

### Simulation 1 : vérification de la conformité de la représentation

Actions nécessaires pour la simulation :

1) ouvrir le diagramme à simuler

Exemple 1

2) cliquer sur RUN

The screenshot shows a SysML activity diagram titled 'act [Activité] Model[ Model ]'. The diagram consists of a start node, followed by activity node 'A', then activity node 'B', and finally an end node. A 'Run' button is visible in the top right corner of the diagram editor.

diminuer la vitesse d'exécution

The screenshot shows the simulation control interface. It includes a 'Vitesse d'Animation' (Animation Speed) slider, a 'Trigger' dropdown, and a 'Console' window displaying log messages: '00:00:00,000 : \*\*\*\* Activité Model is initialized. \*\*\*\*'. A 'Variables' table is also visible.

cliquer sur Exécution

Lancer Exécution ( F8 )

The screenshot shows a close-up of the simulation toolbar. A callout points to the 'Exécution' (Execution) button, which is labeled 'Lancer Exécution ( F8 )'. Below the toolbar, the model name 'Model [Model@73c44481] (Ready)' is displayed.

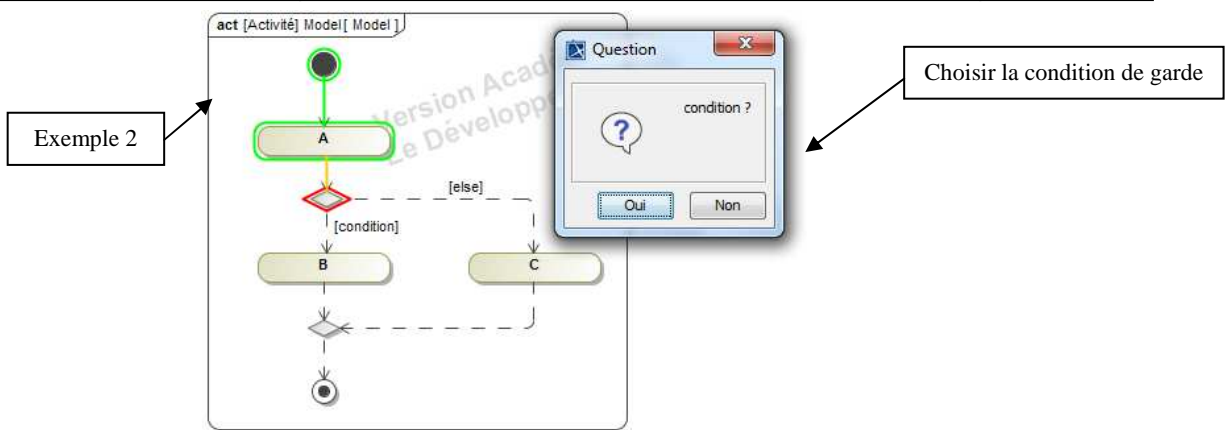
visités (vert)

dernier visité (orange)

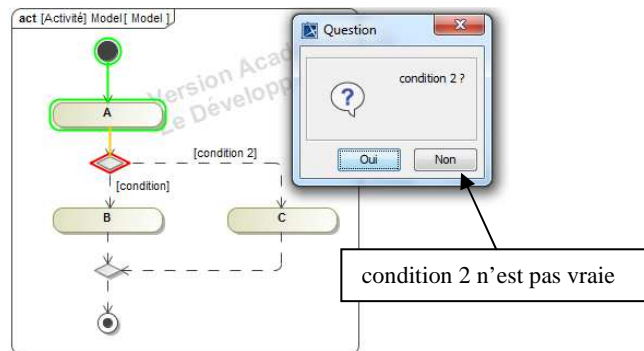
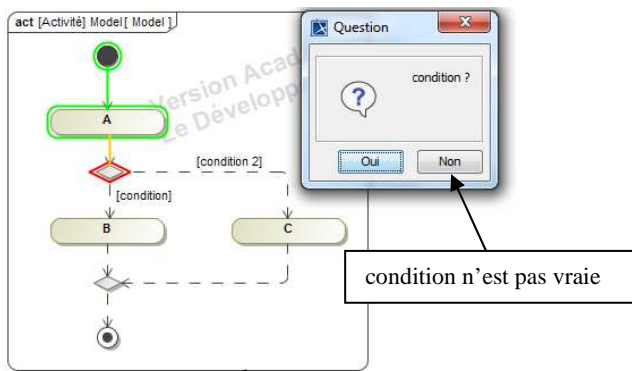
en cours (rouge)

The screenshot shows the simulation in progress. The activity diagram is color-coded: node 'A' is green (visited), node 'B' is orange (last visited), and the flow between them is red (in progress). A callout points to the 'Pause' button in the simulation control panel.

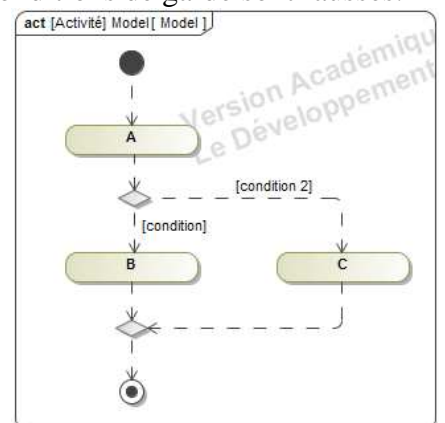
cliquer sur pause pour visualiser le comportement réalisé



Comportement interrompu :



Arrêt immédiat de la simulation lorsque les 2 conditions de garde sont fausses.



Exercice à simuler : sujet E4 2017 simulation 1.mdzip

Conclusion :

Le comportement n'est pas correct lorsqu'il y a des nœuds de décision car le comportement est interrompu.

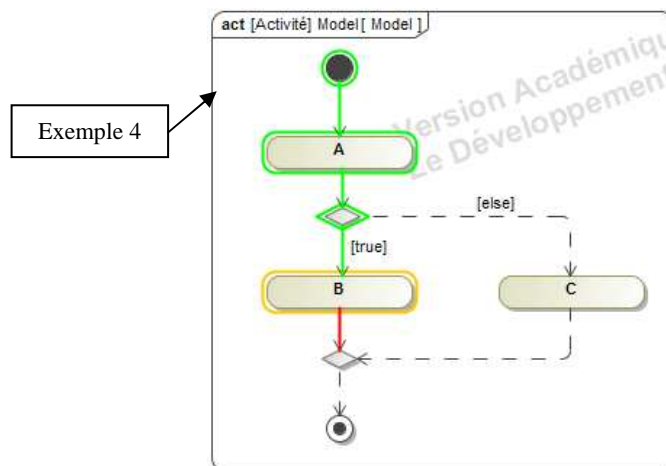
Le comportement est interrompu aussi lorsqu'il y a une erreur de représentation.

**Simulation 2 : obtenir un comportement correct** (non interrompu par un nœud de décision)

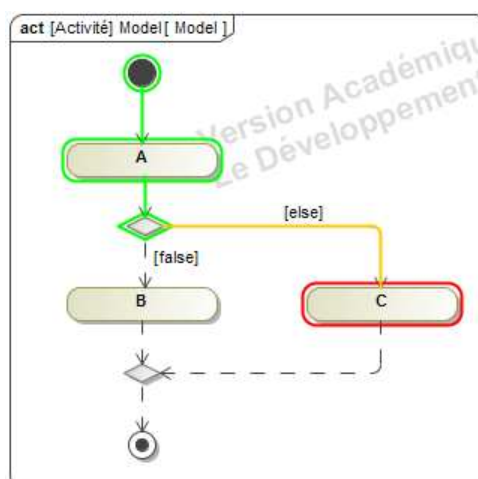
Actions nécessaires pour la simulation :

Solution 1 : **remplacer la condition de garde par le résultat attendu**

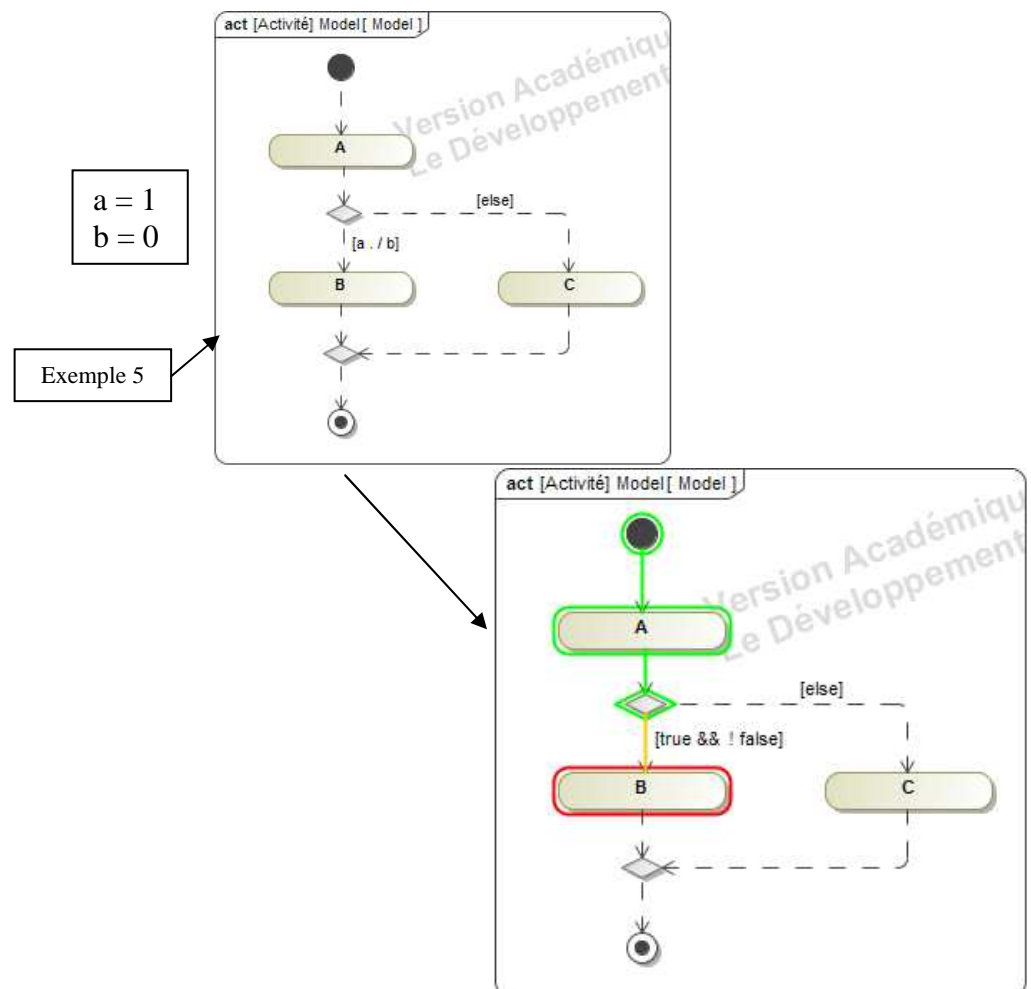
Cas 1 : La condition de garde *condition* est vraie (**true**) avant l'exécution de l'activité.



Cas 2 : La condition de garde *condition* est fausse (**false**) avant l'exécution de l'activité.



