# Les outils de simulation de systèmes automatisés

Les logiciels de simulations de systèmes automatisés connaissent un essor important depuis quelques années avec l'apparition de produits de plus en plus performants et accessibles en terme de prix. Même si des solutions étaient dans les faits déjà utilisées en détournant l'utilisation de produits de supervision à des fins de simulation, il semble évident que la simulation des systèmes automatisés avec un logiciel informatique est entrée dans une phase active et ascendante. La raison en est l'accès à des technologies sophistiquées à moindre coût rendu notamment possible par l'industrie du jeu vidéo ainsi qu'une prise de conscience de l'importance de la simulation par un plus grand nombres d'acteurs industriels de l'automatisation au delà des secteurs habituels où la simulation est depuis longtemps indispensable (dans le secteur nucléaire par exemple).

## Buts de la simulation

Les principaux buts visés par la simulations sont : la mise au point des programmes automates (plus connue sous le nom anglais de virtual commissioning), l'aide à la définition d'un système automatisé (définition des cadences, etc.), la formation des opérateurs, la simulation de pannes et la présentation d'un système automatisé (pour aider à le vendre avant sa réalisation par exemple).

Dans l'industrie, la mise au point des programmes automates a pour but, au delà de la fiabilisation de ceux-ci, de réduire les temps de mise au point sur site, le gain dans ce cas sera réalisé entre le temps consacré à la création du modèle virtuel et le temps "économisé" sur la mise au point sur site. Cette économie sera encore plus importante dans le cas de la mise au point d'un système sur un site éloigné géographiquement, ou dans une ambiance difficile ou encore sur un système critique (temps d'arrêt pour modifications et remise en service devant être le plus possible réduits).

## Technologie

Les principaux constituant d'un logiciel de simulation de systèmes automatisés sont les suivants : un moteur de rendu, un moteur physique, un module de dialogue permettant à un système externe au logiciel (automate ou automate virtuel) de piloter le système automatisé, plus d'éventuels modules annexes, gestion d'une IHM, simulations des schémas, etc.

### Le moteur de rendu

Il permet simplement de visualiser le système automatisé, il peut aujourd'hui bénéficier de fonctionnalités facilement accessibles sur un ordinateur moderne : 3D, multi caméras, objets texturés, etc. Et pourquoi pas affichage en vrai 3D avec lunettes 3D.

### Le moteur physique

Les moteurs physiques sont désormais accessibles sur de nombreuses plateformes, il convient d'être prudent et de comprendre en quoi leur utilisation est importante mais parfois délicate, nous parlerons plus loin dans cet article du "delta" entre la simulation et le monde réel.

La technologie développée par NVidia avec son moteur Physx possède l'avantage de pouvoir s'exécuter sur le processeur de la carte vidéo ce qui peut être intéressant en terme de performances de simulation notamment si la taille du système à simuler est importante.

### Le module de dialogue

Il permet de piloter le modèle virtuel à partir d'un ou plusieurs automates programmables réels ou simulés (simulateurs intégrés aux logiciels constructeurs) connectés au PC. Des passerelles universelles (OPC par exemple) peuvent aussi être utilisées bien qu'engendrant parfois un surcroit de complexité au niveau de la configuration des échanges. Le rôle du module de dialogue sera d'échanger les entrées / sorties entre le programme de pilotage du système et le système simulé. Idéalement, ces échanges s'effectueront en un temps le plus court possible et n'entraineront aucune modifications sur le programme de l'application d'automatisme. Le but ultime est de "laisser croire" au système pilotant qu'il est en lien avec un système réel et non une simulation. Pour fixer cette idée, pensons au film de science-fiction "Matrix" et sa simulation de monde réel usurpant l'esprit humain.

### Autres modules

D'autres éléments pourront se greffer à la simulation, citons par exemple : la création d'IHM, la simulation de la partie schémas (électriques, pneumatiques, hydrauliques, etc.), le lien avec un simulateur numérique (type Matlab simulink), un module de simulation de pannes, etc.

L'ouverture du logiciel de simulation semble importante par rapport à la multiplicité des systèmes pouvant venir se greffer à cette simulation. Prenons l'exemple de la simulation de la gestion des bagages sur des tapis roulants d'un aéroport, il paraît plus qu'intéressant que la simulation puisse accéder à une base de données pour gérer l'aiguillage des bagages.

## Performances

La performance du système de simulation est un point clé et critique. Il convient en effet de ne pas s'éloigner trop du modèle réel et de sa "réactivité".

Bien que dépendant du PC utilisé pour gérer la simulation, la conception de la structure du simulateur reste au moins d'importance égale.

Idéalement, le simulateur devra privilégier la réactivité physique aux tâches de rendus ou tout du moins ne pas rendre tributaire de la tâche de rendu la simulation physique et le dialogue avec le logiciel externe ou l'automate. Les technologies multi tâches permettent aujourd'hui de gérer ceci de façon correcte.

