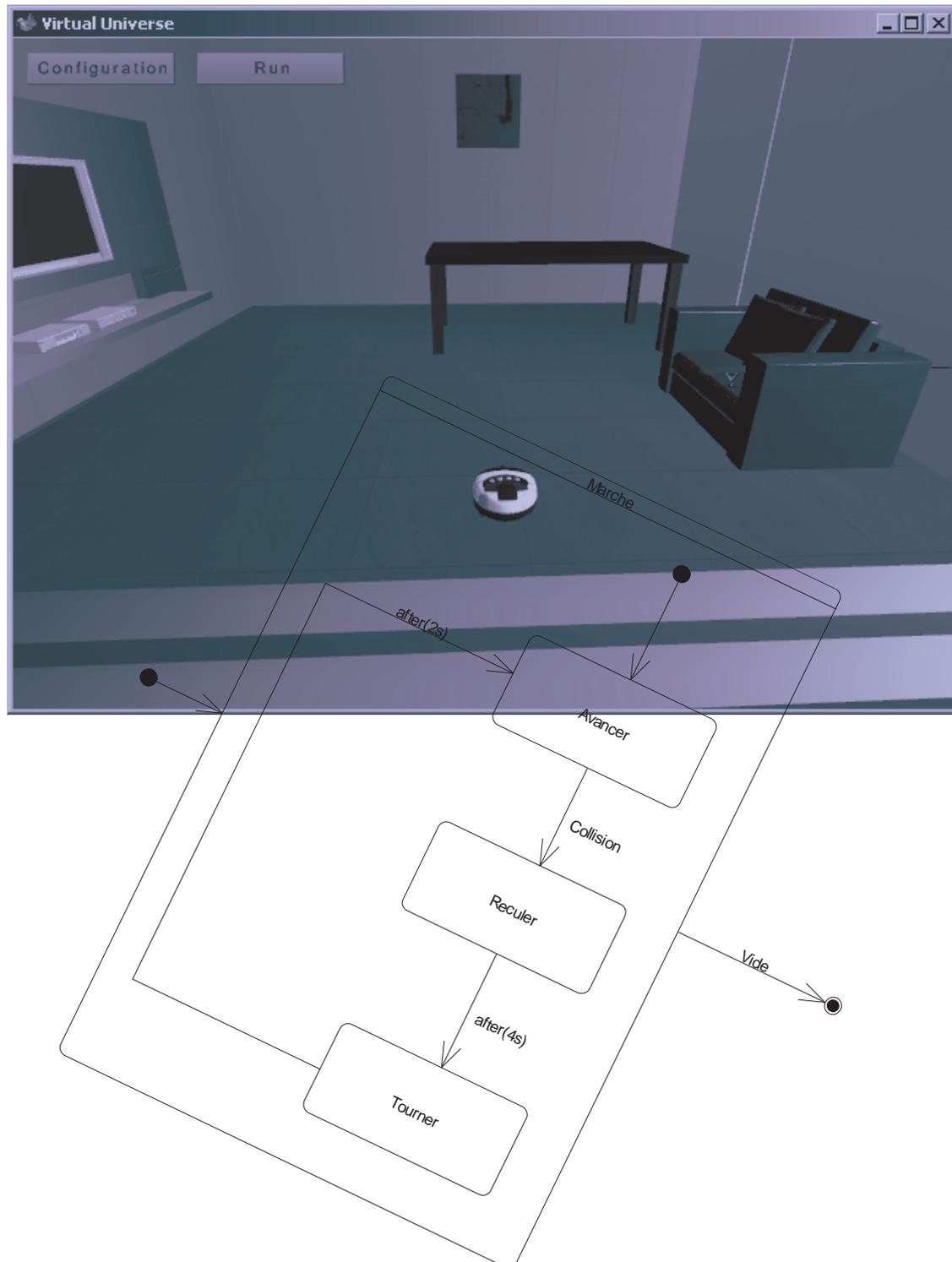


Virtual Universe

Pilotage des modèles 3D en UML

©2010 IRAI



Sommaire

Généralités	5
Configuration de Virtual Universe	5
Editeur UML.....	6
Lancement de l'éditeur	6
Création des programmes.....	6
Représentation	7
Les éléments.....	7
Accès aux entrées / sorties de Virtual Universe	10
Exécution	10
Etat du programme pendant l'exécution	10
Fenêtre de mise au point.....	12
Exemples	12
Robot aspirateur	13
Eolienne.....	14

Généralités

Ce manuel a pour but d'expliquer l'utilisation de l'éditeur UML intégré à Virtual Universe. Cet éditeur permet la création de programmes sous forme de représentation UML dans le but de piloter les modèles numériques de Virtual Universe.