Solution 2 : remplacer la condition de garde par la proposition logique (états logiques et opérateurs)



Les résultats logiques :

vrai (1) : **true**

faux (0) : **false**

Les opérateurs logiques :

Non : **!**

ET : **&&**

OU : **||**

Les opérateurs relationnels :

égalité : **==**

différent : **!=**

inférieur : **<**

inférieur ou égal : **<=**

supérieur : **>**

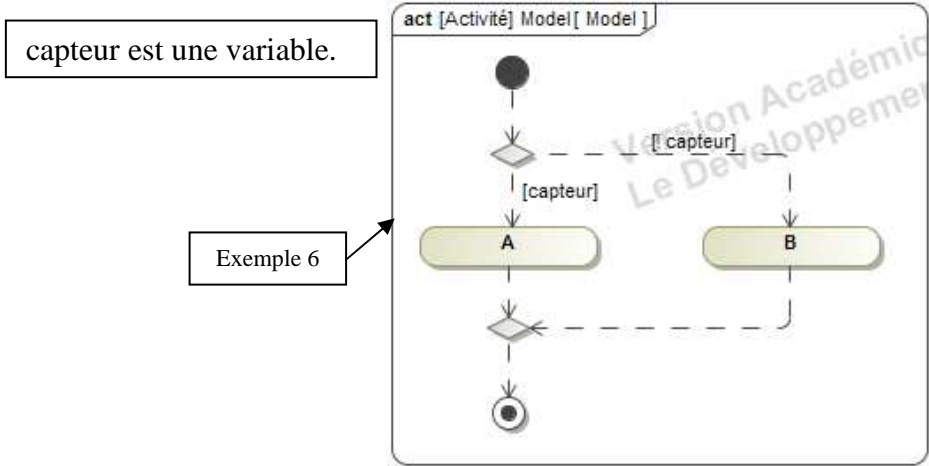
supérieur ou égal : **>=**

Exercices à simuler : **TP01 EX5 simulation 2.mdzip** et **TP6 simulation 2.mdzip**

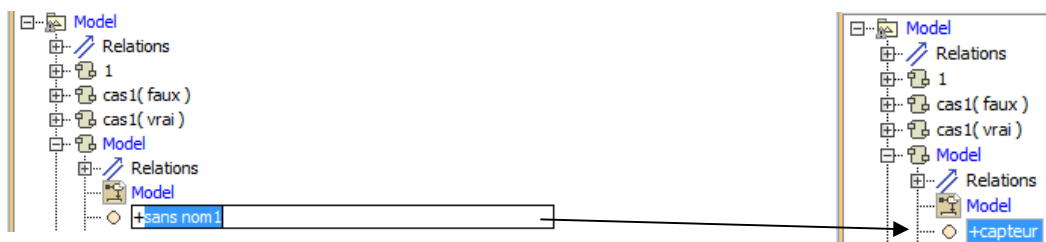
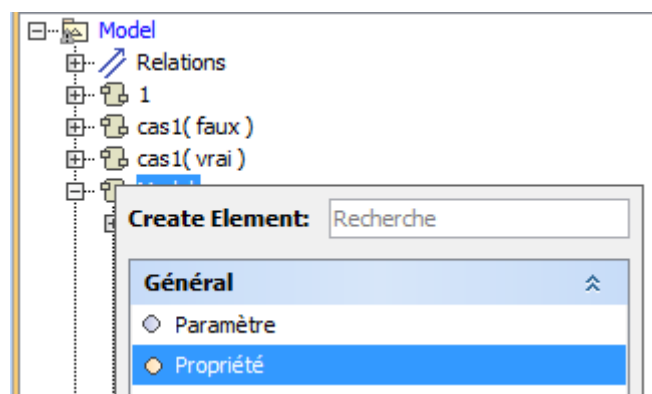
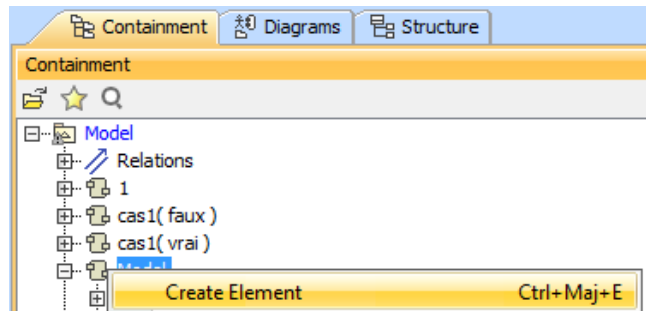
### Simulation 3 : utilisation de variables

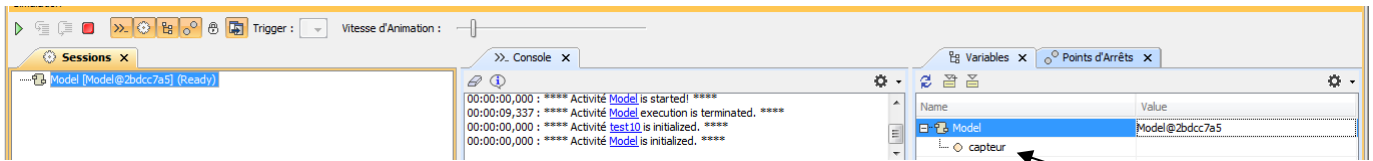
Actions nécessaires pour la simulation :

- utilisation de la console pour déclarer et initialiser les variables

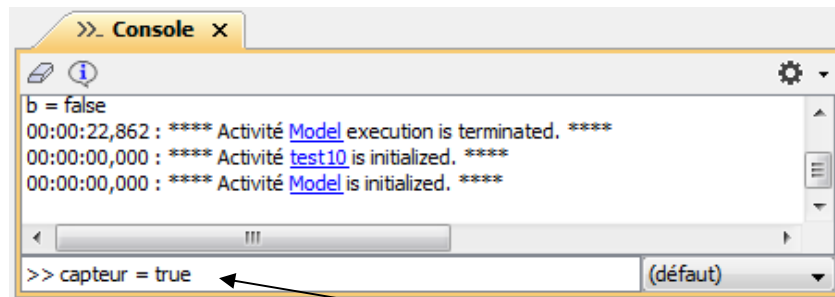


### Création de la variable :

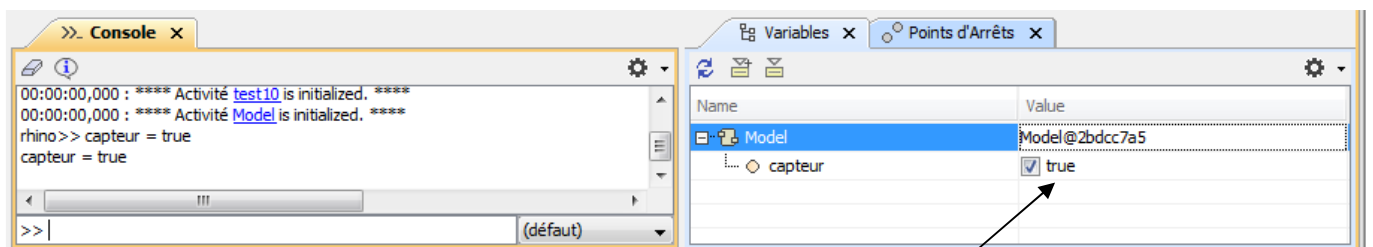




la variable capteur est déclarée

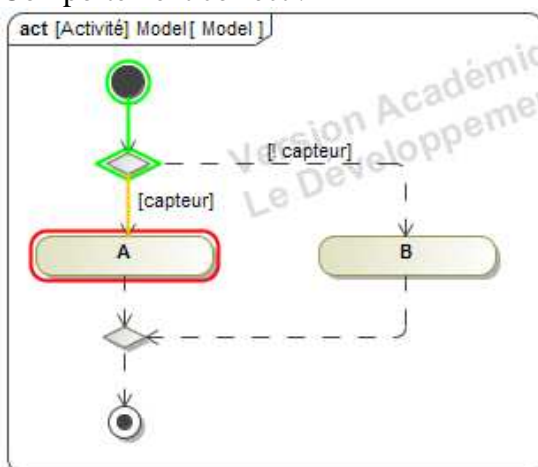


Affectation de la variable capteur d'une valeur à l'aide de la console



Visualisation de la valeur

Comportement correct :

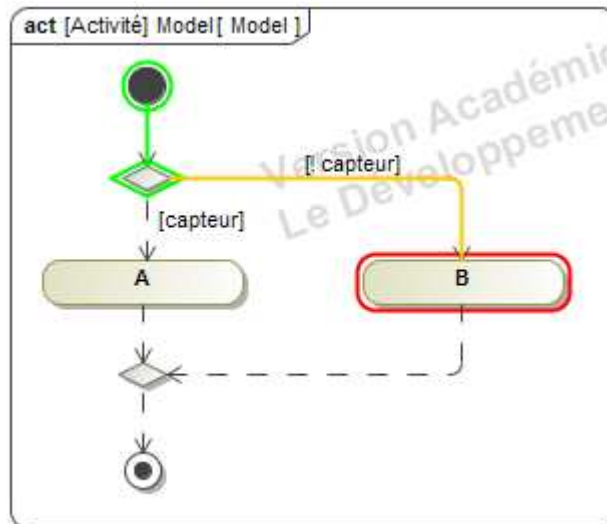


- utilisation de la fenêtre variables pour modifier la valeur d'une variable

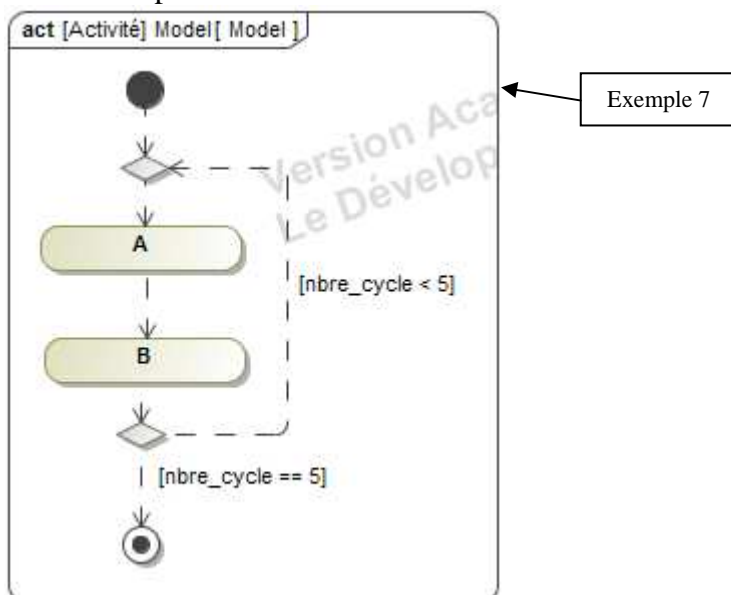
Modification de l'état de la variable capteur :

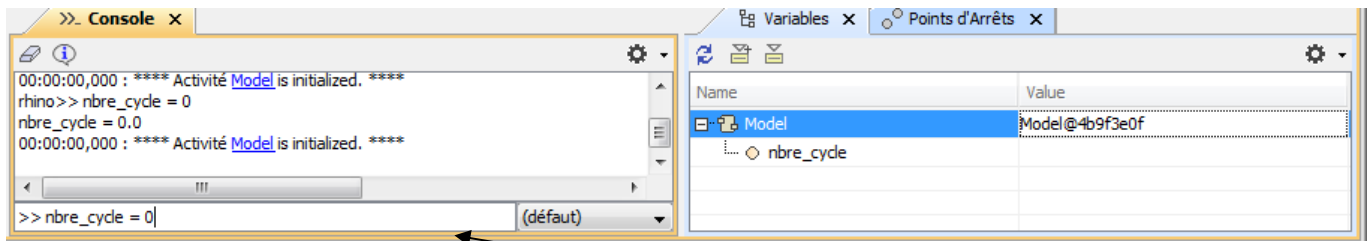
Name	Value
Model	Model@11111a53
capteur	<input type="checkbox"/> false