L'amélioration de la vitesse de rendu pourra quand même être traitée dans le cas de géométries anormalement complexes (ce qui est parfois le cas lorsqu'on exporte des géométries depuis les logiciels de CAO) par la réduction de celle-ci (technologie de décimation de triangles) ou la suppression d'éléments n'apportant que peu d'intérêt à l'aspect visuel de la simulation (éléments de faibles volumes, vis et écrous de petites tailles, etc.). Il conviendra dans ce domaine de trouver un bon compromis entre un aspect visuel de qualité et un affichage fluide du système simulé, ceci se faisant au cas par cas, suivant la taille du système simulé et les performances du PC utilisé pour la simulation.

Notons également que la possibilité de pouvoir "scinder" une simulation sur plusieurs PCs permet de s'affranchir de la limitation de traitement d'un seul PC et permettra d'envisager des projets de simulation sans réelle limitation de taille.

## Simulation vs monde réel

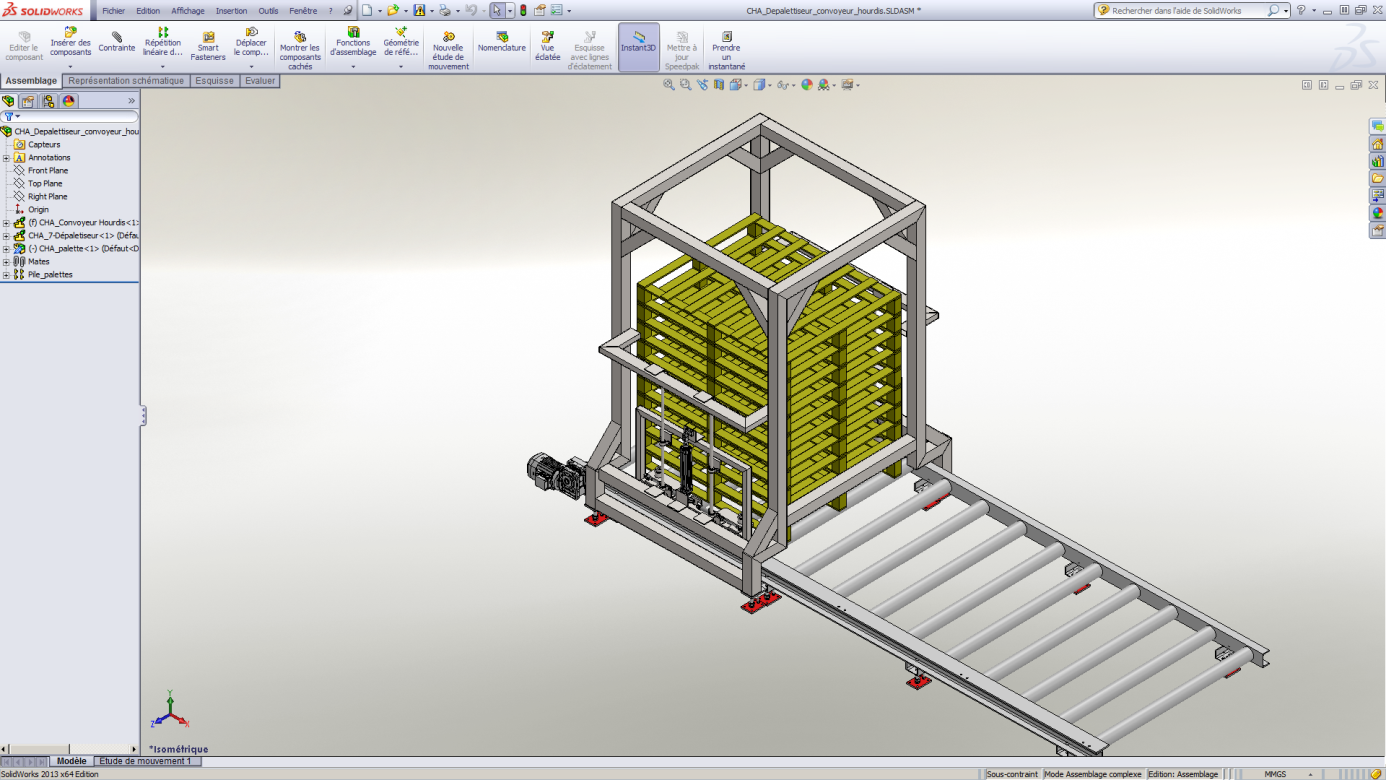
Sujet clé lorsqu'on parle de simulation : le "delta" entre simulation et système réel. La question du débutant en la matière est bien entendu "est-ce que la machine simulée fonctionne comme la machine réelle". La réponse si elle doit être oui ou non sera forcément "non", mais pour que cette réponse soit plus juste, reformulons la question de la façon suivante : "quel est l'écart entre la simulation et la machine réelle" car forcément, ce "delta" existe et il n'est pas forcément facile de le quantifier sous tous ses aspects. Ce que l'on peut dire, c'est qu'il sera important que le logiciel de simulation puisse permettre de réduire ce delta lorsque ceci sera important et également que la création de la simulation pourra admettre un delta plus important si, en contre partie, la création de la simulation peut en être facilitée. Le créateur de la simulation sera ensuite libre de fixer "le curseur" du réalisme de la simulation en fonction de la quantité de travail à fournir et des buts à atteindre. Prenons pour illustrer ceci le cas d'un simple actionneur pneumatique, une simulation basique sera de définir un simple temps pour la sortie de ce vérin, à l'opposé, la définition des formules physiques de calculs de pressions, de pertes de charges, de forces, etc. permettraient d'obtenir une simulation beaucoup plus proche de la réalité. Si le but est simplement, vis à vis d'un programme automate de contrôler qu'un mouvement a bien été réalisé en un temps donné et d'avoir l'information de fin de course, la définition du temps de la course semble suffisant. Encore une fois, l'outil de simulation devra proposer et mettre à disposition des outils de simulation aussi sophistiqués que possible mais ne pas en imposer l'utilisation systématique. Ce choix devra être réservé au concepteur de la simulation.

