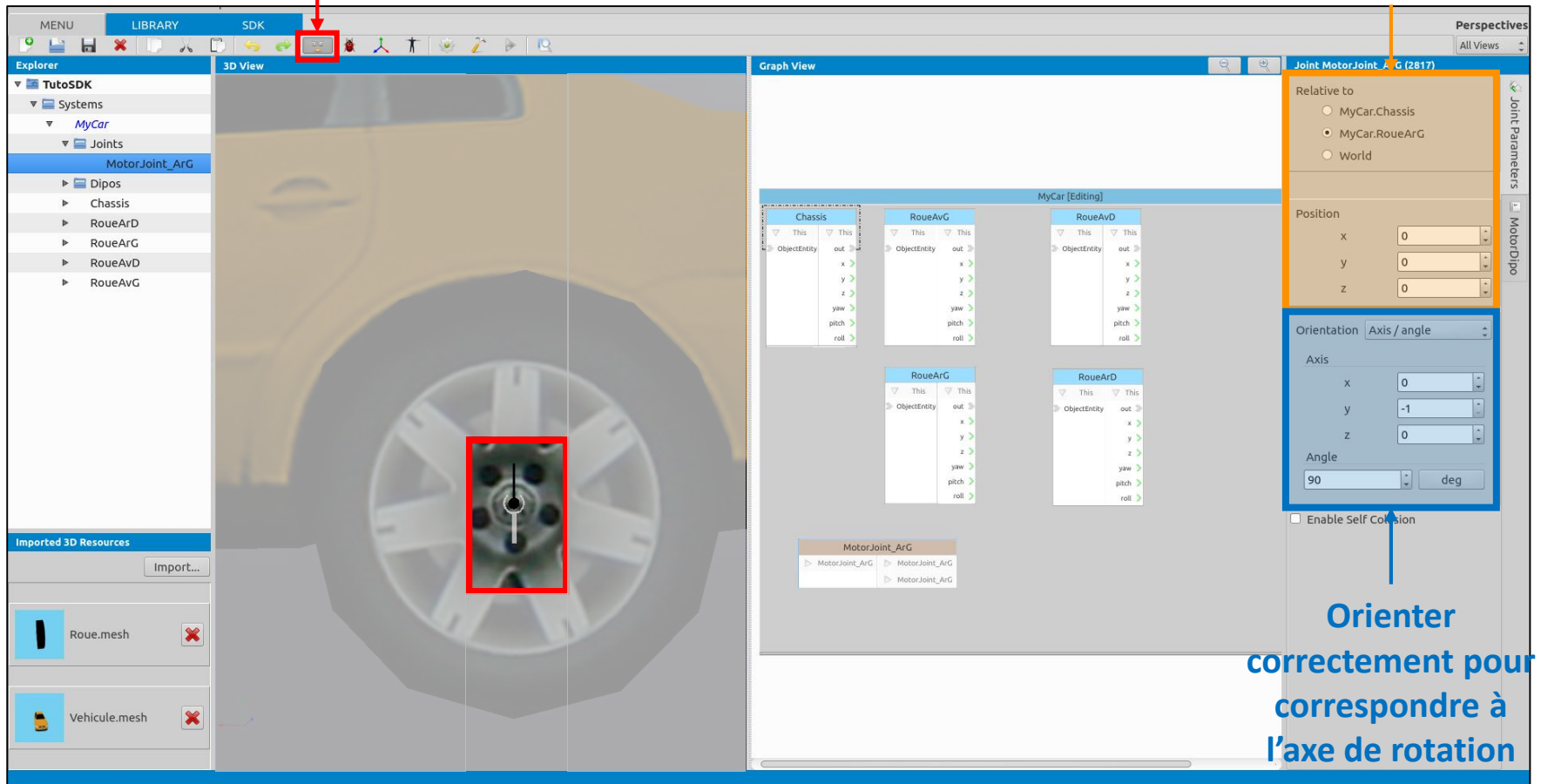
Comme pour le système et les acteurs, penser à renommer la liaison.
Par exemple ici : *MotorJoint_ArG*.