Comportement correct :

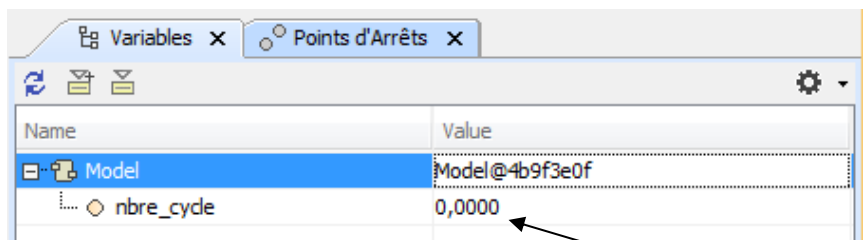


Autre exemple :



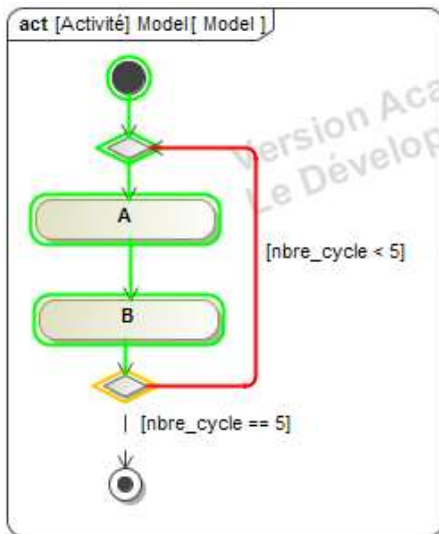


Affectation de la variable **nbre\_cycle** de la valeur **0** à l'aide de la console

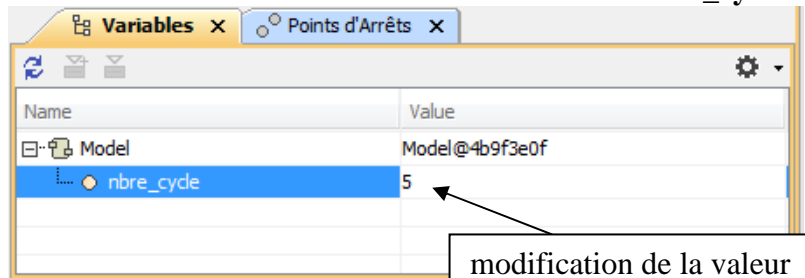


Visualisation de la valeur

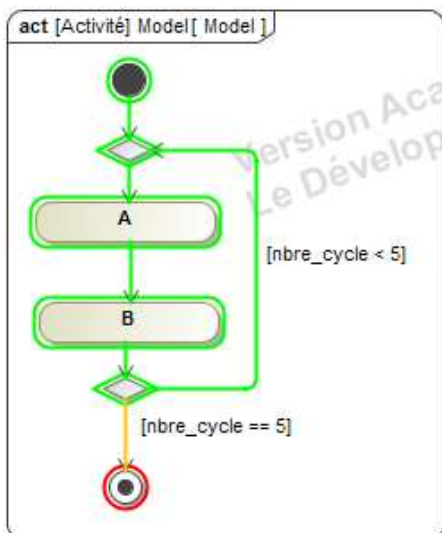
Comportement correct :



Modification de la valeur de la variable **nbre\_cycle** :



modification de la valeur



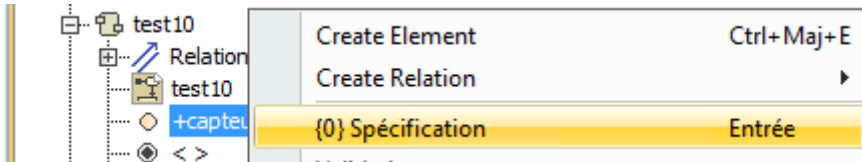
Comportement correct :



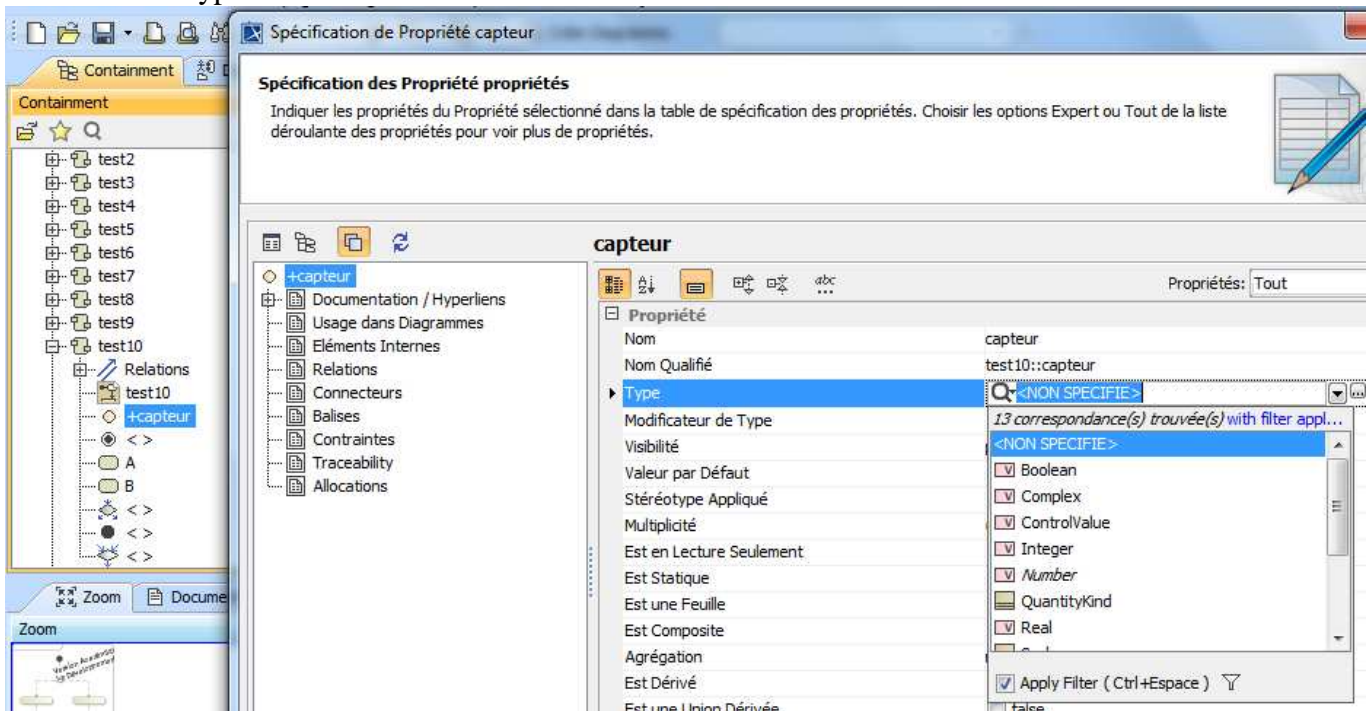


Exercices à simuler : TP01 EX5 simulation 3.mdzip et TP6 simulation 3.mdzip

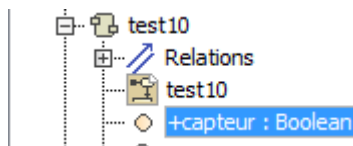
**Simulation 4 : utilisation de variables avec un type**



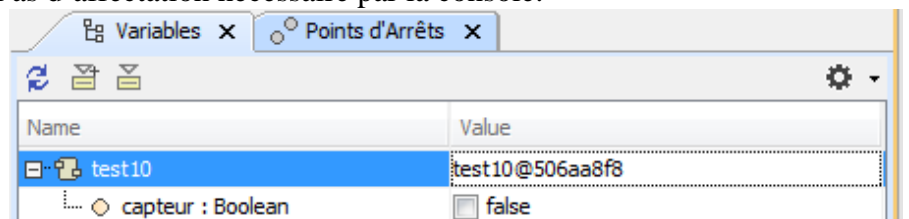
Les différents types :



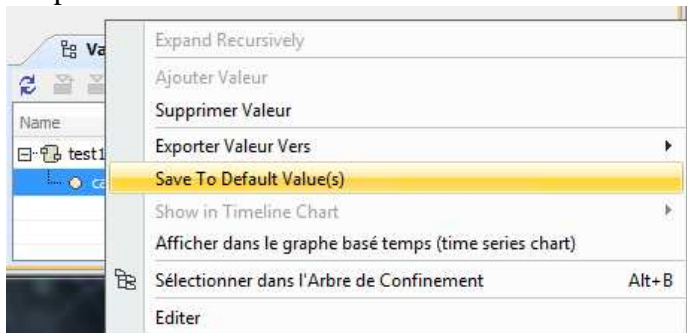
La variable est de type Booléen :



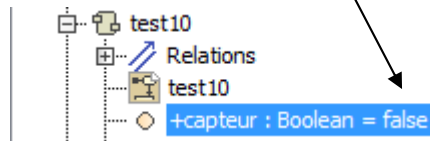
Pas d'affectation nécessaire par la console:



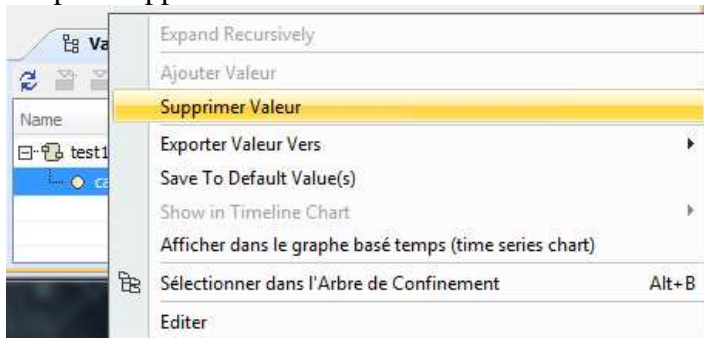
On peut sauver la valeur :



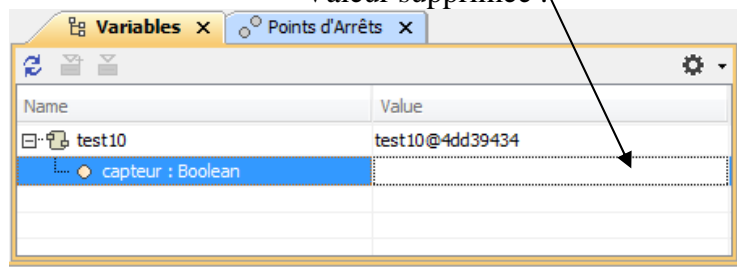
Valeur sauvegardée :



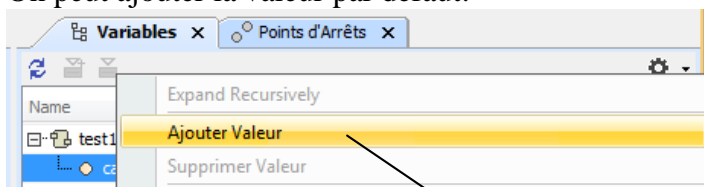
On peut supprimer la valeur :



