**Utilisation de FlexSim**

**Auteurs :**  P. Lacomme ([placomme@isima.fr](mailto:placomme@isima.fr))

D. Lamy ([lamy@isima.fr](mailto:lamy@isima.fr))

**Date de création :** Janvier 2017

**1) Installation**

Le logiciel de simulation est disponible à l'adresse suivante : <https://www.flexsim.com/fr/>



Il faut choisir la version d'évaluation, ce qui passe automatiquement, par le remplissage d'un formulaire nominatif.

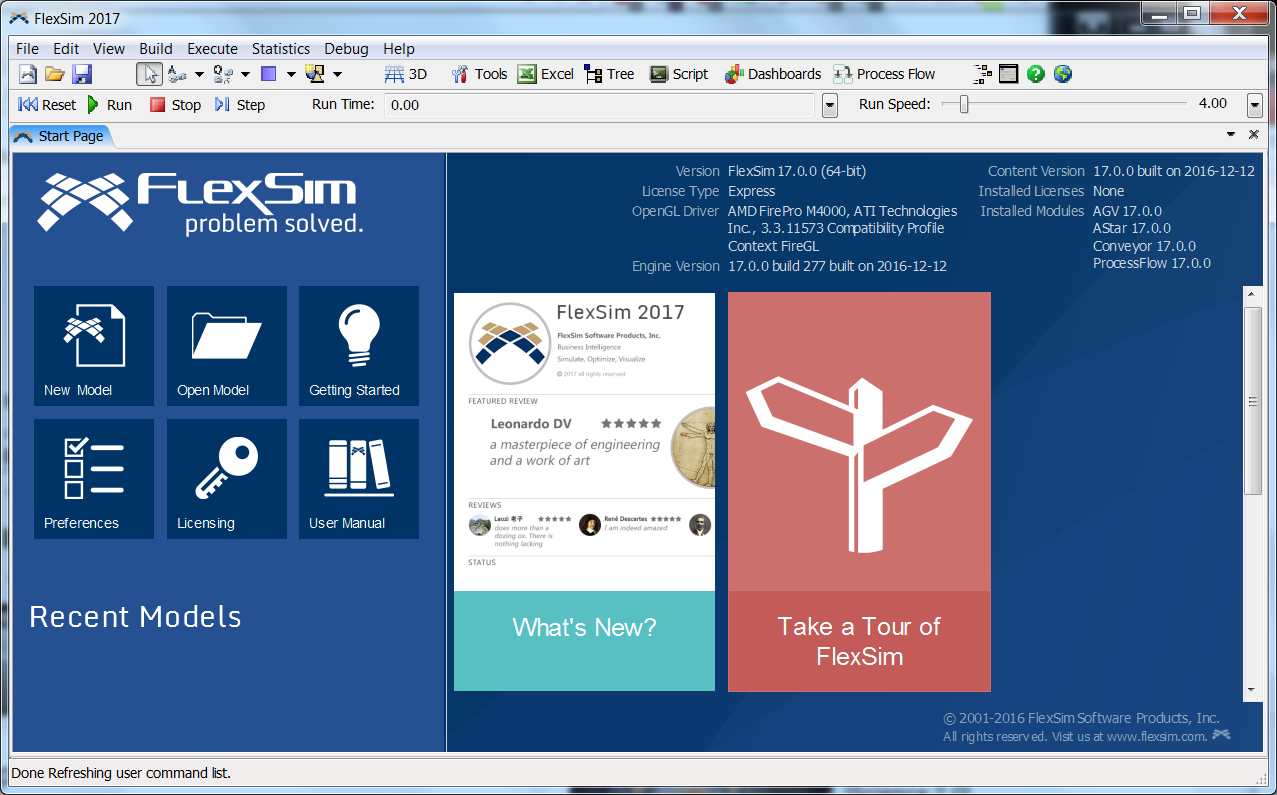


**2) Simuler une file MM1**

L'écran d'accueil offre la possibilité de créer un nouveau modèle soit par le menu **File/New Model** soit en utilisant directement New Modèle à gauche de la page.

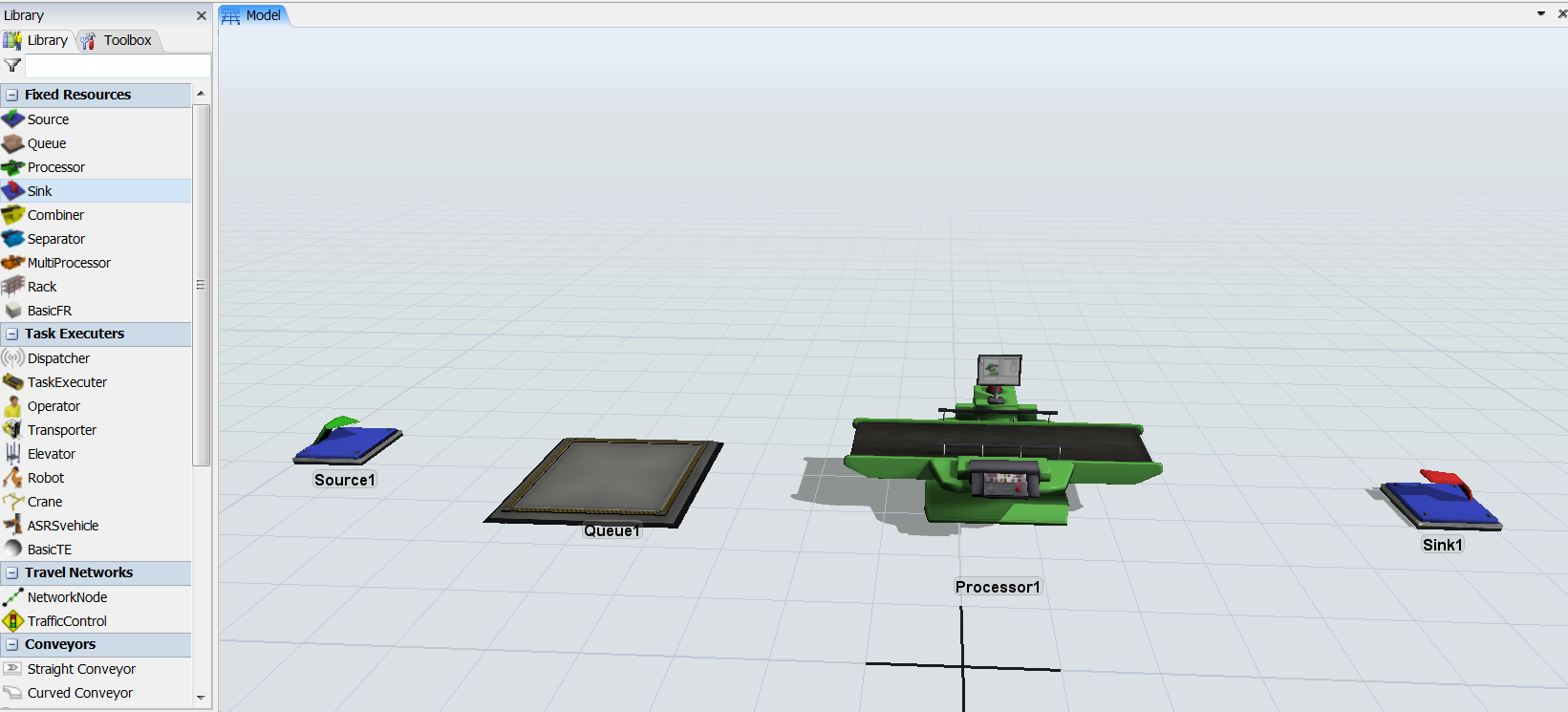
On peut trouver sur Youtube une vidéo explicative relativement claire :