Liaisons des roues arrières

➤ Configurer la liaison

Positionner au centre
de la roue

Affichage des joints



The screenshot shows the 4DVirtualiz SDK 4DV-Editor interface. The left sidebar contains the Explorer panel with a tree view showing the project structure: **TutoSDK** > **Systems** > **MyCar** > **Joints** > **MotorJoint_ArG**. Below this is the Imported 3D Resources panel with **Roue.mesh** and **Vehicule.mesh**. The main 3D View shows a car model with a red box highlighting the rear wheel hub. The right sidebar contains the Graph View and the Joint MotorJoint_ArG (2817) properties panel. The properties panel has a 'Relative to' section with radio buttons for **MyCar.Chassis**, **MyCar.RoueArG** (selected), and **World**. Below this is the 'Position' section with input fields for x (0), y (0), and z (0). The 'Orientation' section has a dropdown set to 'Axis / angle', with 'Axis' set to 'y' (value -1) and 'Angle' set to '90' (unit 'deg'). An 'Enable Self Collision' checkbox is at the bottom.

Orienter
correctement pour
correspondre à
l'axe de rotation

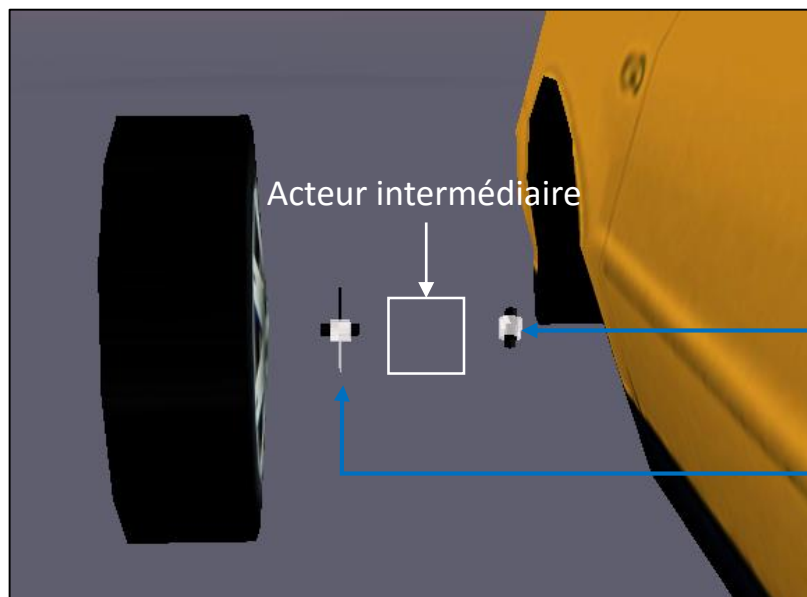
Liaisons des roues avant

➤ Deux types de mouvements : moteur + direction

- ✓ Aucune liaison ne donne cette combinaison
- ✓ Nécessité de deux liaisons en série



Pour que la simulation fonctionne, il faut ajouter un acteur intermédiaire vide (supprimer l'enveloppe de collision).



MotorJoint (Direction)
Axe rotation vertical

MotorJoint
Axe rotation horizontal

Vue éclatée des joints au niveau de la roue avant gauche

Etape 4 :