Configuration de Virtual Universe

Pour que l'éditeur UML puisse dialoguer avec Virtual Universe, il faut modifier la propriété suivante dans Virtual Universe :

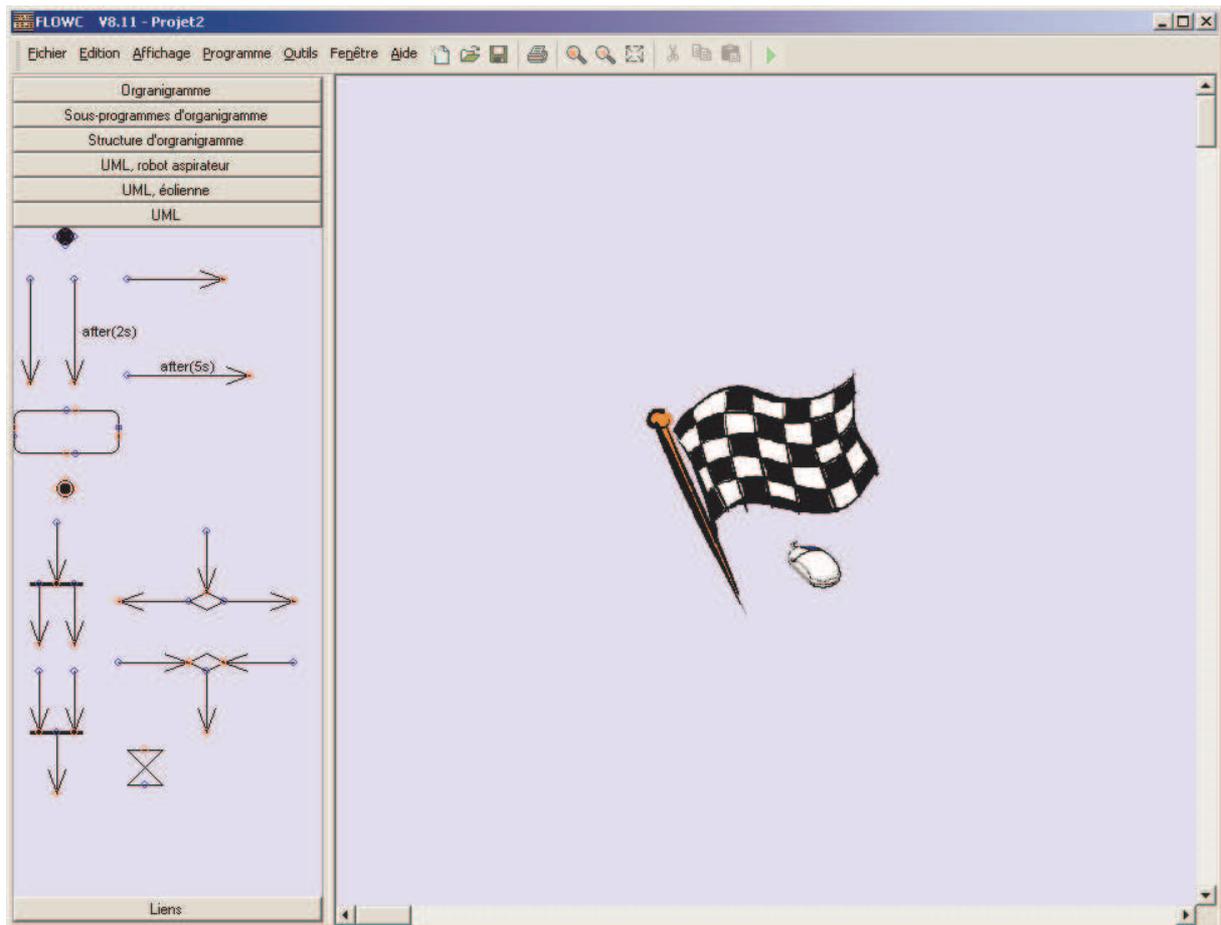
- Univers : « serveur TCP » : choisir « True ».

Editeur UML

Lancement de l'éditeur

L'éditeur se trouve dans le sous-répertoire « ae » du répertoire d'installation de Virtual Universe et se nomme « Flowc.exe ». Un raccourci de lancement est également créé dans le menu Démarrer/programme de Windows à l'installation de Virtual Universe.