## Apports de la simulation de systèmes automatisés dans l'enseignement

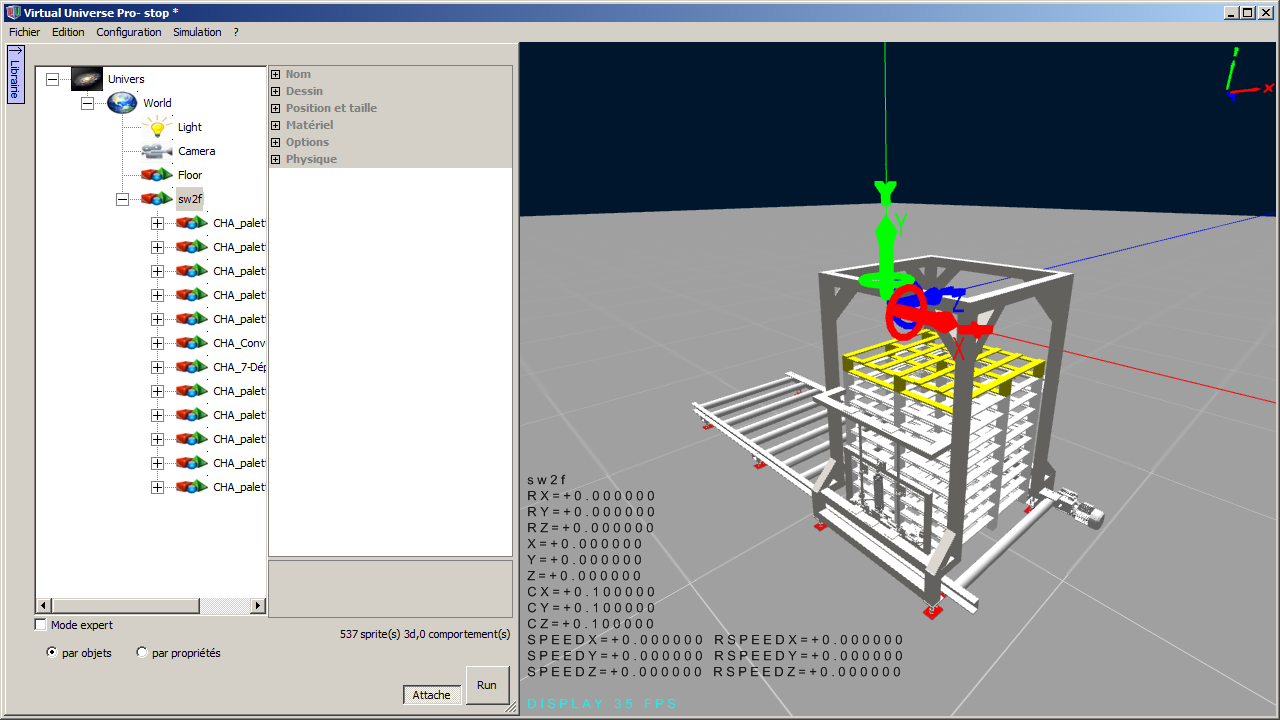
Officialisée dans certains programmes comme pour le nouveau BTS CRSA, la simulation de systèmes automatisés offre de nombreux avantages : multiplication des systèmes à moindre coût, même s'il ne faut bien entendu pas céder au "tout simuler" pour des raisons évidentes d'éloignement des contraintes du monde réel.