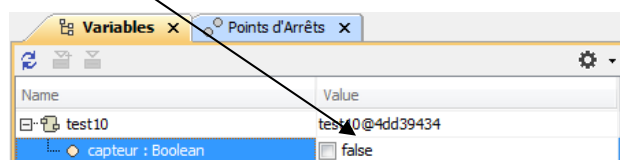
Valeur supprimée :



On peut ajouter la valeur par défaut:



Valeur ajoutée :



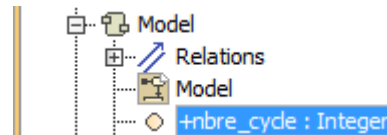
Comportement correct :

act [Activité] test10 [test10]

Variables x Points d'Arrêts x

Name	Value
test10	test10@4dd39434
capteur : Boolean	<input checked="" type="checkbox"/> true

La variable est de type Entier :



Valeur par défaut avant simulation :

act [Activité] Model [ Model ]

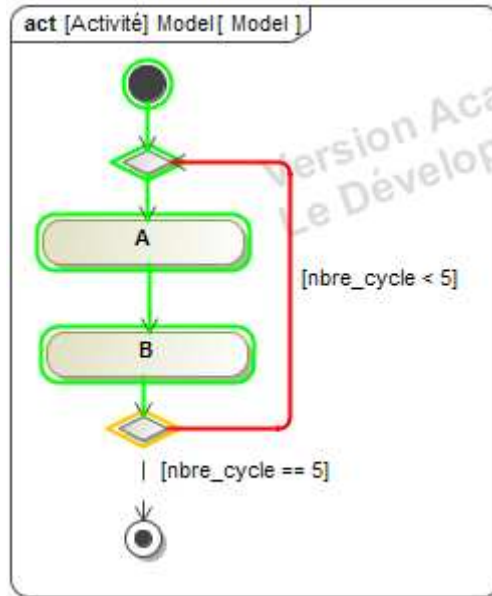
Simulation

Console x

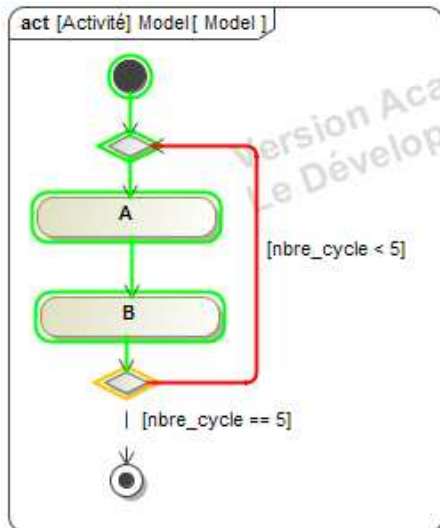
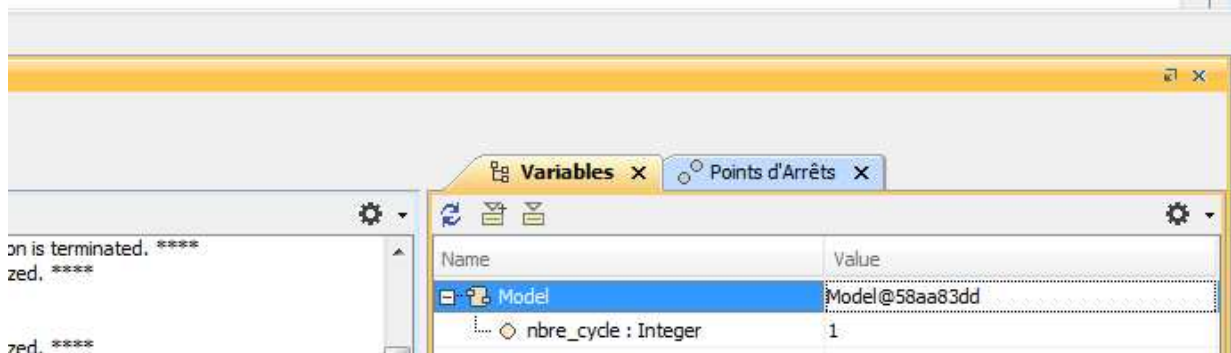
```
00:00:08,284 : **** Activité Model execution is terminated, ****
00:00:00,000 : **** Activité Model is initialized, ****
rhino>> nombre_cycle = 0
nombre_cycle = 0,0
00:00:00,000 : **** Activité Model is initialized, ****
```

Name	Value
Model	Model@'
nombre_cycle : Integer	0

Comportement correct :

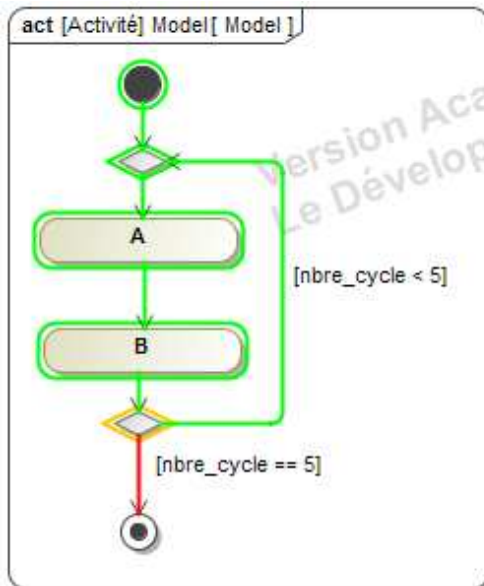


Valeur modifiée pendant la simulation :

Name	Value
Model	Model@58aa83dd
nbre_cycle : Integer	1

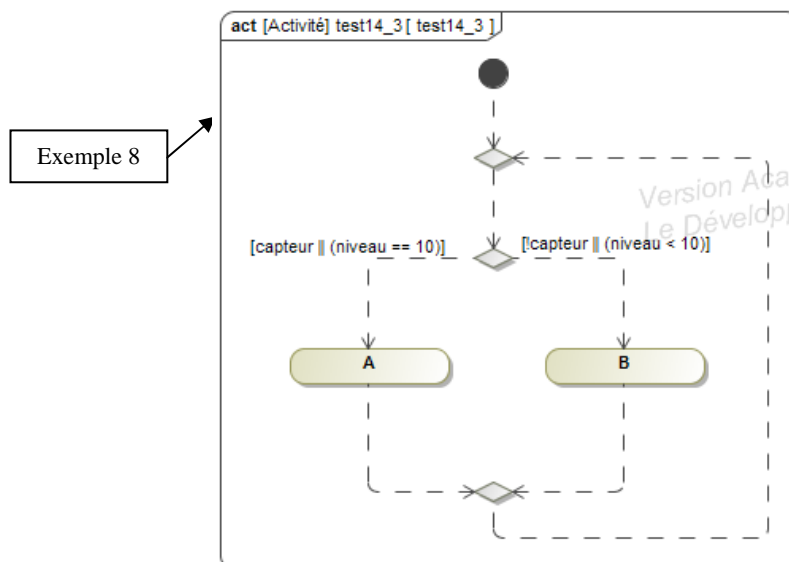
Valeur modifiée une deuxième fois pendant la simulation :



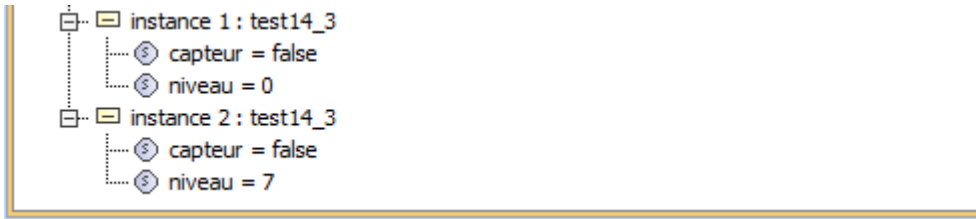
Name	Value
Model	Model@58aa83dd
nbre_cycle : Integer	5

Exercices à simuler : TP01 EX5 simulation 4.mdzip et TP03 EX2 simulation4.mdzip

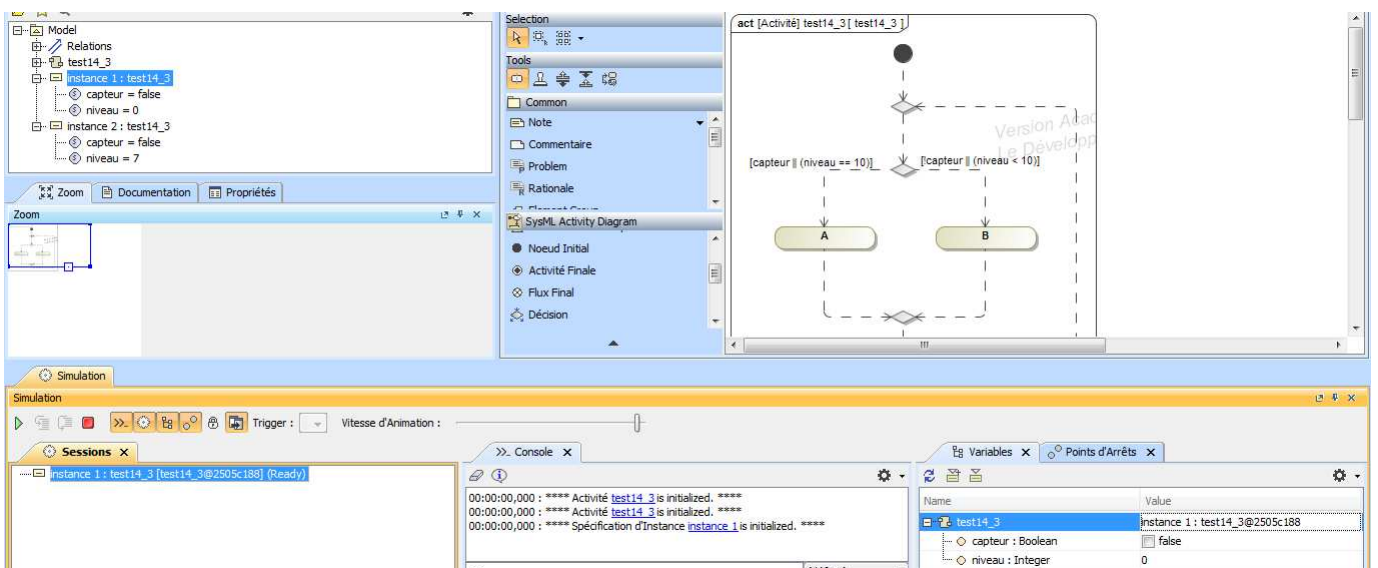
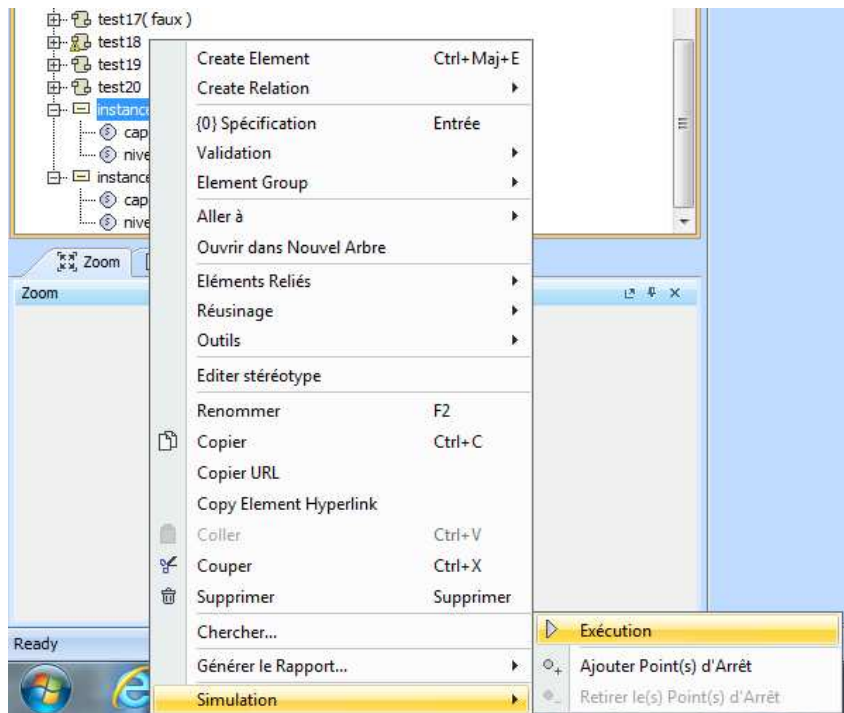
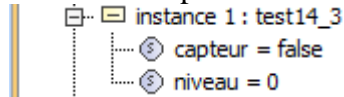
**Simulation 5 : utilisation d'instances contenant les différentes configurations**