L'éditeur se présente sous la forme suivante :



Création des programmes.

Saisissez les éléments sur la partie gauche de l'écran et déplacez-les sur la page. Des palettes préconfigurées pour les exemples sont également disponibles.

Vous pouvez ouvrir un assistant en cliquant avec le bouton droit de la souris sur la page et en choisissant « Ajouter un objet ».

Pour documenter les éléments, double cliquez dessus (ou sélectionnez-les, puis cliquez dessus avec le bouton droit de la souris et choisissez « Propriétés »), la boîte de dialogue des propriétés de l'objet s'ouvre alors.

Les éléments peuvent être déplacés et redimensionnés (saisir un des carrés affichés autour des éléments lorsqu'ils sont sélectionnés). Des pastilles colorées montrent les connexions des éléments. Les éléments peuvent être placés pour que les pastilles coïncident, des liaisons peuvent également

être créées en cliquant sur les pastilles (le curseur affiche une forme de cible) puis en cliquant sur le chemin où doit être tracé la liaison et en finissant par un clic sur la pastille de destination. Seules les pastilles de mêmes couleurs peuvent être reliées.

Représentation

La représentation utilisée est UML. Cette notice ne détaille pas les règles d'utilisation de la norme UML, merci de vous reporter à la norme ou à des manuels spécialisés si vous souhaitez apprendre les concepts d'UML.

Les éléments



Cet élément marque le début du programme. Il devrait y avoir au moins un élément « Etat initial » dans un programme. Cet élément est actif au démarrage du programme

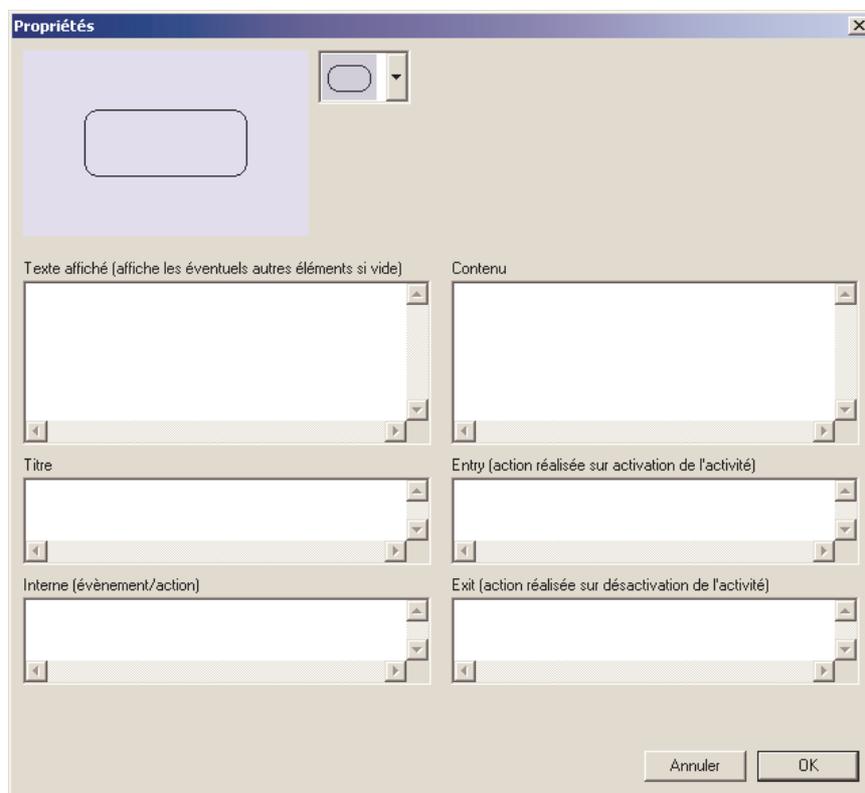


Cet élément marque la fin du programme. Cet élément peut ne pas être utilisé si le programme ne doit jamais se terminer.



Cet élément est une activité.

Voici la boîte de dialogue des propriétés d'une activité :



La zone « contenu » peut contenir du code à exécuter.

La zone « Entry » peut contenir du code qui sera exécuté une fois sur activation de l'activité.

La zone « Exit » peut contenir du code qui sera exécuté une fois sur désactivation de l'activité.

La zone « Interne » peut contenir du code qui sera exécuté de façon conditionnelle.