### Etude d'un cas de création d'une machine simulée avec Virtual Universe Pro

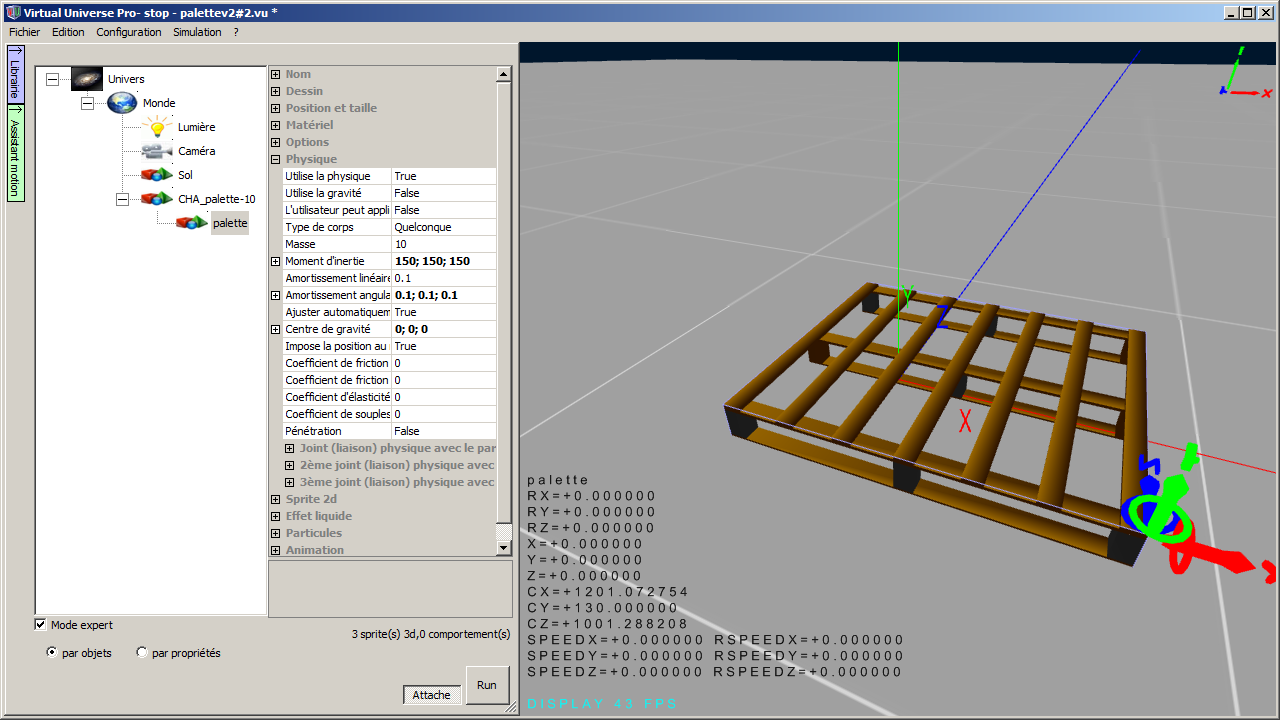
Nous partirons d'un modèle numérique de dépalettiseur (thème de BTS CRSA du Lycée Vauvenargues d'Aix en Provence) réalisé sous Solidworks:



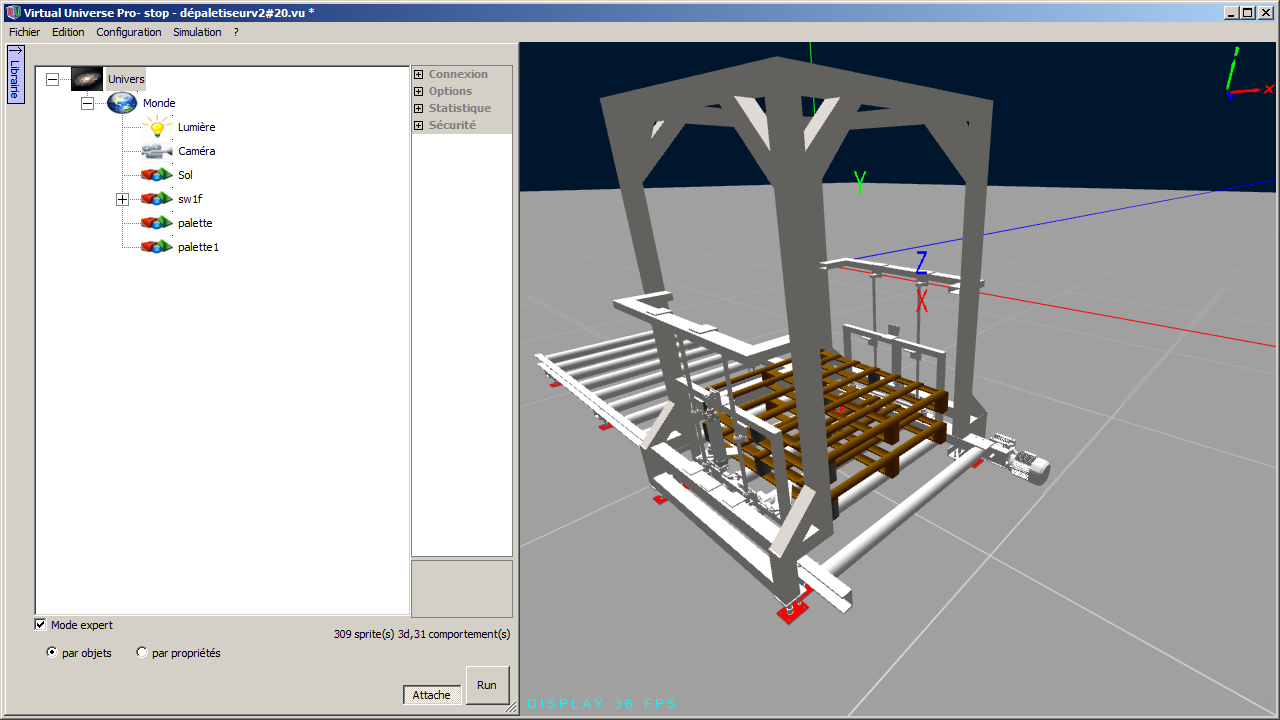
La procédure de transfert de la maquette consiste à exporter depuis Solidworks au format 3DXML (par un simple "Enregistrer sous") et à importer ce même fichier directement dans Virtual Universe Pro :