<https://www.youtube.com/watch?v=Bfru1ynSTUQ&index=4&list=PLAlimuUHNaOv63JzeRMIaCXyR1kdtAMLg>

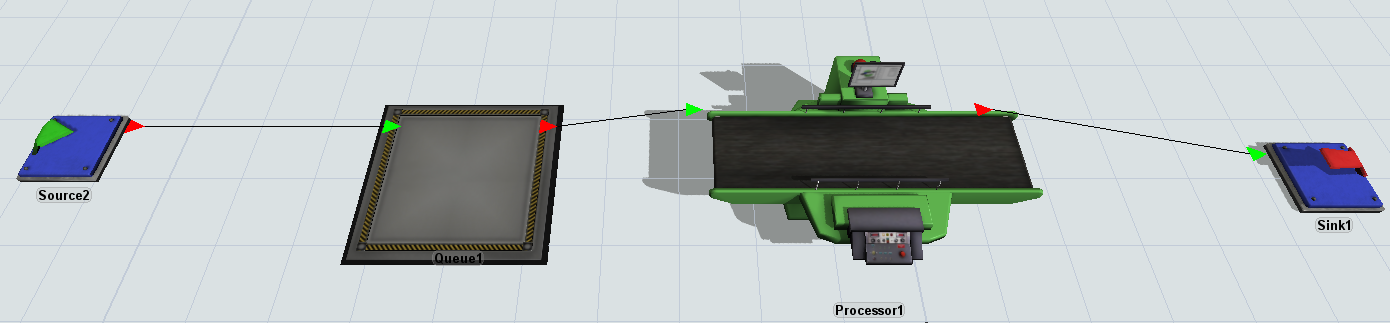


Il faut ajouter au modèle 4 éléments :

* Une source ;
* Une Queue ;
* Un Processeur ;
* Un puits.

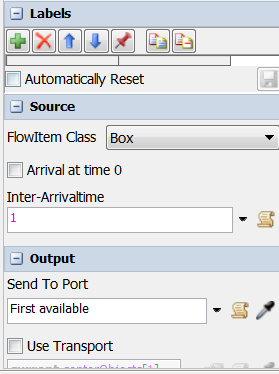


En maintenant la touche Q enfoncée, on relie les éléments entre eux.

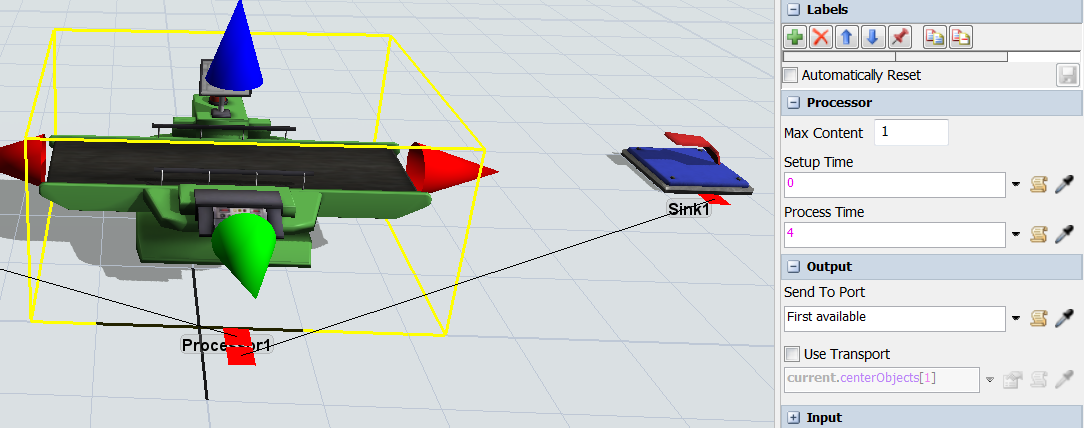


Attention à relier correctement une sortie avec une entrée, comme sur le schéma ci-dessus.

Une fois la source sélectionnée, on peut modifier la durée inter-arrivée et lui donner la valeur 1.

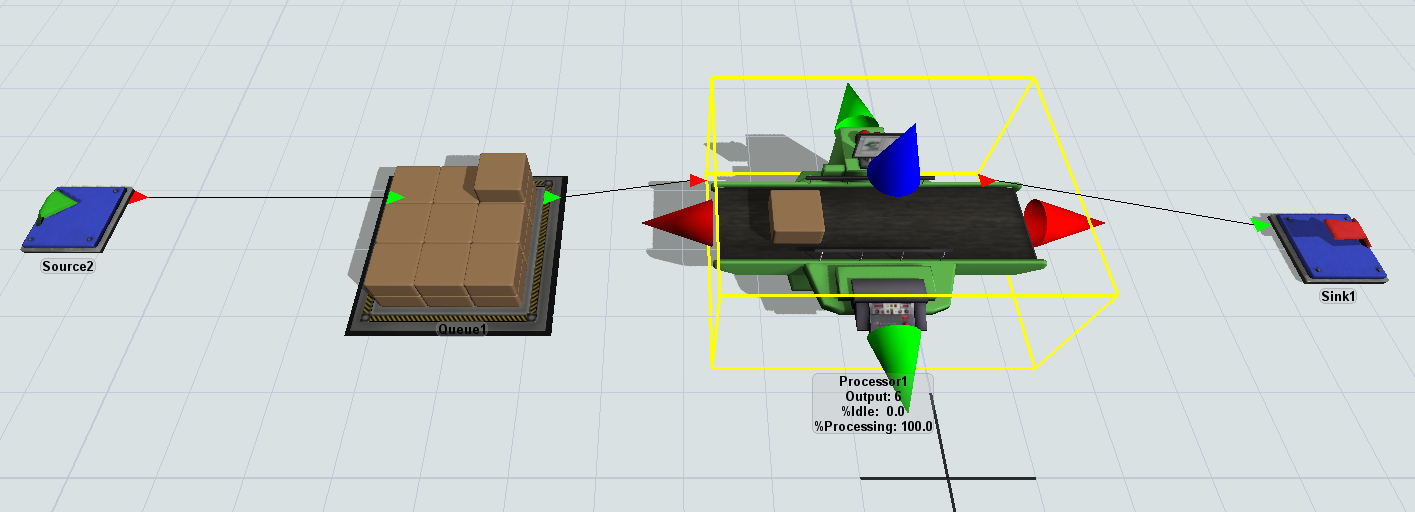


On peut ensuite affecter un **processing time** de 4 au processeur.