La syntaxe à utiliser pour ces éléments est celle du langage Basic décrite dans le manuel « Editeur de script » de Virtual Universe.

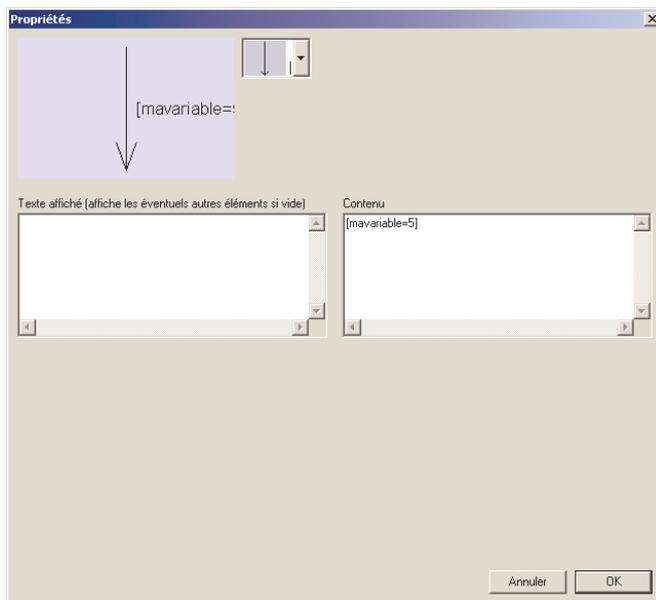
Pour répondre à la norme UML, la syntaxe « do/action » peut être utilisée.

La zone « Texte affiché » peut contenir un texte qui sera affiché à la place du code.

La zone « Titre » peut contenir un texte qui sera affiché comme titre de l'activité.



Cet élément est une transition. La syntaxe à utiliser est celle d'un test du langage Basic : une expression telle qu'on peut l'écrire derrière une instruction IF du langage Basic. Par exemple :

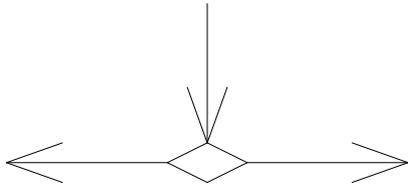


Pour répondre à la norme à la norme UML :

- des crochets peuvent être utilisés pour encadrer le test,
- le test peut être encadré par « when(...) »,
- le test peut être « after(durée) », par exemple : « after(10s) ».

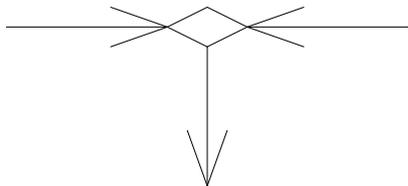
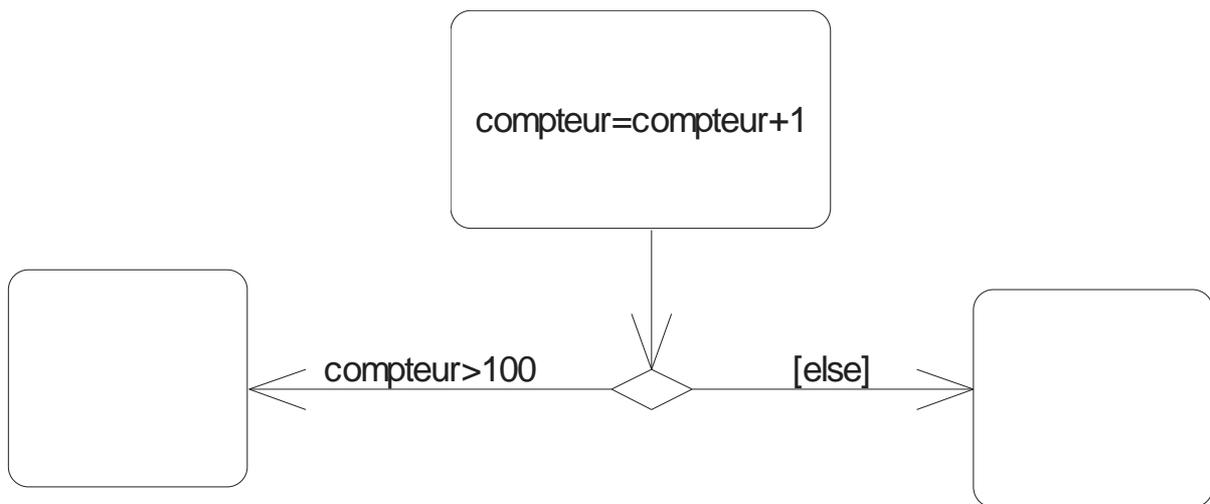
Si la zone « Contenu » est vide, la transition est toujours passante.