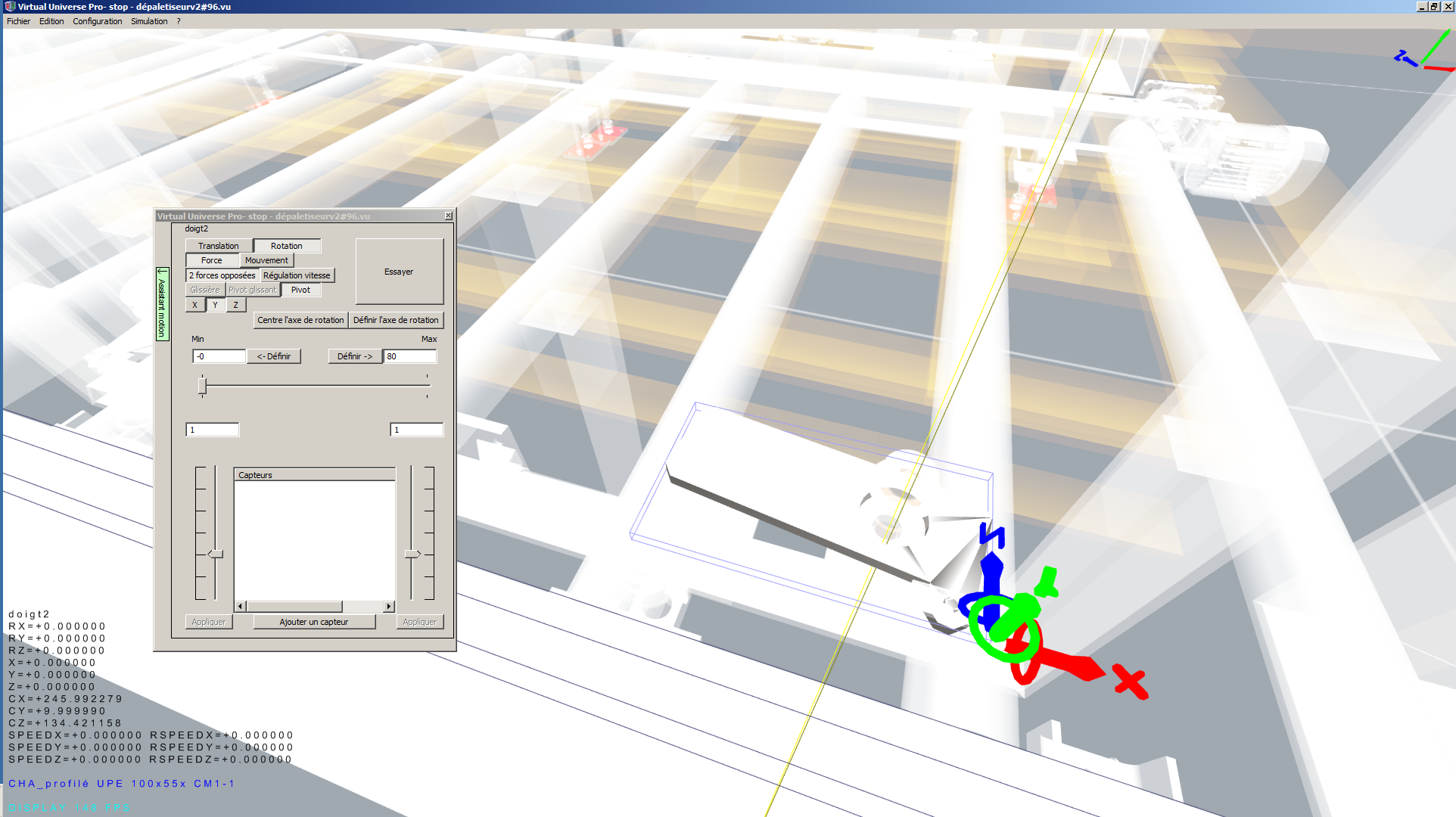
Une fois importée la partie "dessin" de la machine, une phase configuration des mouvements et autres comportements débute. Intéressons nous tout d'abord aux pièces "libres", les palettes. Après isolation de la pièce (par commodité), les propriétés physiques de celle-ci sont définies. L'aspect (ici la couleur) peut également être retouché.