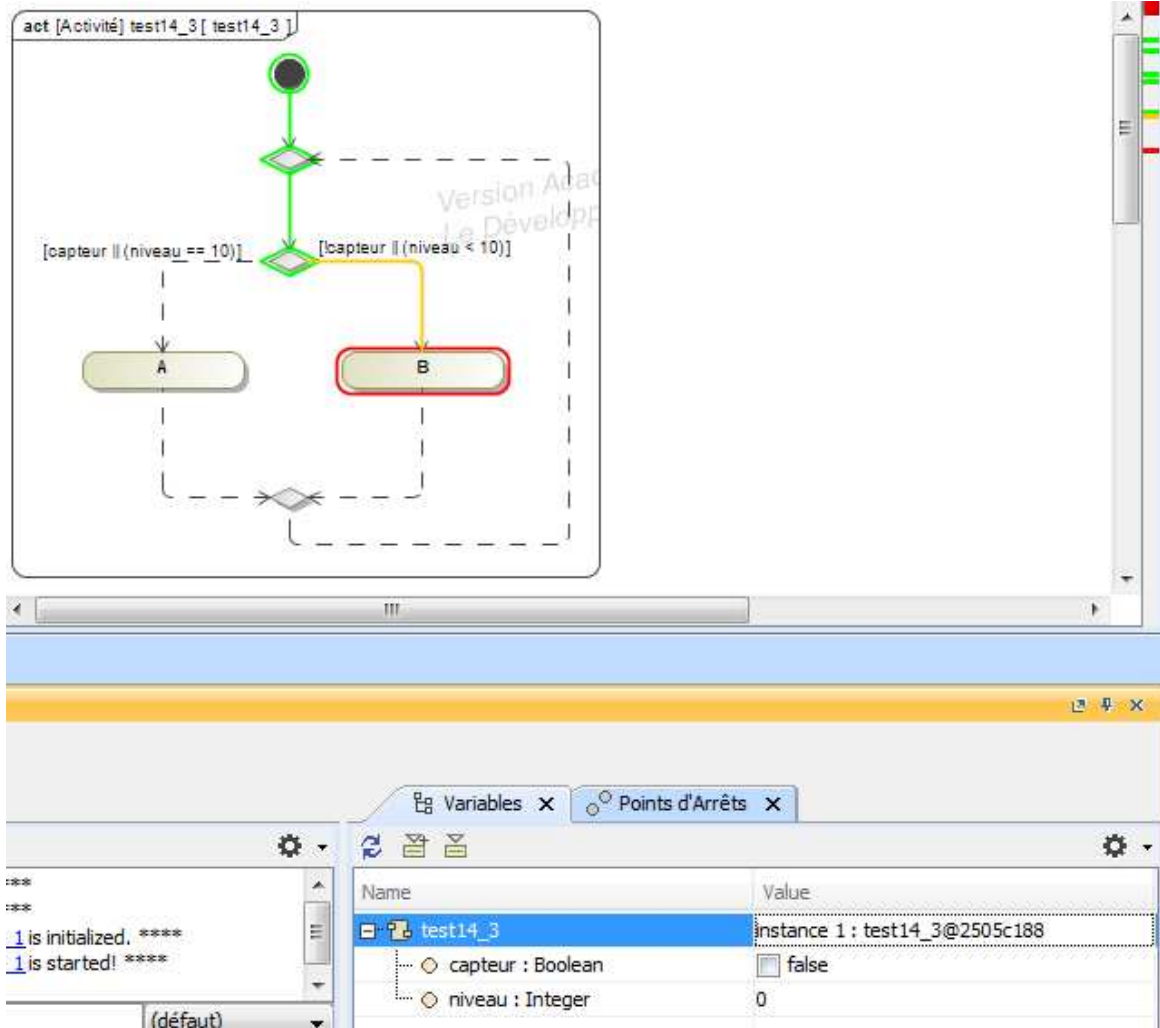
Les 2 instances créées :



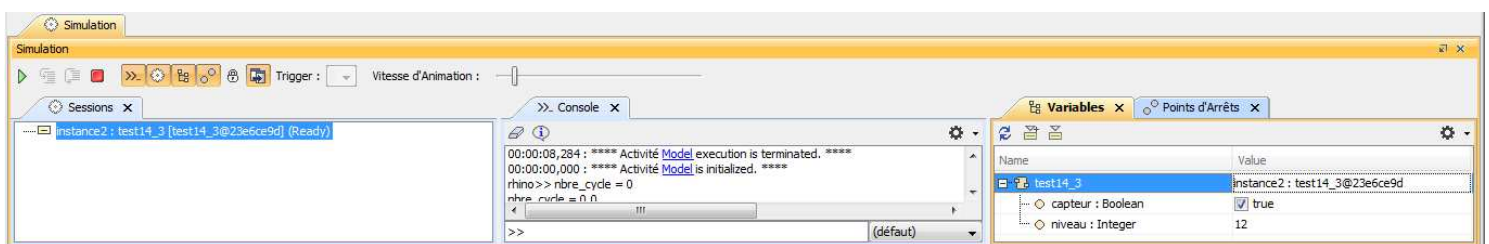
Simulation à partir d'une instance :



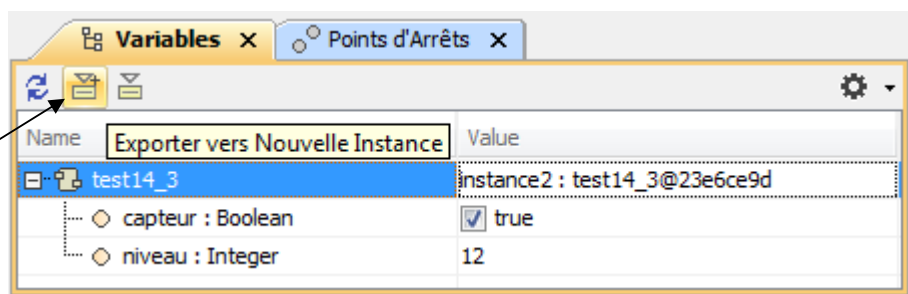
Comportement correct :

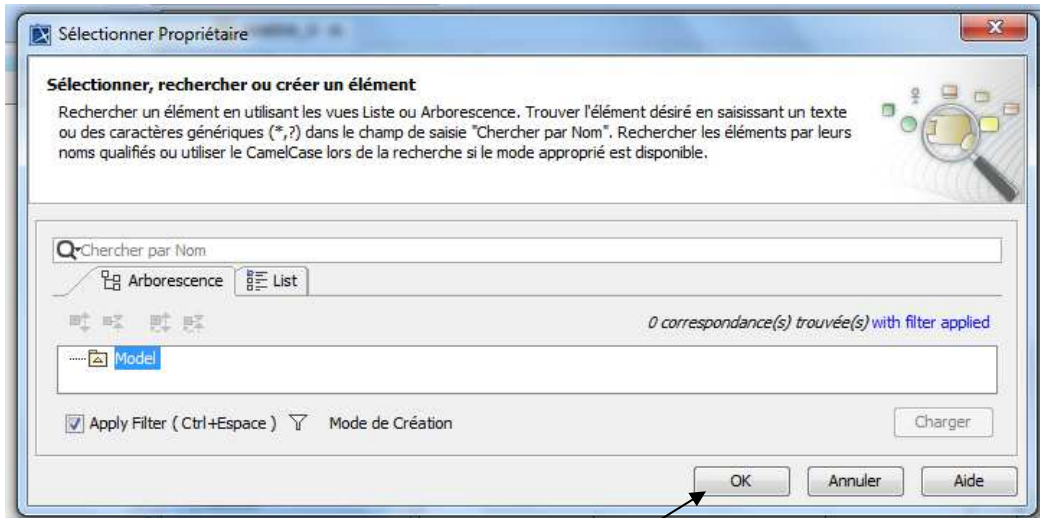


Créer une nouvelle instance :



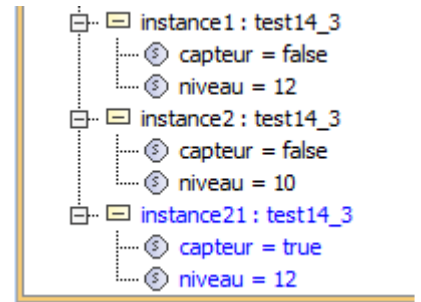
Cliquer sur



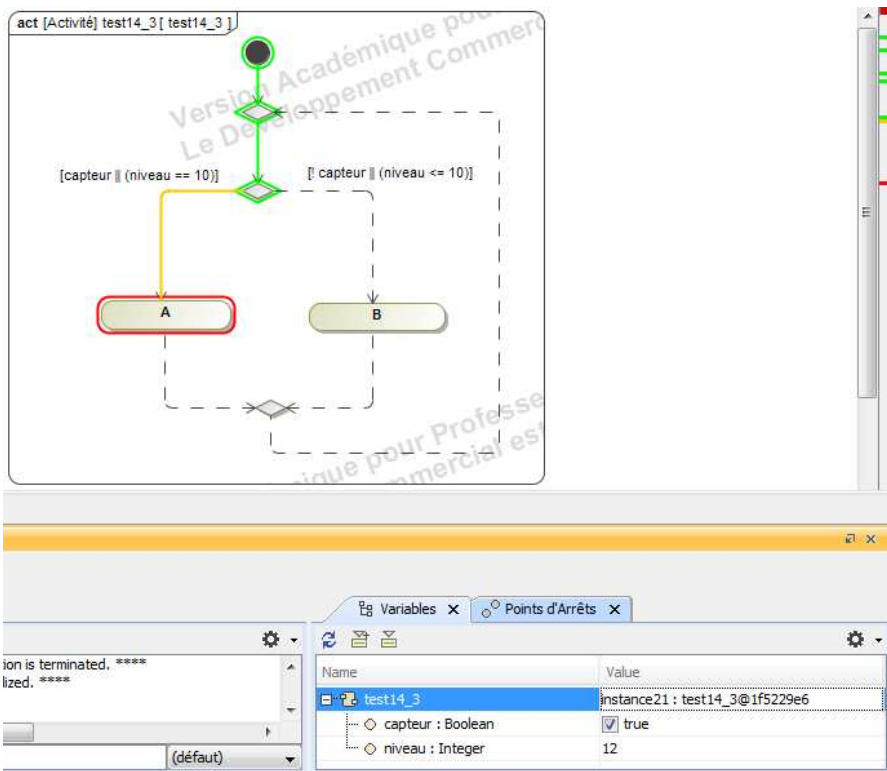


Cliquer sur

La nouvelle instance dans l'arborescence :