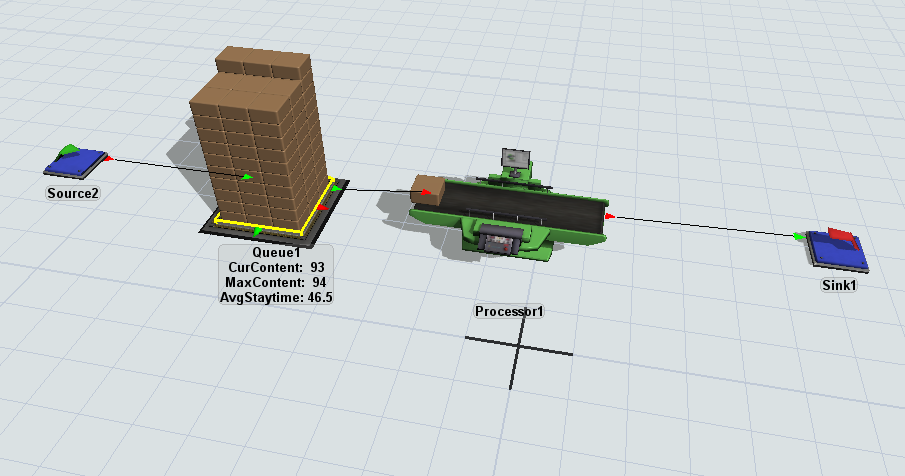


**Le résultat d’exécution**

Il permet d’avoir une animation avec une qualité très élevée en 3D.

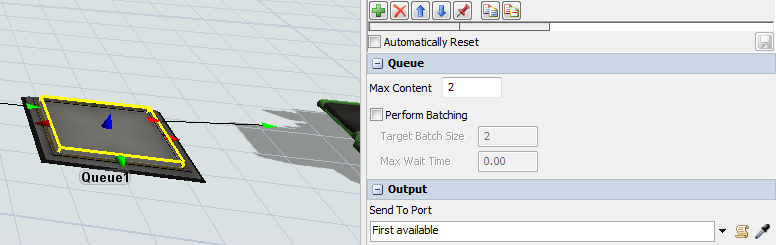


Comme la taille de la file n’est pas limitée, on obtient un état du stock en augmentation.

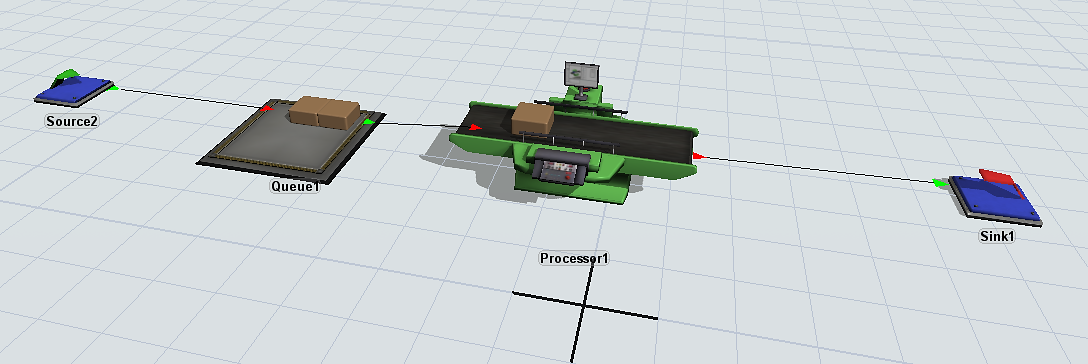


**3) Simuler une file MM1 avec un stock de capacité limitée**

Il faut changer l’attribut **Max Content** de la Queue et lui affecter, une valeur, par exemple, la valeur 2.

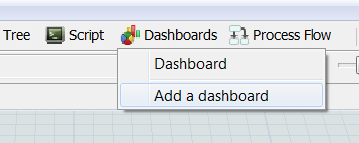


Ce qui donne à l’exécution la visualisation suivante :

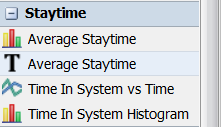


**Analyse des résultats**

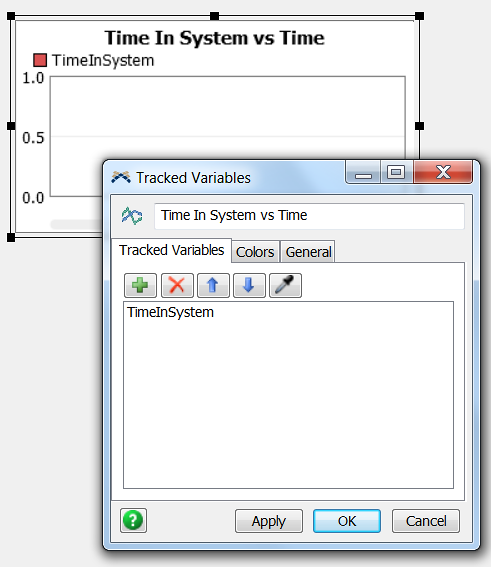
On peut ensuite ajouter un **Dashboard**, c'est-à-dire un état de sortie présentant les statistiques et autres éléments. Pour cela, dans le menu **Dashboard** choisir **Add** **a** **Dashboard**.

****

Dans le **dashbord**, on peut ensuite ajouter une courbe de type **Time in System vs. Time** dans la section **StayTime**.