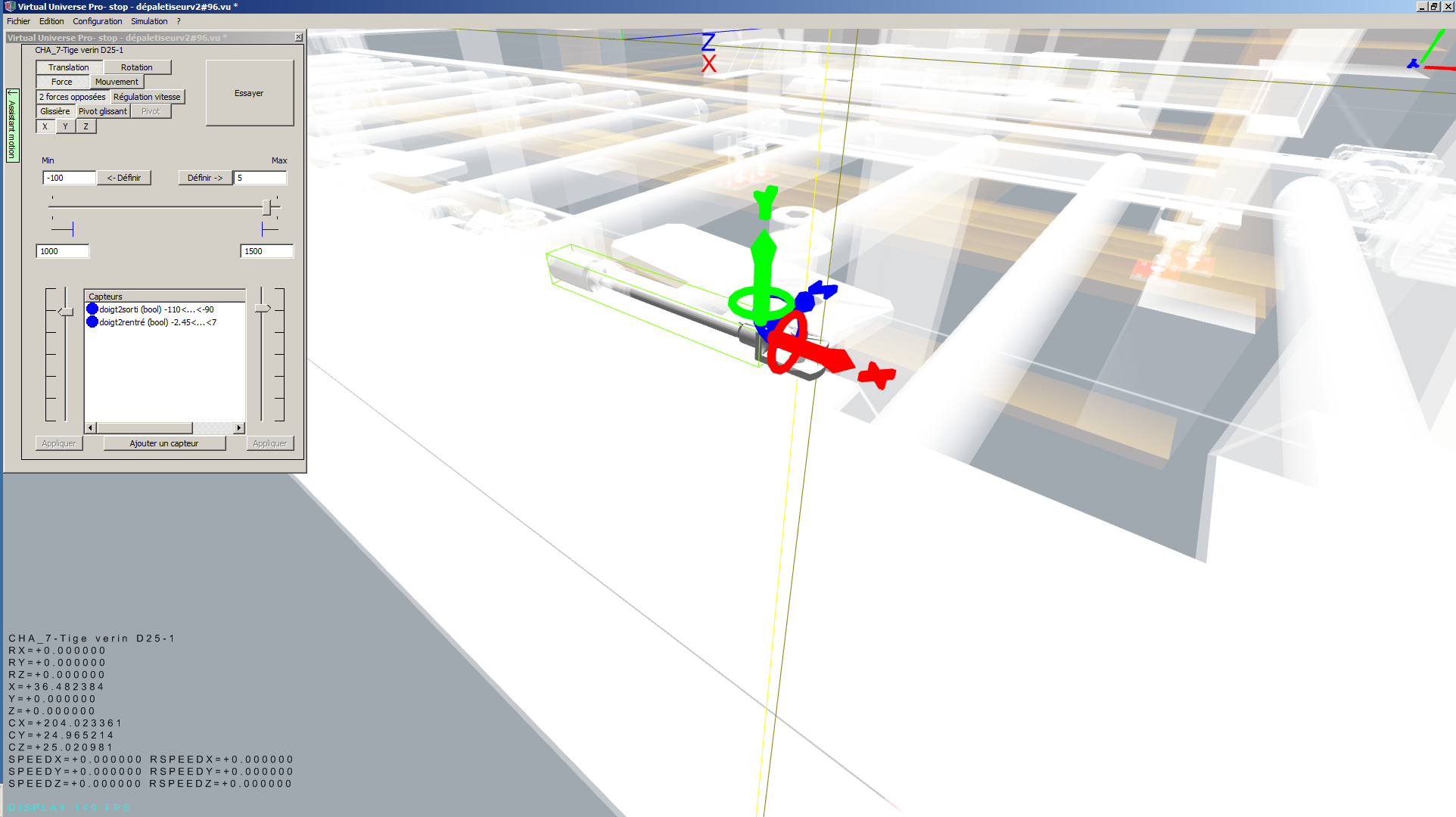


L'objet ainsi paramétré est réintégré et dupliqué dans le modèle original.



A l'aide d'assistants, les mouvements du modèle sont définis, par exemple la rotation "des doigts" et le vérin qui anime ces même doigts.