Comportement correct :

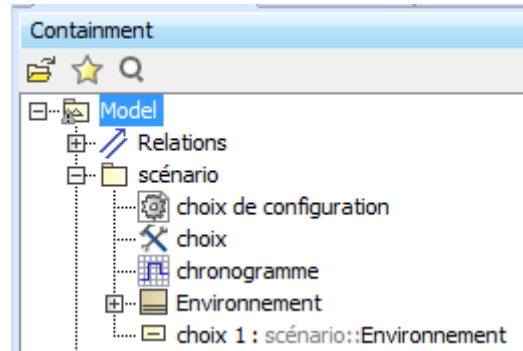




Exercice à simuler : TP01 EX5 simulation 5.mdzip

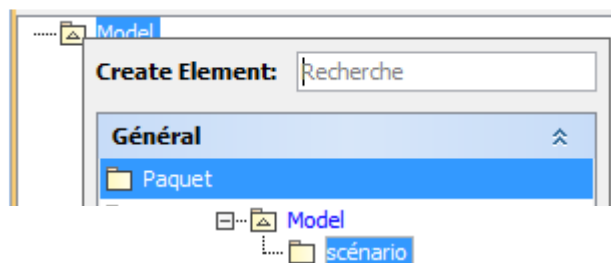
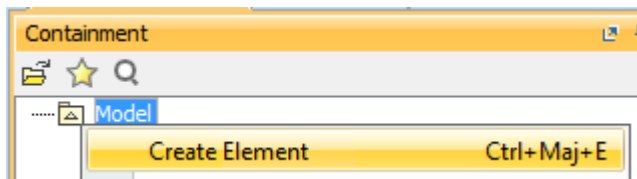
## Simulation 6 : utilisation de scénarios contenant les différentes configurations et une représentation temporelle

La structure du projet :

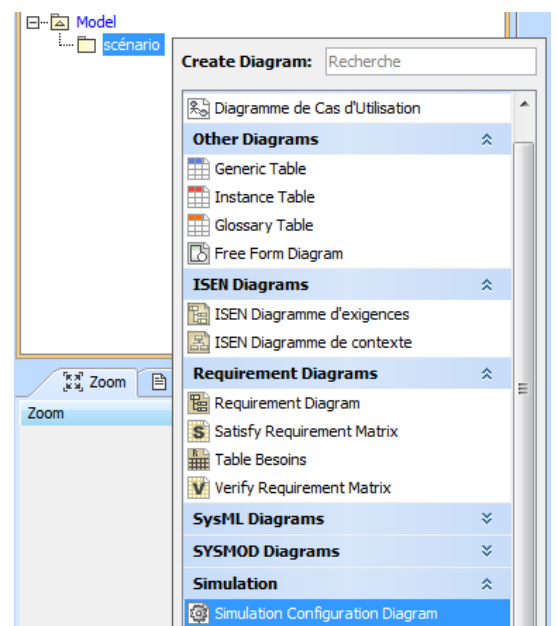


Développement du projet :

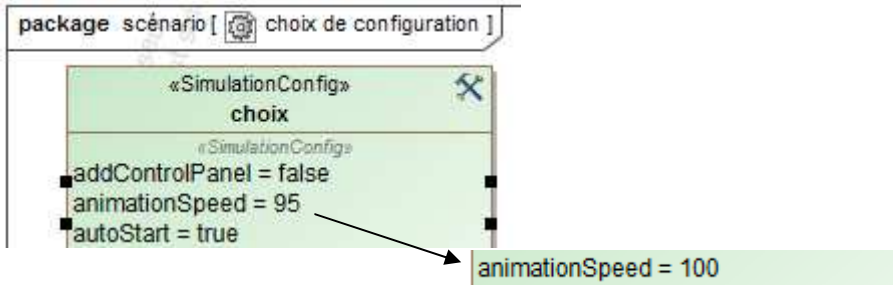
### 1) Création d'un paquet **scénario**



### 2) Création du diagramme de configuration de la simulation **choix de configuration**



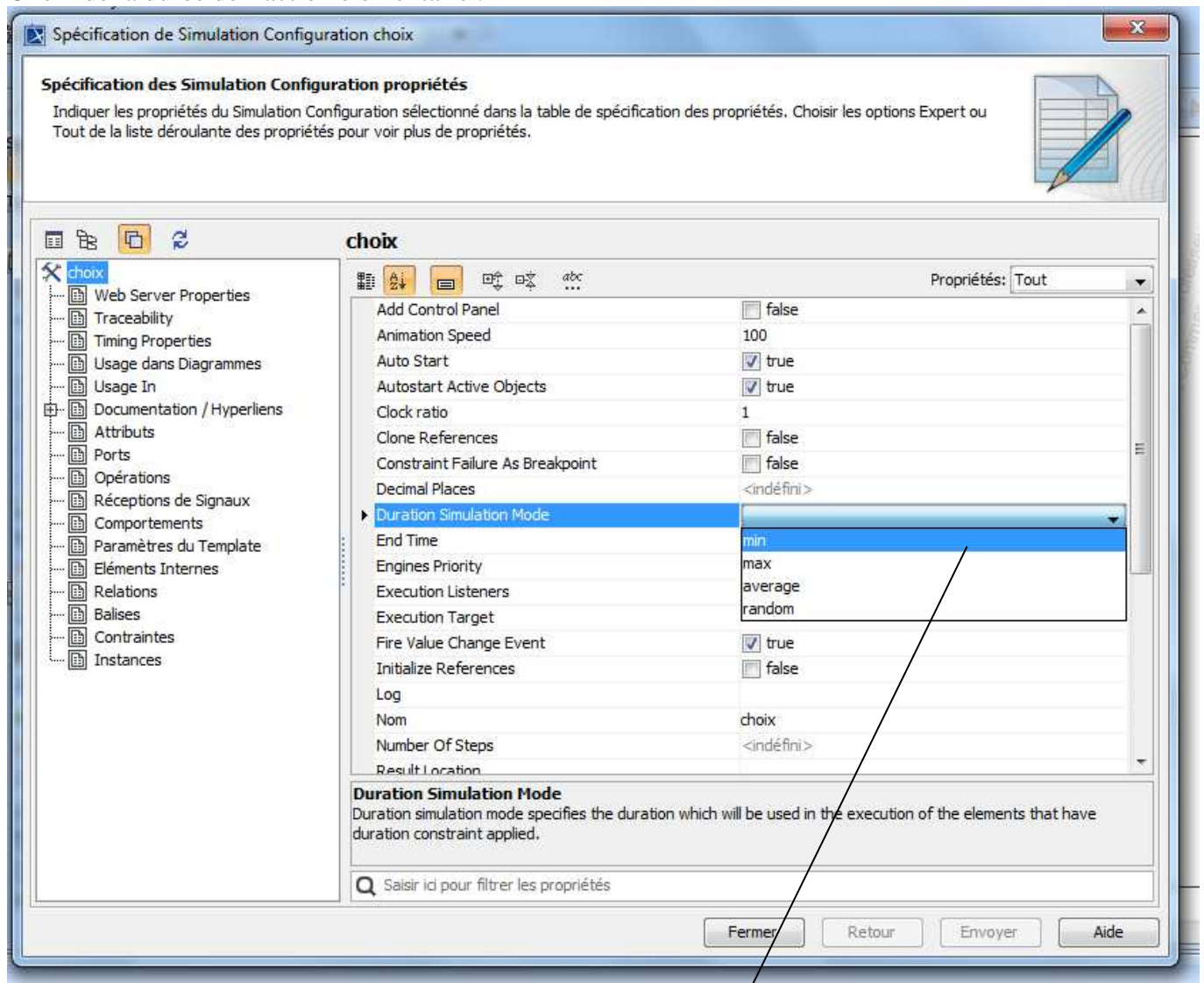
Modification de la **vitesse de l'animation à 100 %** :



Choix non exécutable actuellement :



Choix de la durée de l'action élémentaire :



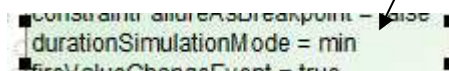
**Spécification des Simulation Configuration propriétés**  
Indiquer les propriétés du Simulation Configuration sélectionné dans la table de spécification des propriétés. Choisir les options Expert ou Tout de la liste déroulante des propriétés pour voir plus de propriétés.

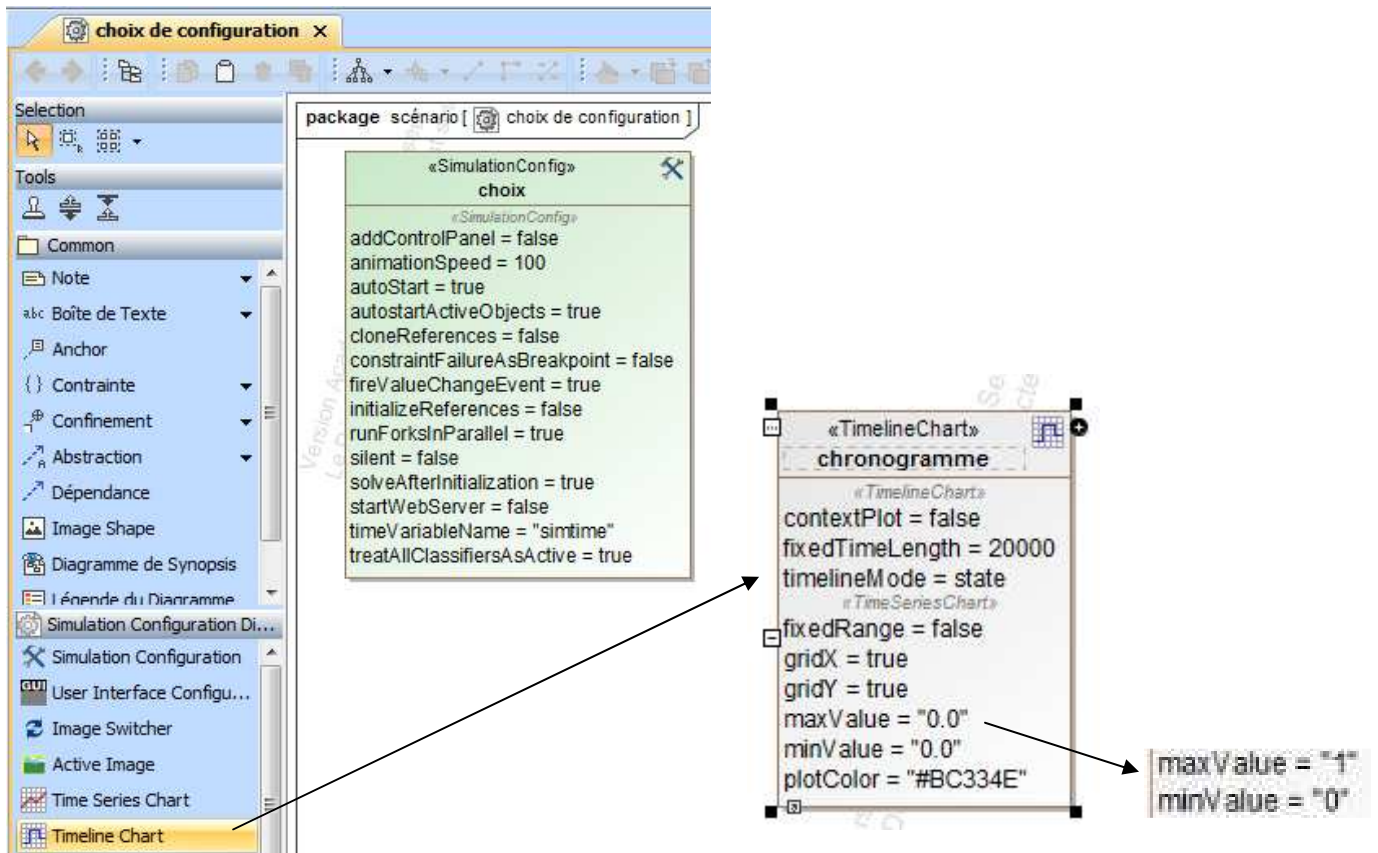
Propriété	Valeur
Add Control Panel	<input type="checkbox"/> false
Animation Speed	100
Auto Start	<input checked="" type="checkbox"/> true
Autostart Active Objects	<input checked="" type="checkbox"/> true
Clock ratio	1
Clone References	<input type="checkbox"/> false
Constraint Failure As Breakpoint	<input type="checkbox"/> false
Decimal Places	<indéfini>
<b>Duration Simulation Mode</b>	<b>min</b>
End Time	min
Engines Priority	max
Execution Listeners	average
Execution Target	random
Fire Value Change Event	<input checked="" type="checkbox"/> true
Initialize References	<input type="checkbox"/> false
Log	
Nom	choix
Number Of Steps	<indéfini>
Result Location	