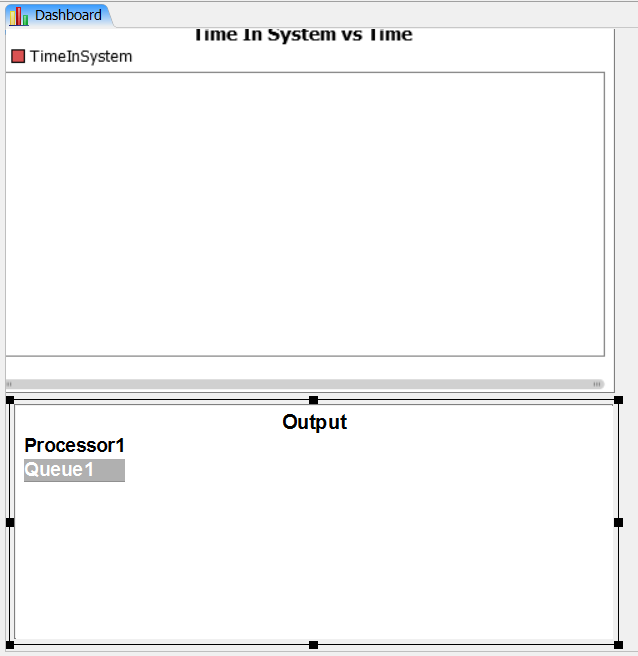
Une fois posé sur le panel, un double clic sur l'objet permet encore de modifier les différents éléments via le menu contextuel.



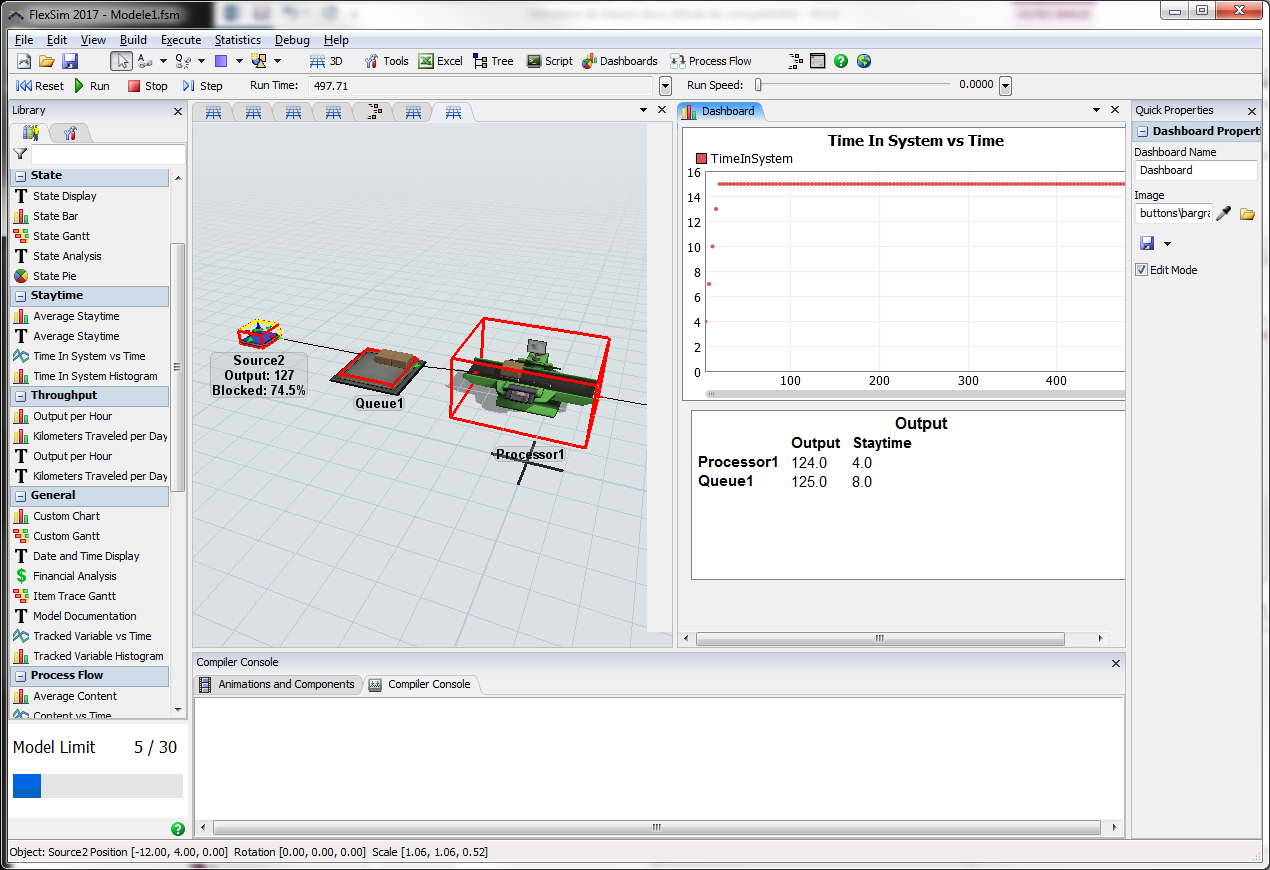
|  |  |
| --- | --- |
|  |  |

Une fois crée, on peut facilement sélectionner les éléments du modèle et les attributs à afficher en utilisant un double clic sur le panel **Output**.

Au final, le **dashboard** peut se présenter comme suit :



Pendant le déroulement du modèle, les courbes et différents états de sortie sont mis à jour.



Une analyse précise de la courbe permet d’identifier les différents temps de séjour. Ces données méritent quelques éléments d’explication.



Le temps de séjour dans le système est compté à partir du moment où la station d’entrée génère la pièce.

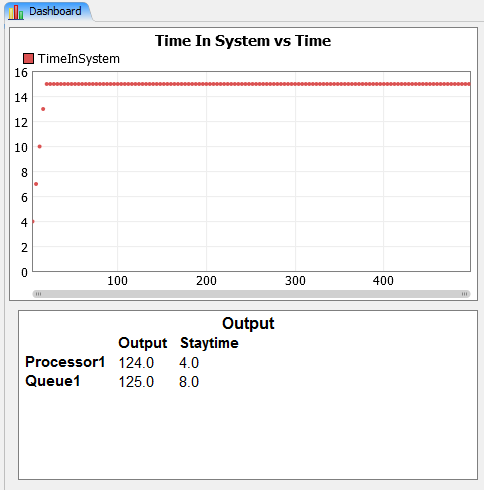
Ainsi la pièce 0 est générée à la date 0 et sort à la date 4. Ceci correspond au premier point rouge de la courbe.

La pièce 1 est générée à la date 1 est sort à la date 8 et elle a donc passé 7 unités de temps dans le système. Ceci correspond au deuxième point rouge.

La pièce 2 est générée à la date 2 et sort à la date 12 ce qui donne un temps de séjour de 10 unités de temps. C’est le point numéro 3 de la courbe.