Actionneur et paramétrage général du véhicule

Actionneur



➤ Mise à disposition d'entrées de commandes

- ✓ Vitesse linéaire ou angulaire
- ✓ Direction des roues avant ou arrières

➤ Choix du modèle cinématique du véhicule

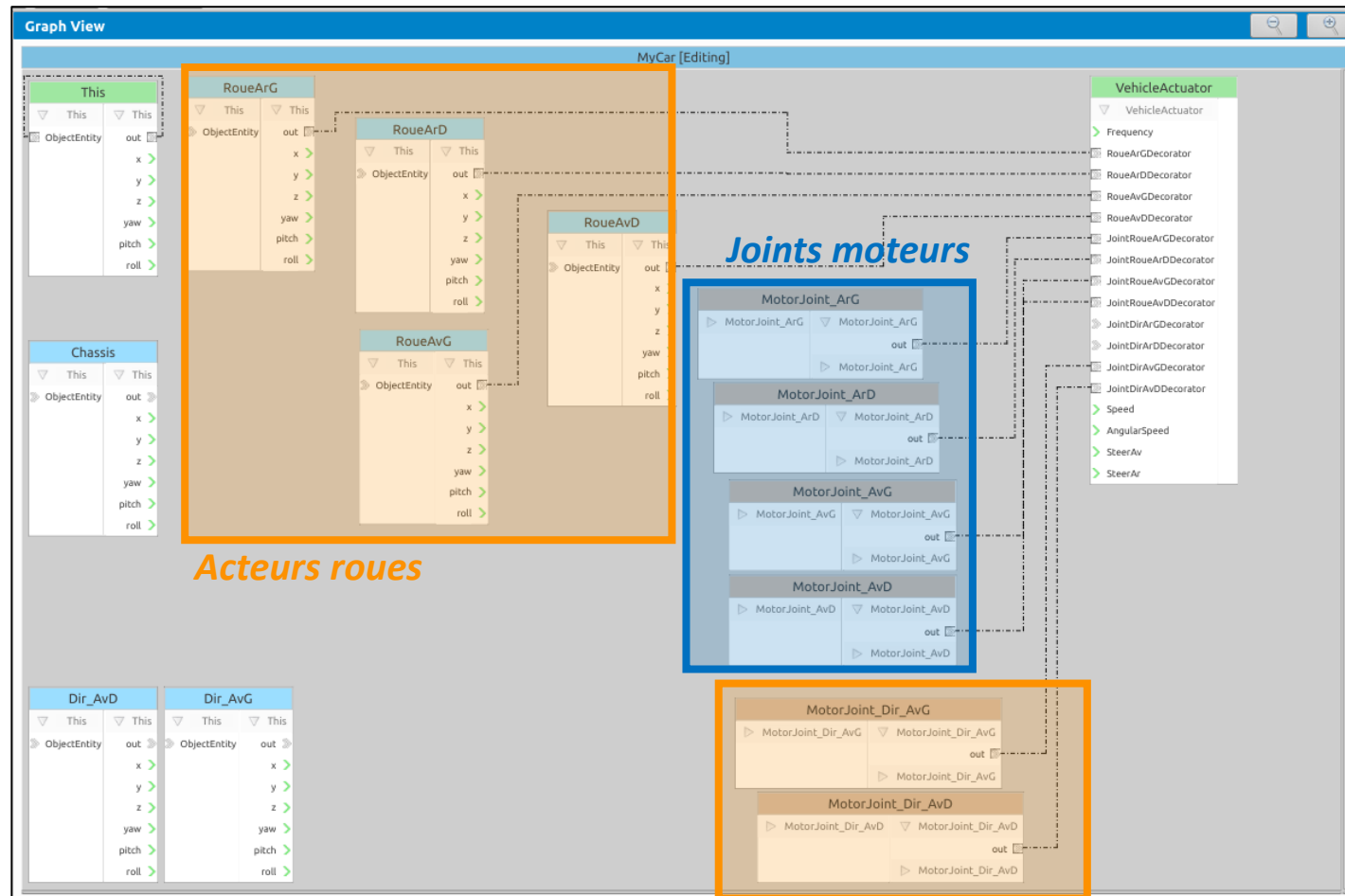
- ✓ Ackermann
- ✓ Mode « Tank »