**Duration Simulation Mode**  
Duration simulation mode specifies the duration which will be used in the execution of the elements that have duration constraint applied.

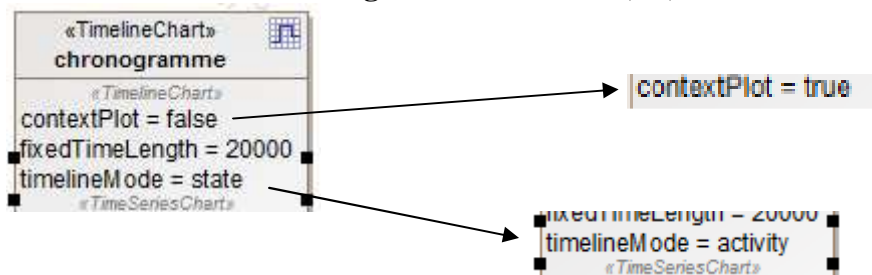
Saisir ici pour filtrer les propriétés

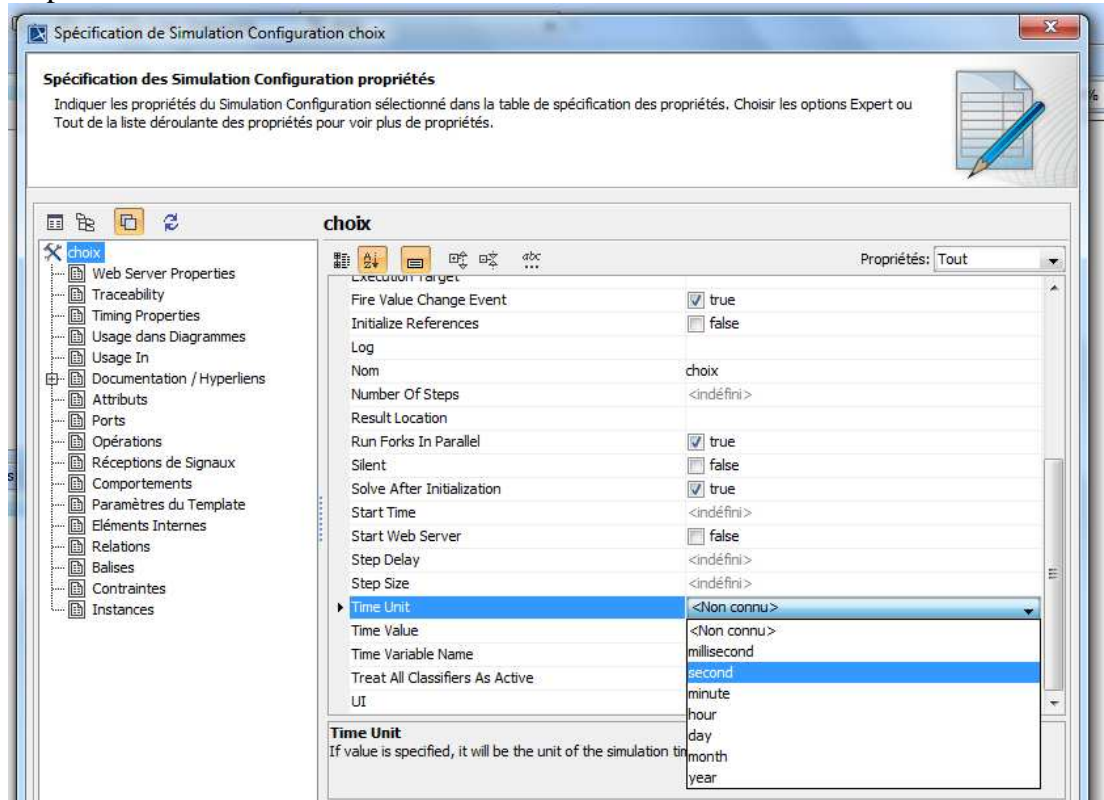
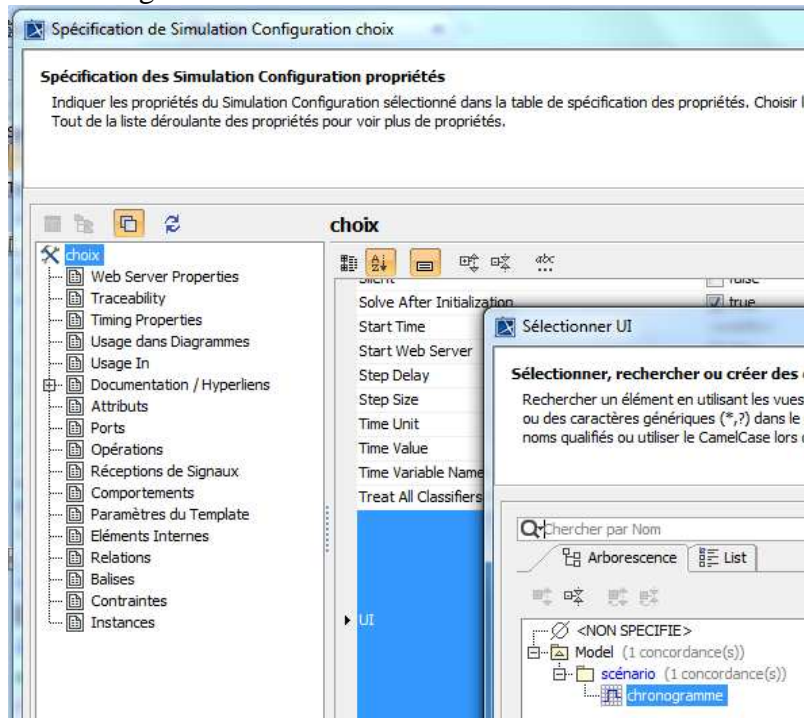
Fermer Retour Envoyer Aide

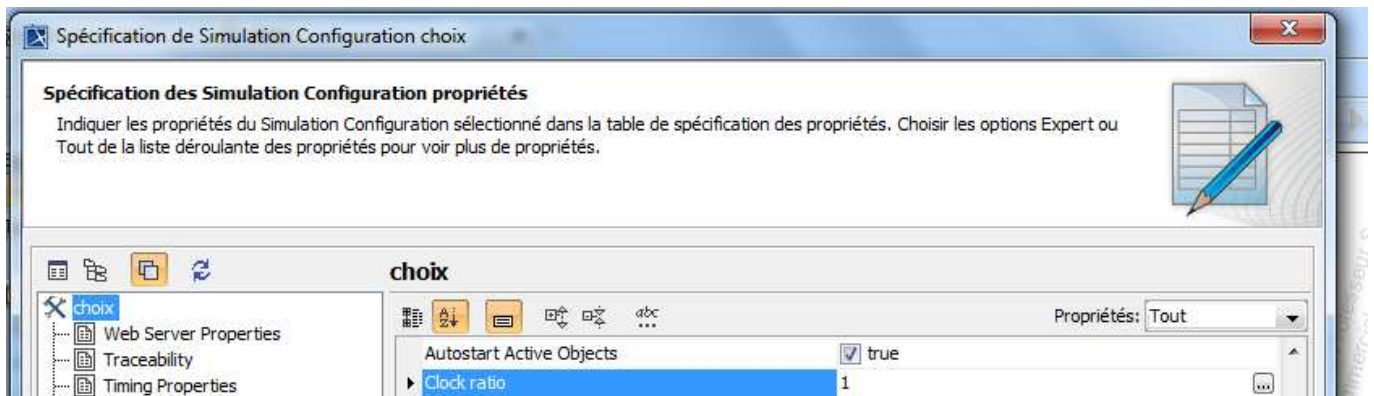


3) Création de la Timeline Chart **chronogramme**

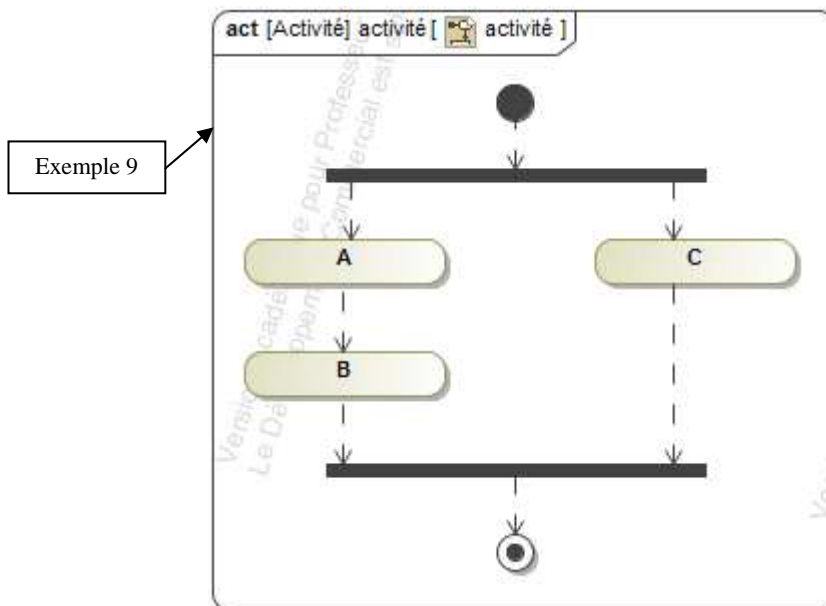
C'est l'animation d'un diagramme d'activité (act).