La pièce 3 est générée à la date 3 mais ne rentrera dans la file d’attente qu’a la date 4. Toutefois, sa date d’entrée dans le système est de valeur 3. Elle sort à la date 16 et possède donc un temps de séjour de valeur 13.



Ce qui se passe ensuite est plus étrange. Visiblement, voilà ce que fait le moteur de simulation….

La pièce 4 devrait normalement être disponible dans la source à la date t=4 et bien il n’en n’est rien. Le moteur de simulation gérant les événements dans un ordre « incorrect » passe alors par un état ou aucune pièce n’est disponible sur la source et la pièce 5 apparaît sur la source à la date t=5.



Le pièce numéro 4 entre donc à la date t=5 et quitte le système à la date t=20, ce qui représente un temps de séjour de 15 unités de temps.

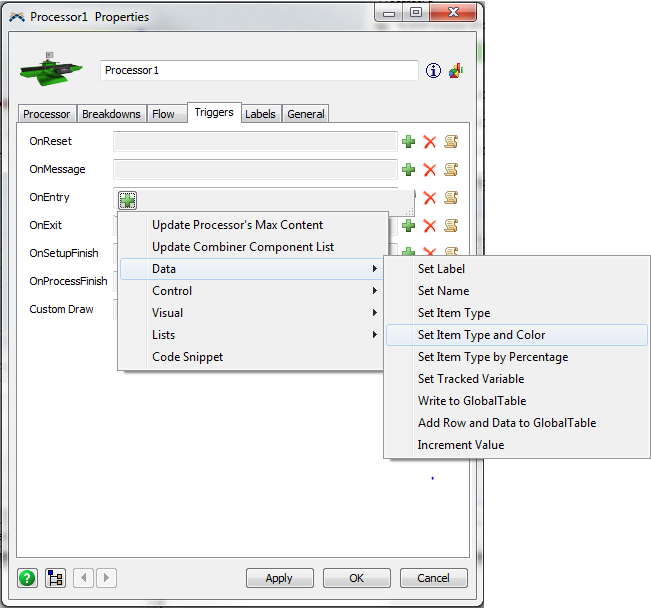
Un autre moyen de visualiser la situation est de représenter les différents états du système.

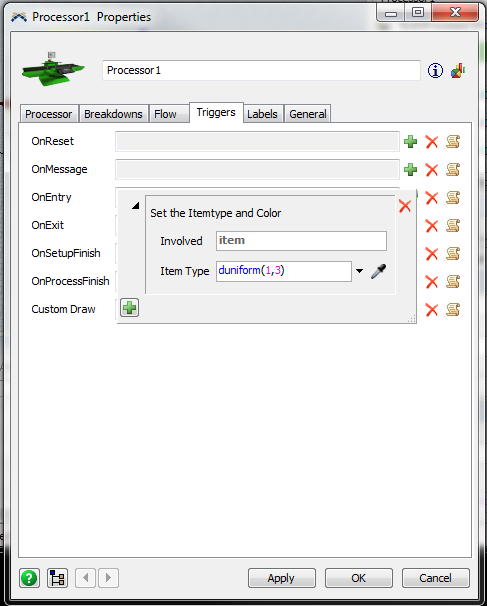


Le même phénomène se produit à la date t=8, expliquant ainsi la durée de séjour de la pièce P5 qui lui aussi sera de valeur 15.

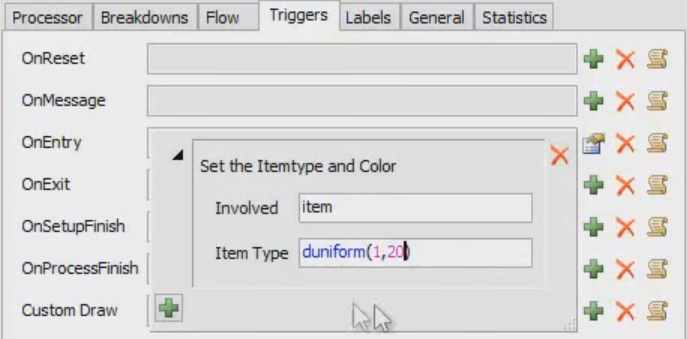
**Configuration de la machine pour visualiser les pièces en mouvement**

Il faut faire un double clic sur le processeur pour obtenir l’ensemble des **properties**. L’action à ajouter concerne l’événement **OnEntry** et utiliser  pour sélectionner dans le menu contextuel le menu **Data** et ensuite **Set item Type and Color**.

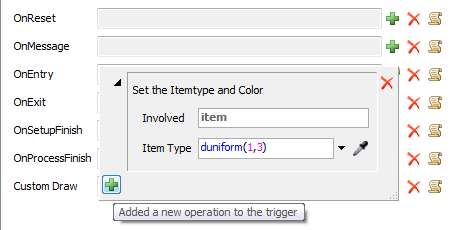




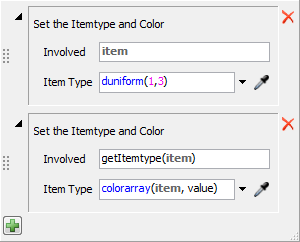
Un peut alors choisir aléatoirement un type pour l’item en lui affectant une valeur aléatoire entre 1 et 20.



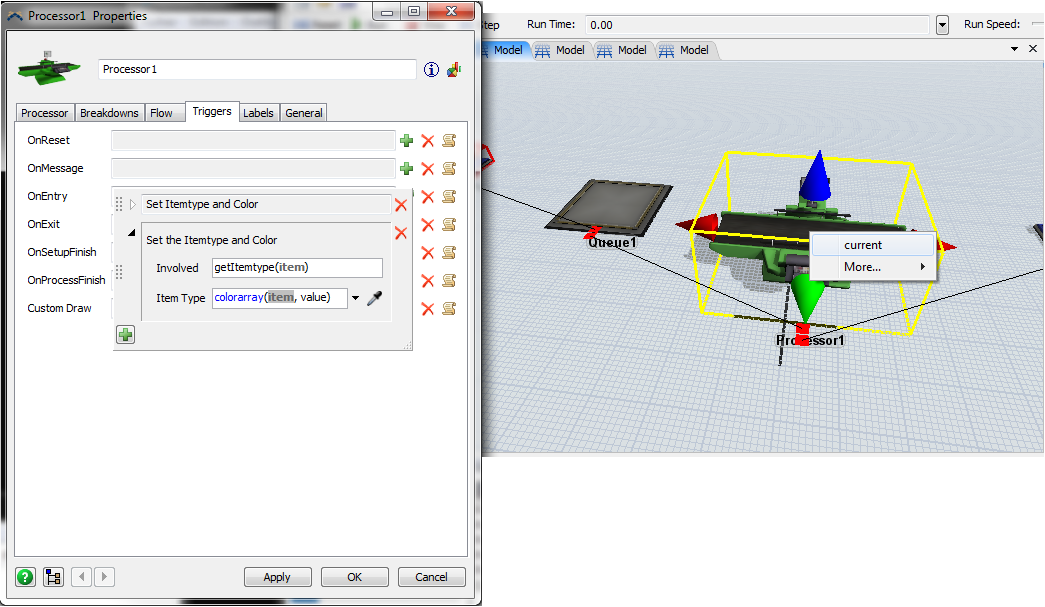
En cliquant sur le , on peut ajouter d’autres actions à cet événement.