➤ Ajouter un actionneur au véhicule

- ✓ Onglet SDK > Internal > Actuator > VehicleActuator

Actionneur

➤ Connexions dans la partie *Graph View*

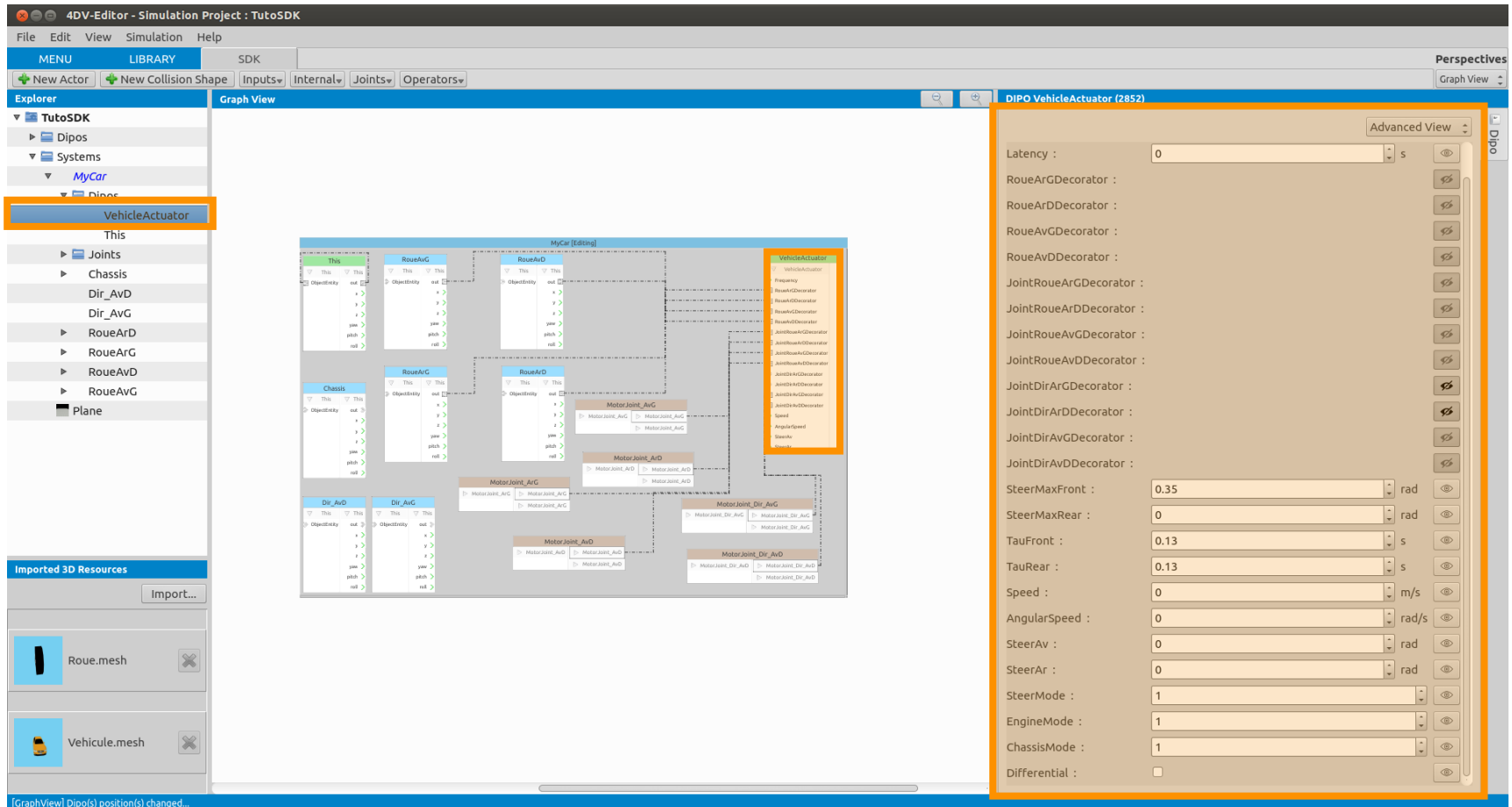


Joints directionnels

Configurer l'actionneur

➤ Définir certains réglages

- ✓ Par exemple : l'angle de braquage des roues arrières (SteerMaxRear) est nul car elles ne sont pas directionnelles



The screenshot displays the 4D-Editor SDK interface for a simulation project named 'TutoSDK'. The main window shows a graph view of the 'MyCar' system, which includes components like 'RoueArG', 'RoueArD', 'RoueAvG', 'RoueAvD', 'Chassis', 'Dir_AvD', 'Dir_AvG', 'MotorJoint_AvG', 'MotorJoint_AvD', 'MotorJoint_Dr_AvG', and 'MotorJoint_Dr_AvD'. The 'VehicleActuator' component is highlighted in the Explorer panel on the left. The 'Advanced View' of the 'VehicleActuator' is shown on the right, listing various parameters and their values:

- Latency : 0 s
- RoueArGDecorator : [icon]
- RoueArDDecorator : [icon]
- RoueAvGDecorator : [icon]
- RoueAvDDecorator : [icon]
- JointRoueArGDecorator : [icon]
- JointRoueArDDecorator : [icon]
- JointRoueAvGDecorator : [icon]
- JointRoueAvDDecorator : [icon]
- JointDirArGDecorator : [icon]
- JointDirArDDecorator : [icon]
- JointDirAvGDecorator : [icon]
- JointDirAvDDecorator : [icon]
- SteerMaxFront : 0.35 rad
- SteerMaxRear : 0 rad
- TauFront : 0.13 s
- TauRear : 0.13 s
- Speed : 0 m/s
- AngularSpeed : 0 rad/s
- SteerAv : 0 rad
- SteerAr : 0 rad
- EngineMode : 1
- ChassisMode : 1
- Differential : [checkbox]