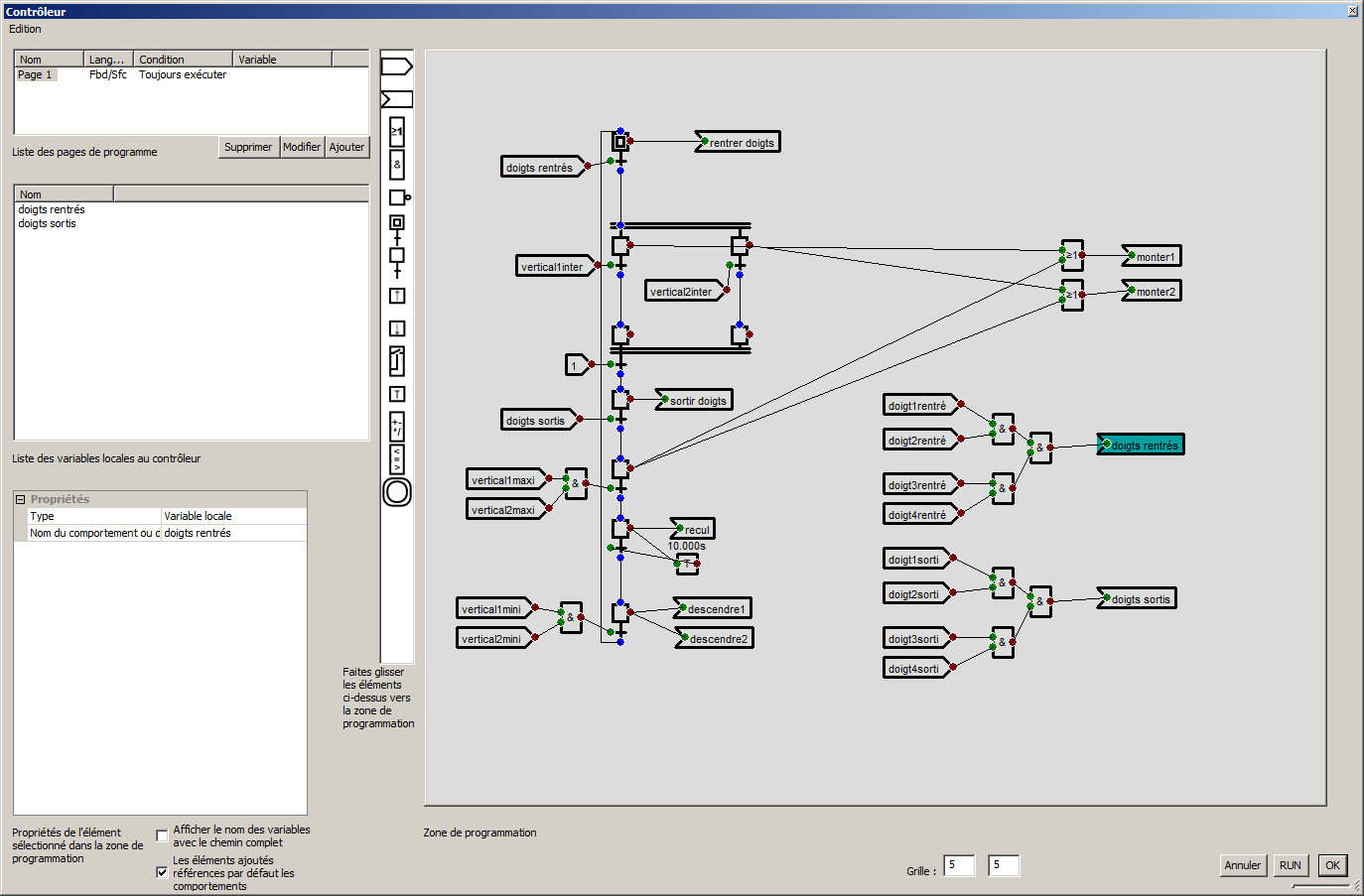




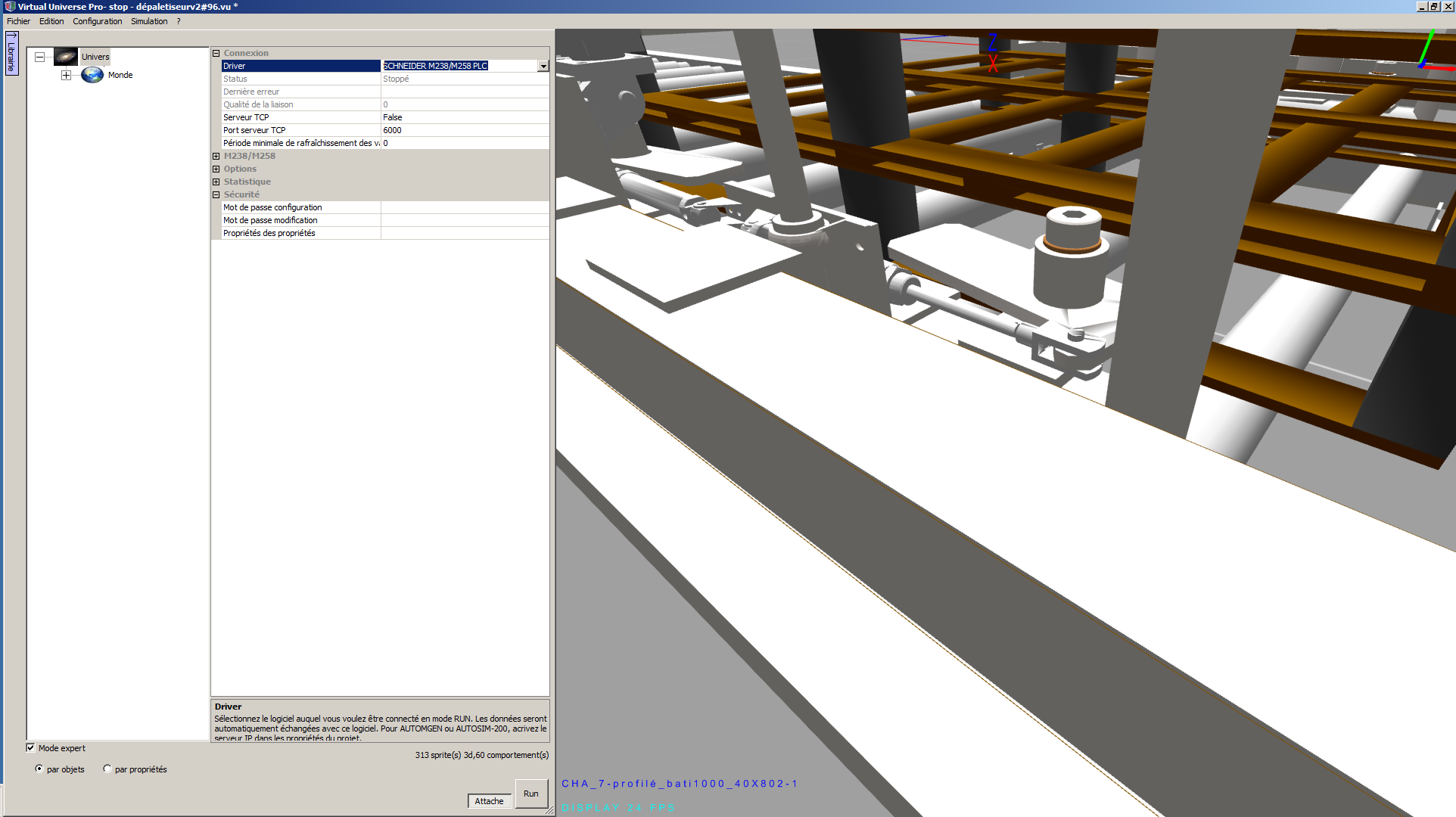
Les assistants offrent un choix complet de type de pilotage pouvant allez jusqu'à une gestion d'axes avec profils d'accélérations et décélérations.