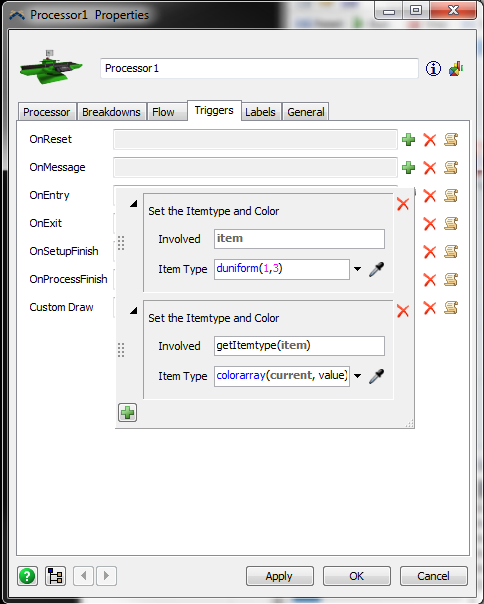
Il faut mettre à jour la couleur en fonction du type d’item.



Un moyen très simple de trouver le « bon » attribut est de sélectionner à la souris **item** et ensuite d’utiliser le , et en montrant ainsi à la souris un objet du modèle (ici le processeur) on a accès à l’ensemble des attributs « compatibles ».



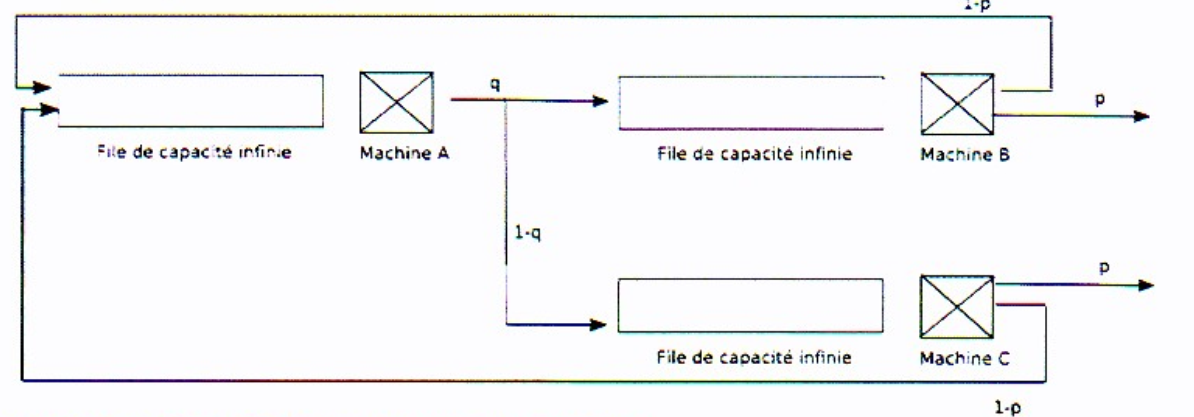
La configuration correcte est la suivante :



**3) Simuler un système avec routage probabiliste**

La durée inter-arrivée ainsi que les temps de traitement suivent une loi constante de paramètre :

* **Lam** pour la durée inter-arrivée
* **Sa** pour la durée de traitement sur la machine A
* **Sb** pour la durée de traitement sur la machine B
* **Sc** pour la durée de traitement sur la machine C
* Les paramètres **p** et **q** sont des paramètres de transition.



Il s’agit de réaliser un modèle de simulation Simio en prenant par exemple :

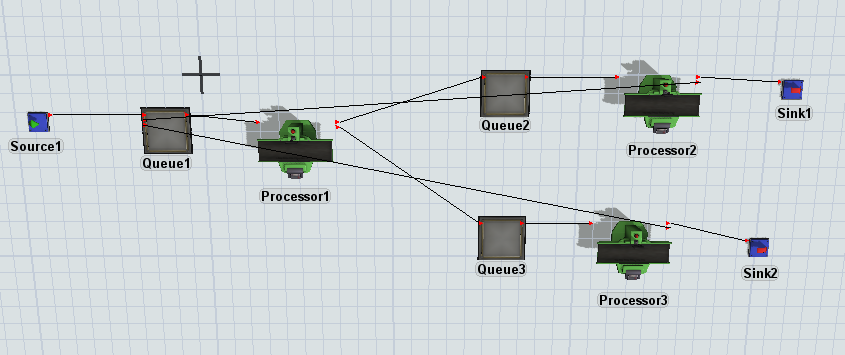
Sa=1, Sb=0.8, Sc=0.6, p=0.6, q=0.4 et Lam=10.

**3.1) Réalisation du modèle**

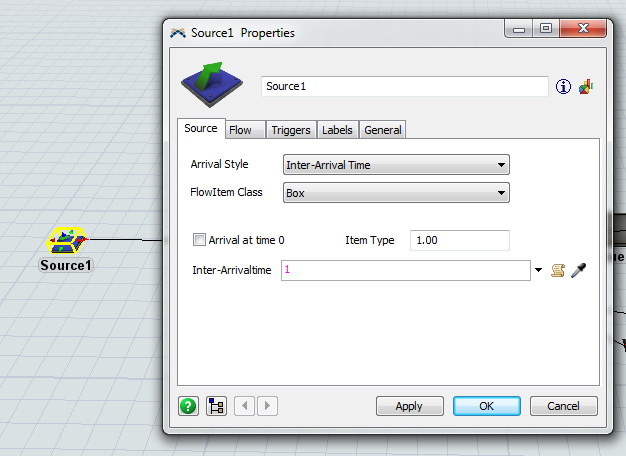
En utilisant 1 source, 3 serveurs et 1 puits, on peut construire le réseau.