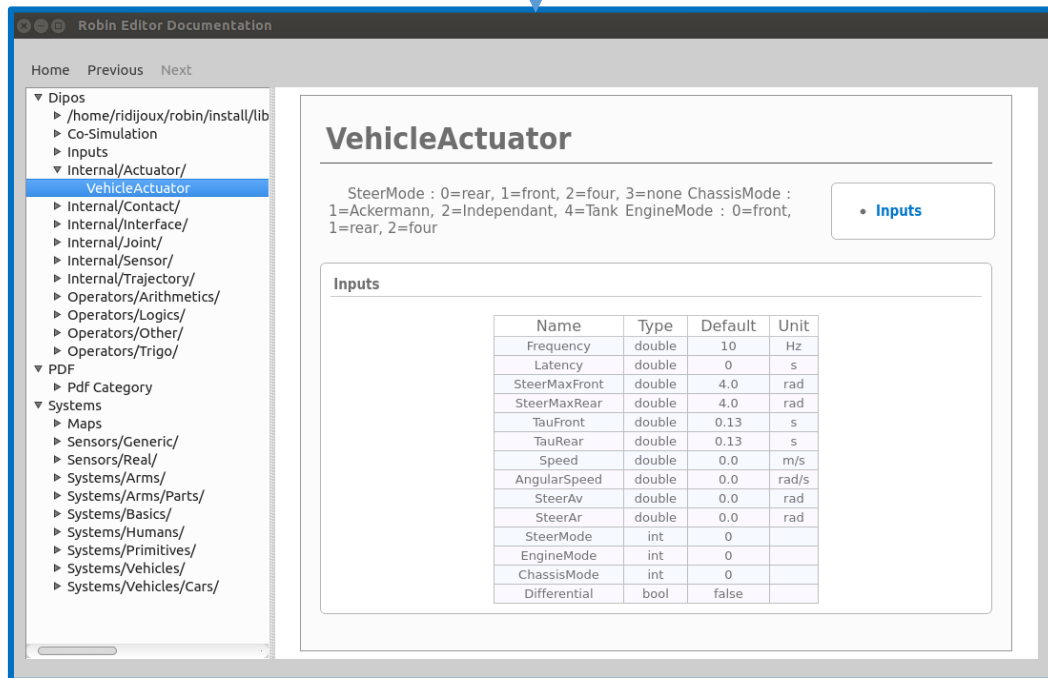
Configurer l'actionneur

➤ Configurations avancées

1 Choix : **Advanced View**

2 Paramètres cinématiques

- ✓ Se référer à la **documentation**
- ✓ Help > **Documentation**



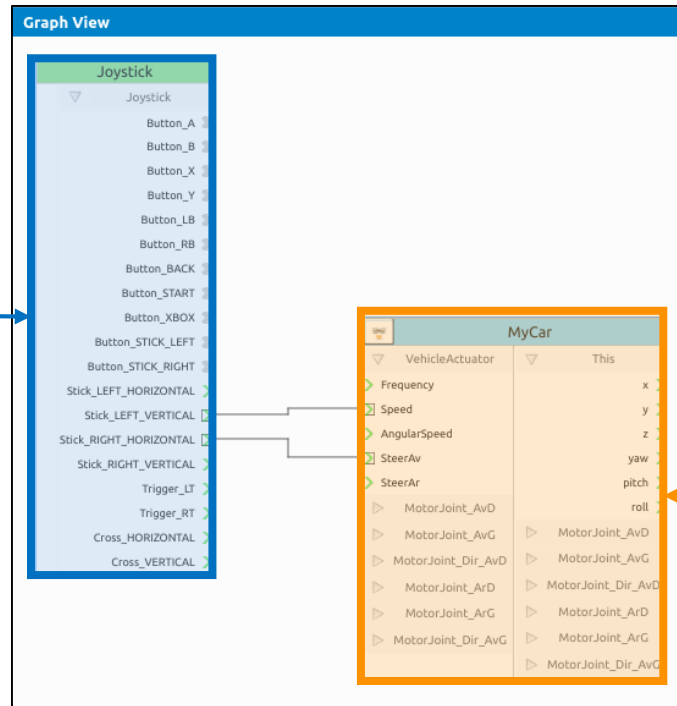
Utiliser le système dans le projet :