Ici aussi la zone texte permet de substituer un texte pour l'affichage sur la page

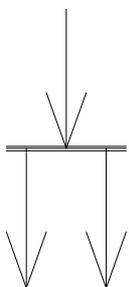


Cet élément représente une décision. Les transitions horizontales permettent de spécifier les tests. Le contenu d'une des 2 transitions peut être documenté avec le mot clé « [else] ».

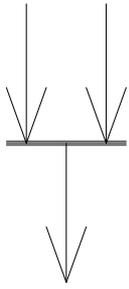
Exemple :



Cet élément est une fusion, les flux de contrôle provenant des deux transitions horizontales fusionnent.



Cet élément est une ouverture de synchronisation, le flux de contrôle provenant de la transition du haut est communiqué simultanément aux deux transitions du bas.



Cet élément est une fermeture de synchronisation. Le flux de contrôle doit arriver sur les deux transitions du haut pour être transmis à la transition du bas.



Cet élément représente une attente. La durée de l'attente est à définir dans ses propriétés.



Cet élément représente l'état History (voir la norme UML pour plus d'informations).

Accès aux entrées / sorties de Virtual Universe

Si les noms des variables utilisées dans le script commencent par la suite de caractères « in_ » ou « out_ », alors les variables concernées sont considérées comme des variables externes appartenant à Virtual Universe.

Plus précisément, les variables commençant par « in_ » seront lues depuis Virtual Universe et les variables commençant par « out_ » seront écrites vers Virtual Universe.

Les caractères suivants « in_ » ou « out_ » correspondent à un nom de comportement ou un alias de nom de comportement de Virtual Universe.

Par exemple, « in_capteur » sera une variable lue depuis l'état du comportement nommé « capteur » de Virtual Universe. « out_moteur » sera une variable écrite vers l'état du comportement nommé « moteur » dans Virtual Universe.