Ces mêmes assistants permettent de définir des capteurs.

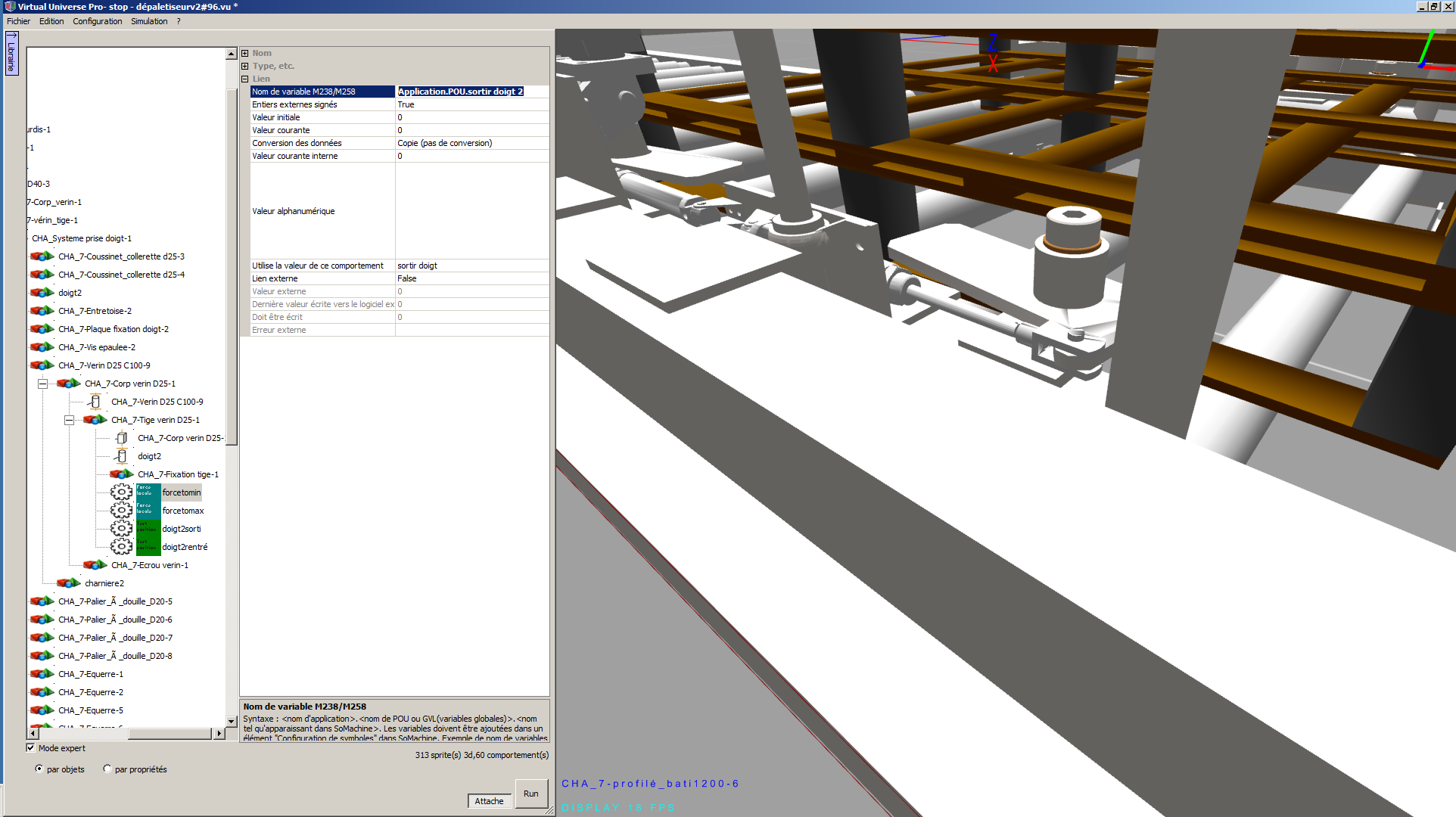
Des contrôleurs virtuels intégrés au projet de Virtual Universe Pro permettent de créer tout ou partie du programme de pilotage du système, même si l'intérêt est bien entendu in fine de piloter le modèle virtuel avec le programme qui fera fonctionner la machine réelle.



La phase de pilotage du modèle simulé par un automate ou un simulateur intégré à un logiciel constructeur reste simple : après la sélection du type de l'automate (ici un automate Schneider M258 sous SoMachine).



on associe les pilotages de mouvements et les capteurs à des variables automates.



En mode connecté à l'automate, on suivra l'évolution du programme dans l'outil constructeur et l'évolution de la machine simulée simultanément.