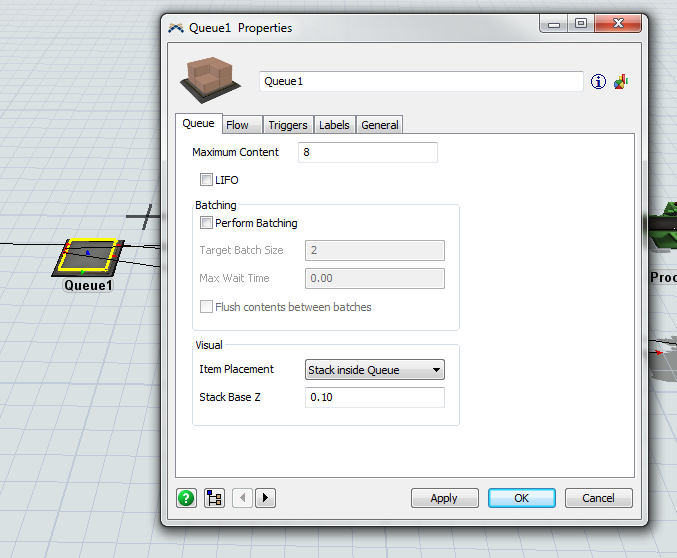
Il faut ensuite relier les entités. La machine nommée Processor2 est reliée à la fois aux puits « Sink1 » et à la file 1 nommée « Queue1 ». Il faut procéder de la même manière avec le Processor3.

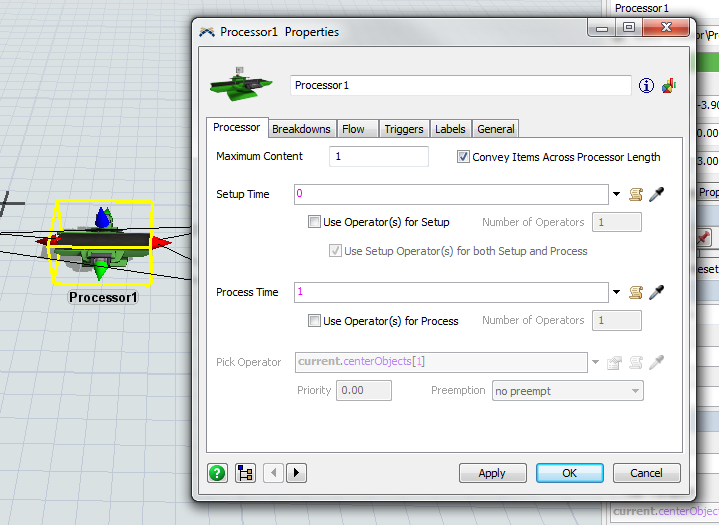


Pour la source, on simule l’arrivée d’une pièce toute les 1 unité de temps.

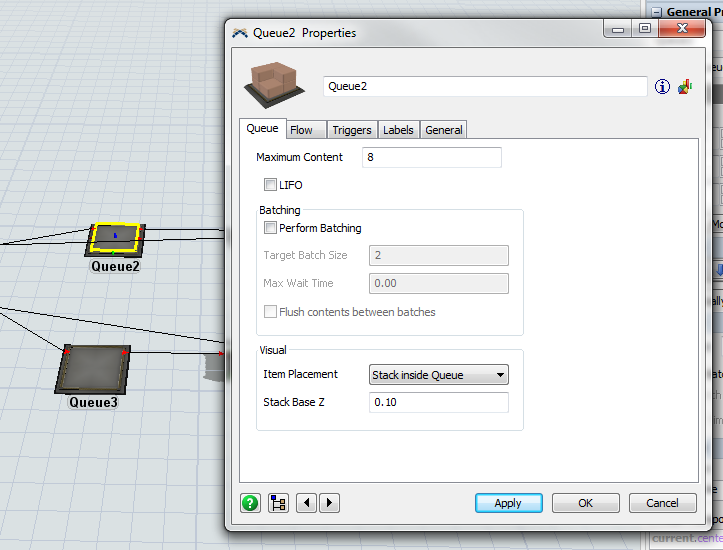


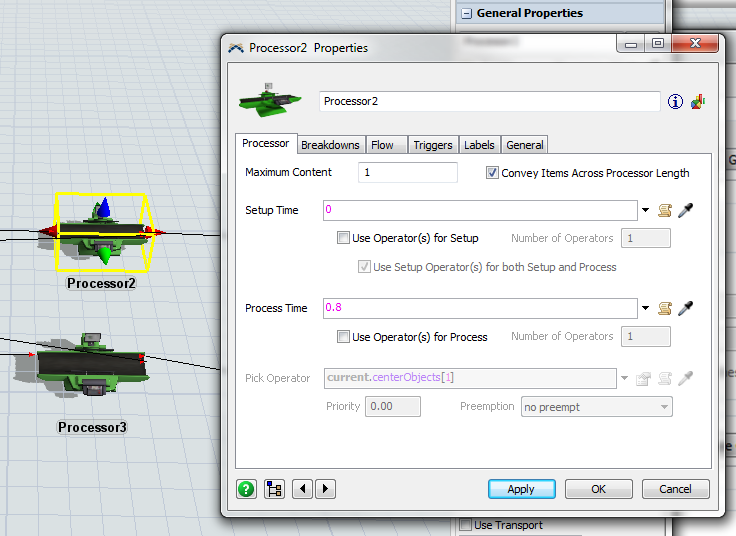
La file du serveur 1 est dimensionnée avec 8 places et le serveur 1 (ici Processor1) a un temps de traitement de 1 unité.