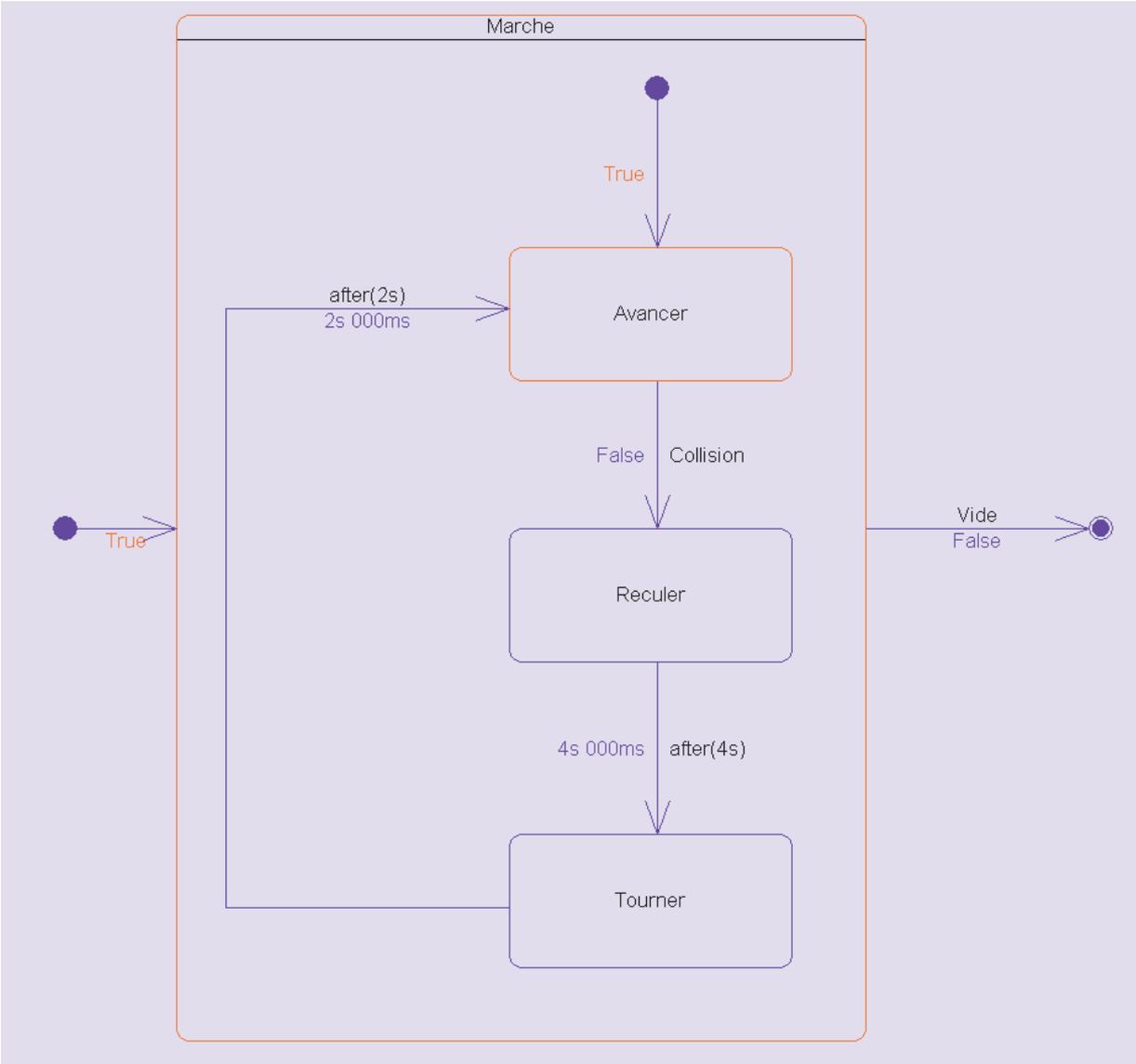
Exécution

Pour lancer l'exécution du programme, cliquez sur le bouton « PLAY » (triangle vert) dans la barre d'outils ou sélectionnez « GO » dans le menu « Programme ». Virtual Universe doit être lancé et le projet Virtual Universe doit être réglé comme serveur TCP (paramètre dans les propriétés de l'univers). Pour mettre fin au programme, cliquez sur ce même bouton (il prend l'aspect d'un bouton stop quand le programme est lancé).

Etat du programme pendant l'exécution

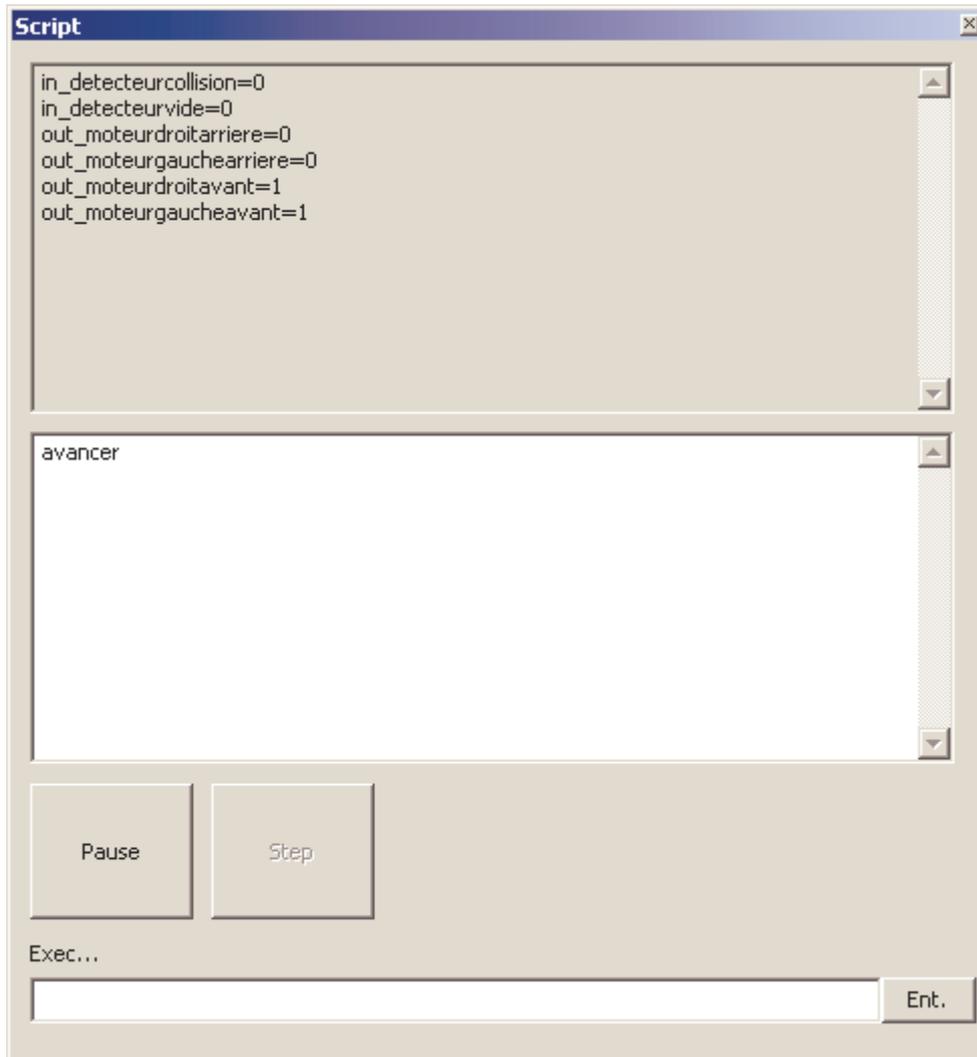
Pendant l'exécution, les états des éléments sont affichés (éléments actifs en rouge, indication False/True pour les transitions) en dynamique dans l'éditeur. Les durées en cours associées aux tests liés au temps sont également affichées.