Exemple : commander le véhicule avec un Joystick

Simulation avec commande Joystick

➤ Fermer le système et ajouter un Joystick

- ✓ Onglet LIBRARY > Inputs > Joystick



➤ Connexions

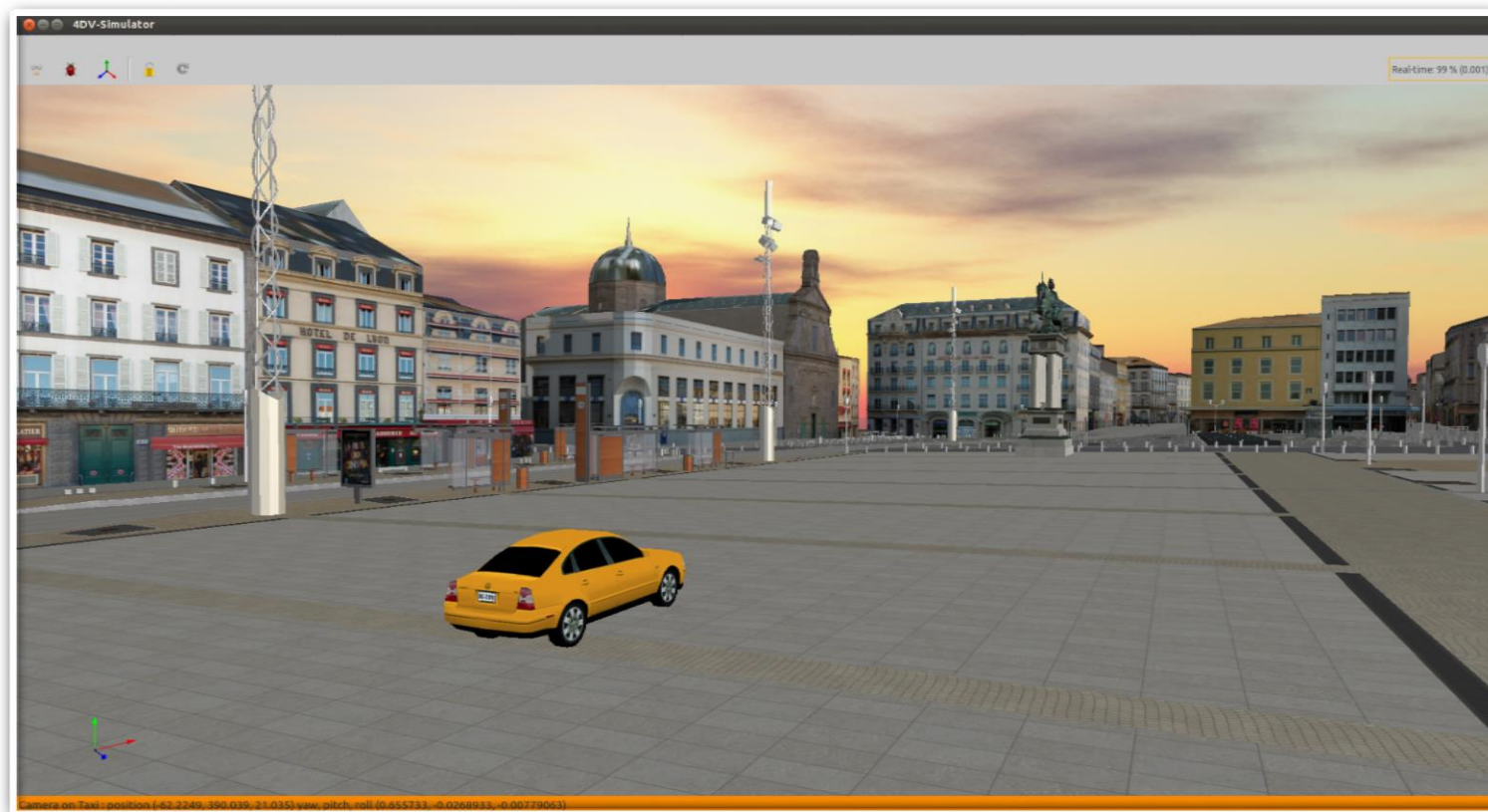
- ✓ « Speed » pour avancer et reculer
- ✓ « SteerAv » pour se diriger

Le véhicule que l'on vient de créer

➤ Lancer le simulateur

- ✓ « Build » puis « Run »

Contact : support@4d-virtualiz.com



Résultat en simulation sur la Place de Jaude (Clermont Ferrand)