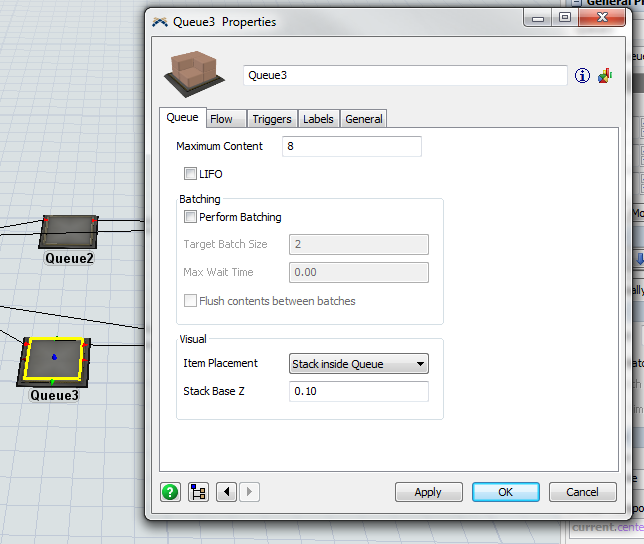


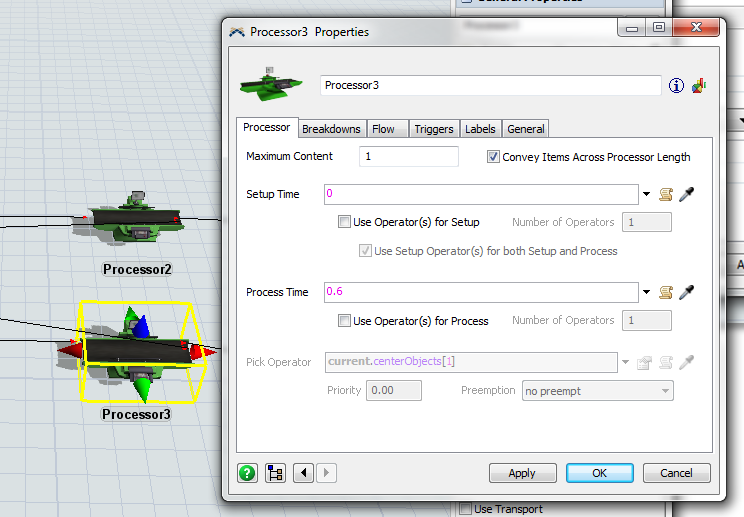
Il faut procéder de manière similaire avec le Queue2 et le processor2.





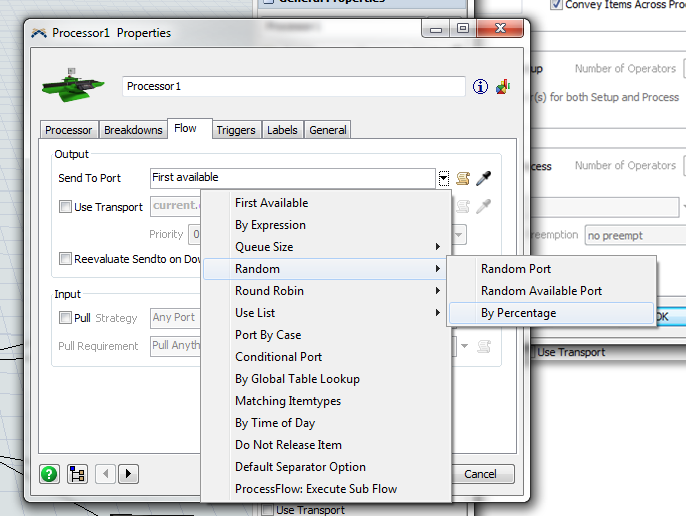
Il faut procéder de manière similaire avec le Queue3 et le processor3.





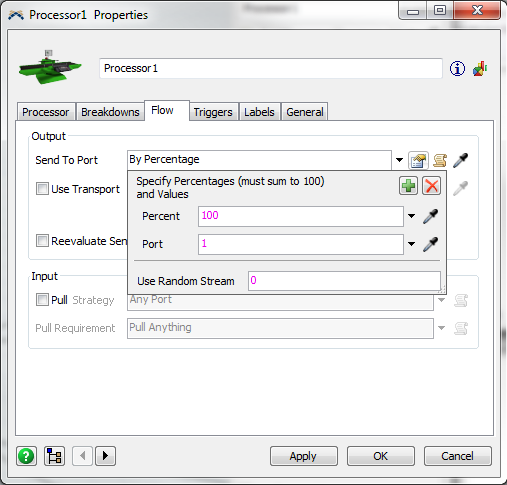
**Routage probabilité au niveau du Processor1**

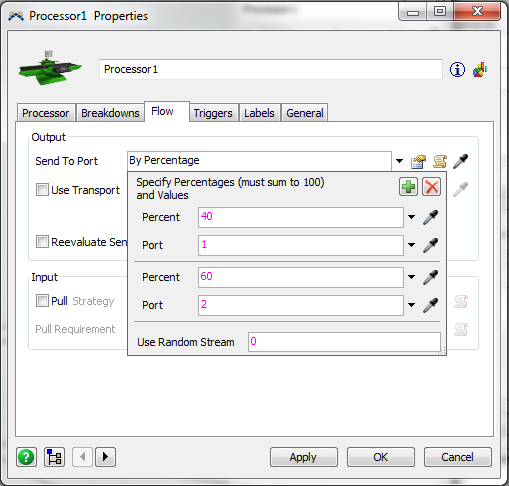
On peut commencer par le Processor1 et choisir l’onglet **Flow** puis **setToPort**.



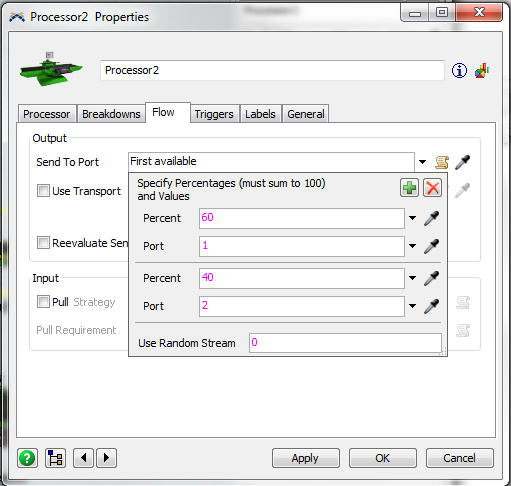
Il faut sélectionner la règle **Random** et **By** **Percentage**.

L’onglet permet d’accéder à la saisie des pourcentages.

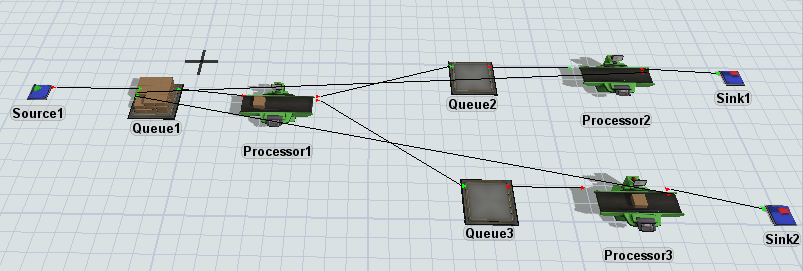




Il faut réitérer l’opération pour le Processor2 et le Processor 3.



Avec une file de capacité limité sur la Queue1, le système se bloque assez rapidement car une pièce du Processor2 ou 3 demande à être routée vers la Queue1 alors que la file est pleine… mais en augmentant la taille de la Queue1, on peut plus facilement visualiser la dynamique du système….



Une fois l’exécution lancée et arrêtée, on peut obtenir un rapport statistique sur le système via le menu Statistiques.

On peut alors générer un fichier Excel et retravailler la présentation en fonction des besoins.