Exemple :



Fenêtre de mise au point

Au lancement de l'exécution du programme, la fenêtre suivante s'ouvre :



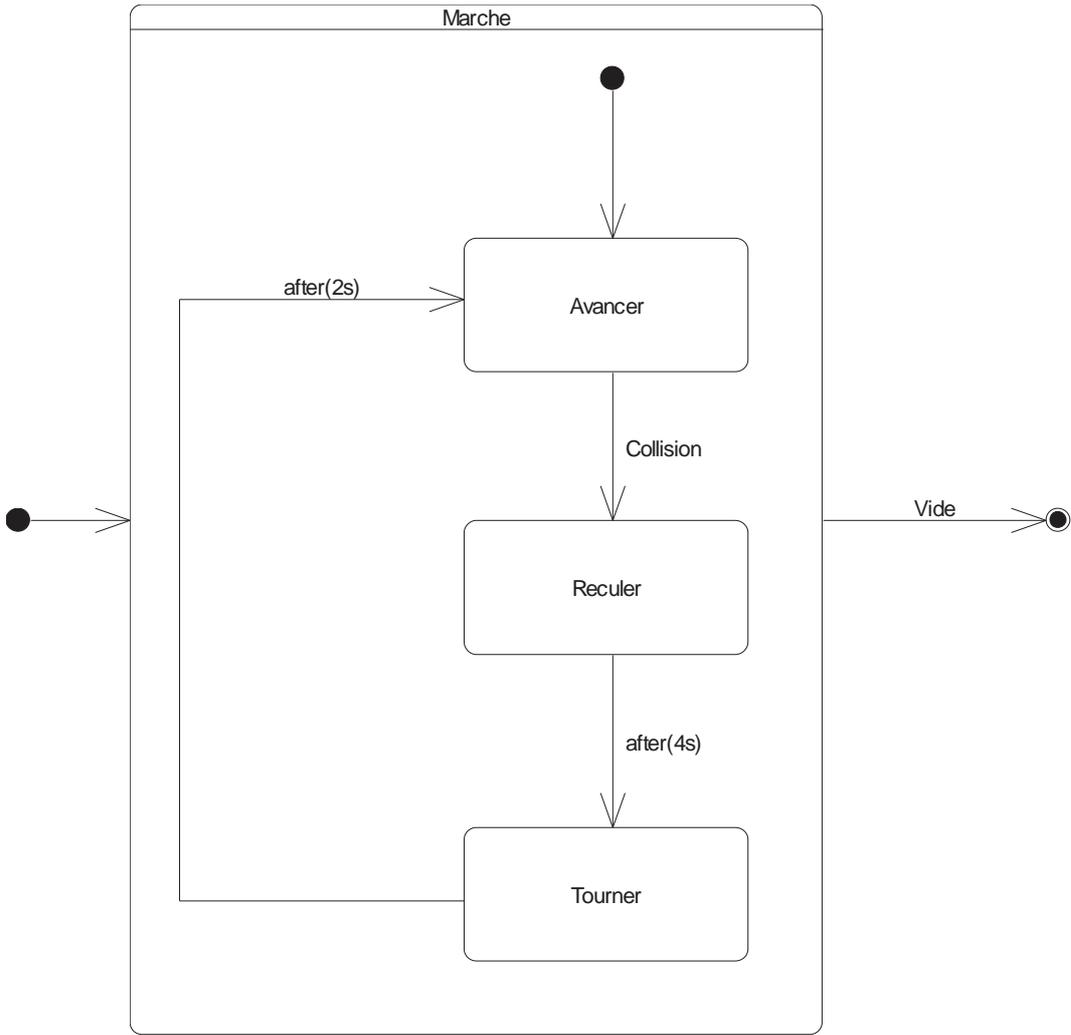
L'état des variables ainsi qu'une fenêtre affichant les sorties des commandes « print » utilisées dans le programme sont affichées. Le bouton « Pause » permet de mettre le programme en pause ou de reprendre son exécution. Le bouton « Step » permet de faire avancer le programme d'un pas.

La zone « Exec... » permet de saisir du code qui sera exécuté une fois si le bouton « Ent » est cliqué. Ceci permet par exemple d'affecter une valeur à une variable. Tapez par exemple « mavariable=10 » puis cliquez sur « Ent ».

Exemples

Le sous-répertoire « Exemples » du répertoire d'installation de Virtual Universe contient des exemples Virtual Universe pilotables par UML ainsi que les fichiers UML correspondants (fichiers .FLC).

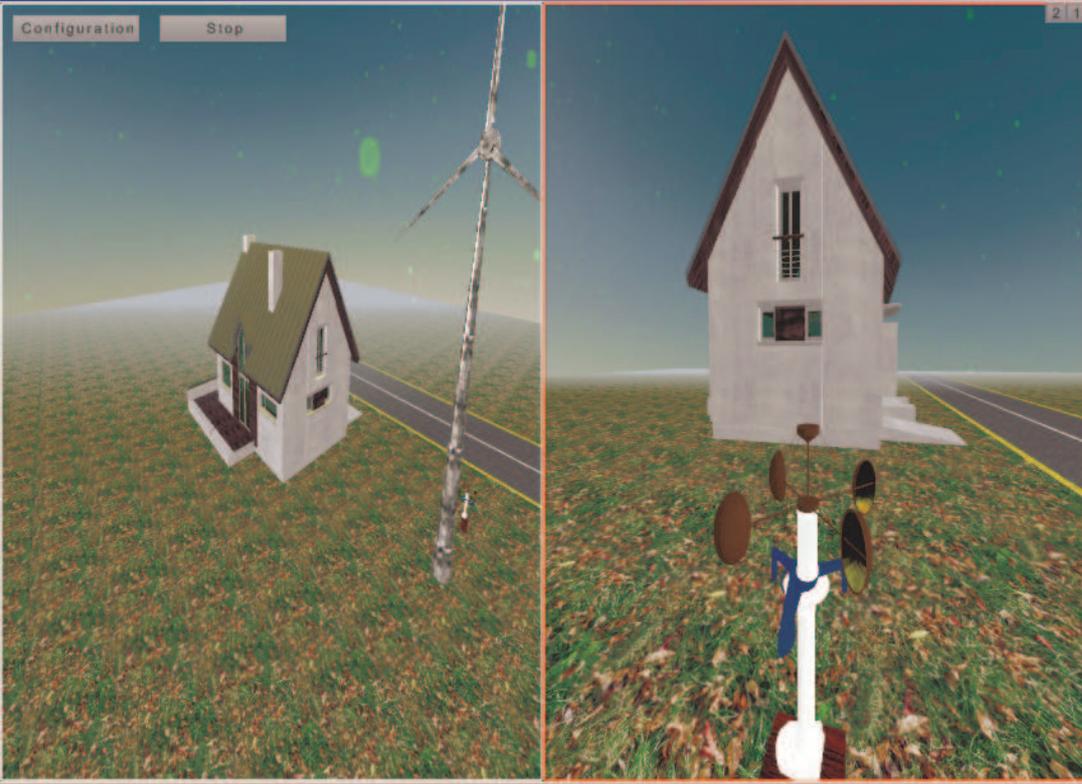
Robot aspirateur



Eolienne

Virtual Universe

Configuration Stop



Eolienne



Tourner sens horaire

Tourner sens anti-horaire

Drapeau

Vitesse Vent

Direction Vent

Direction Eolienne

Vitesse De Rotation de l'éolienne