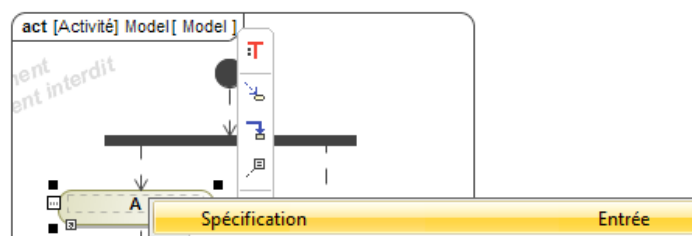
4) Configuration de la simulation **choix de configuration**Unité de temps en **seconde** :Le chronogramme sera **visible** dans la fenêtre d'animation.

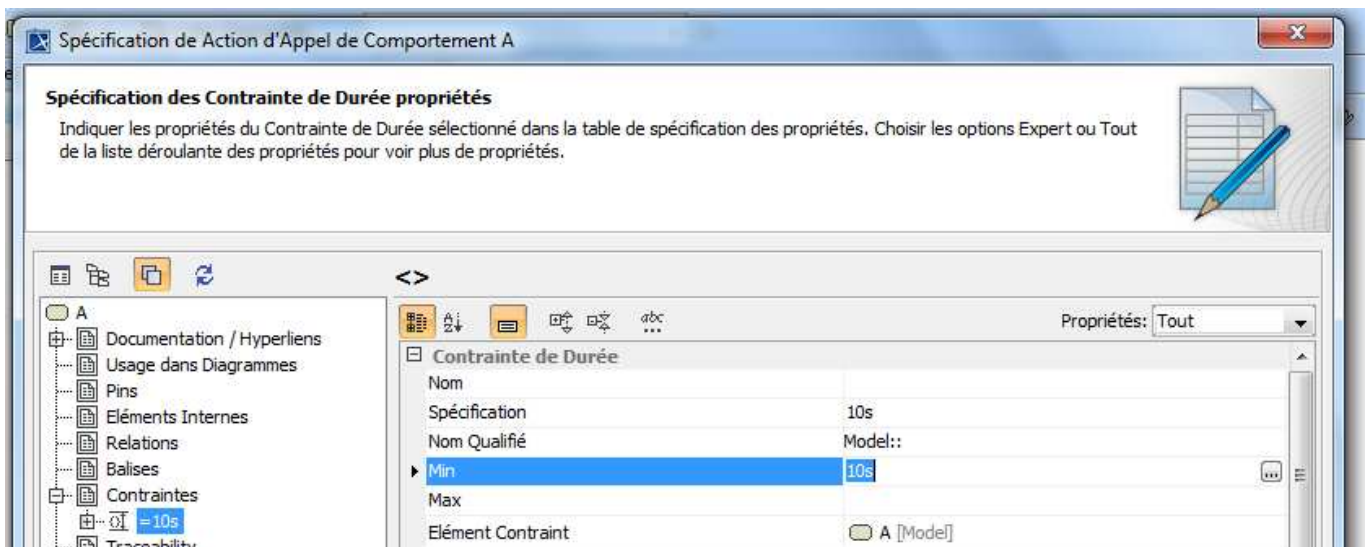
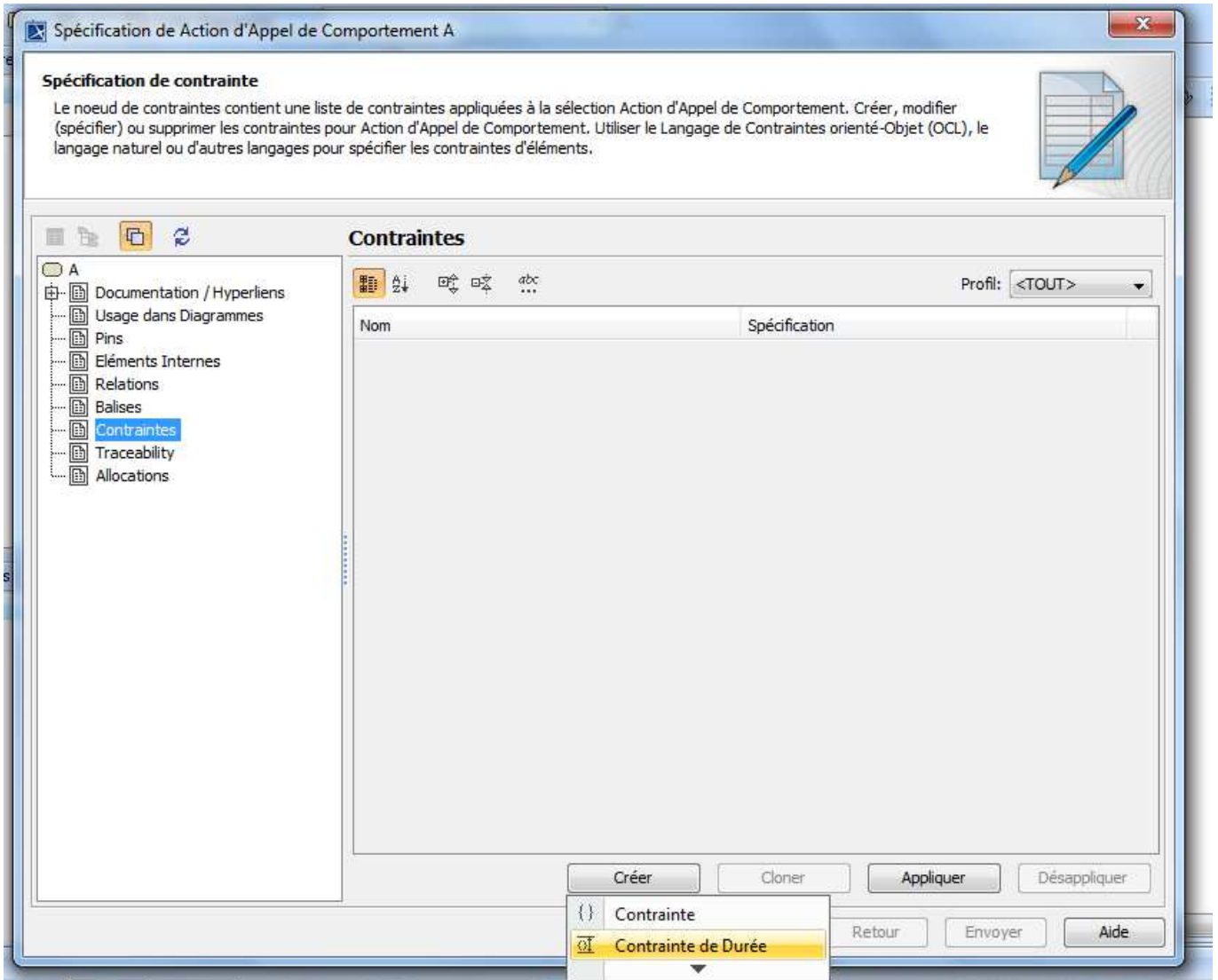


5) Editer le diagramme d'activité **activité**

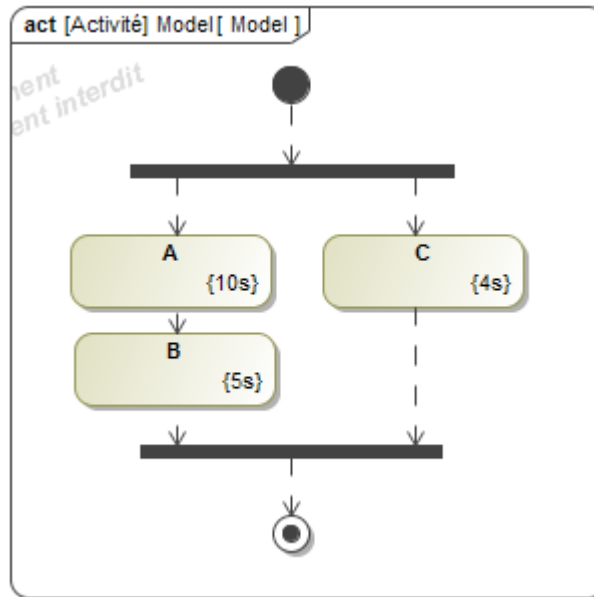


Configuration du temps de l'action élémentaire :

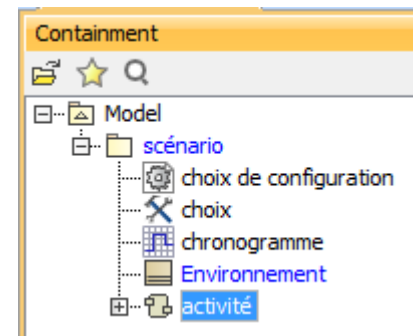
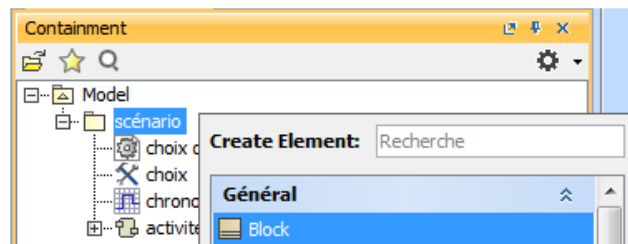




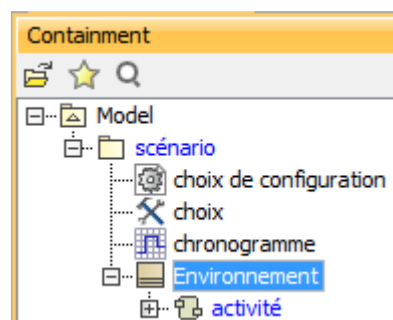
Finir le diagramme :

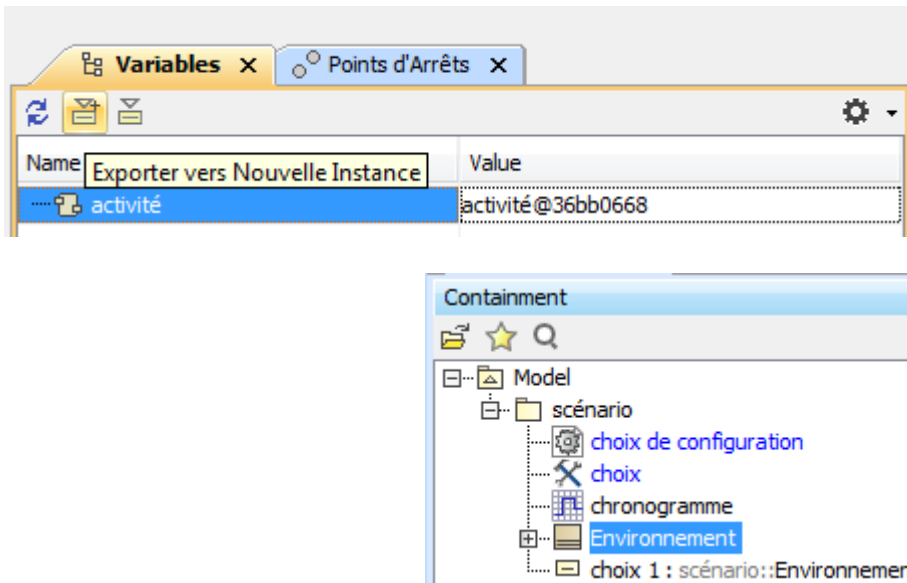
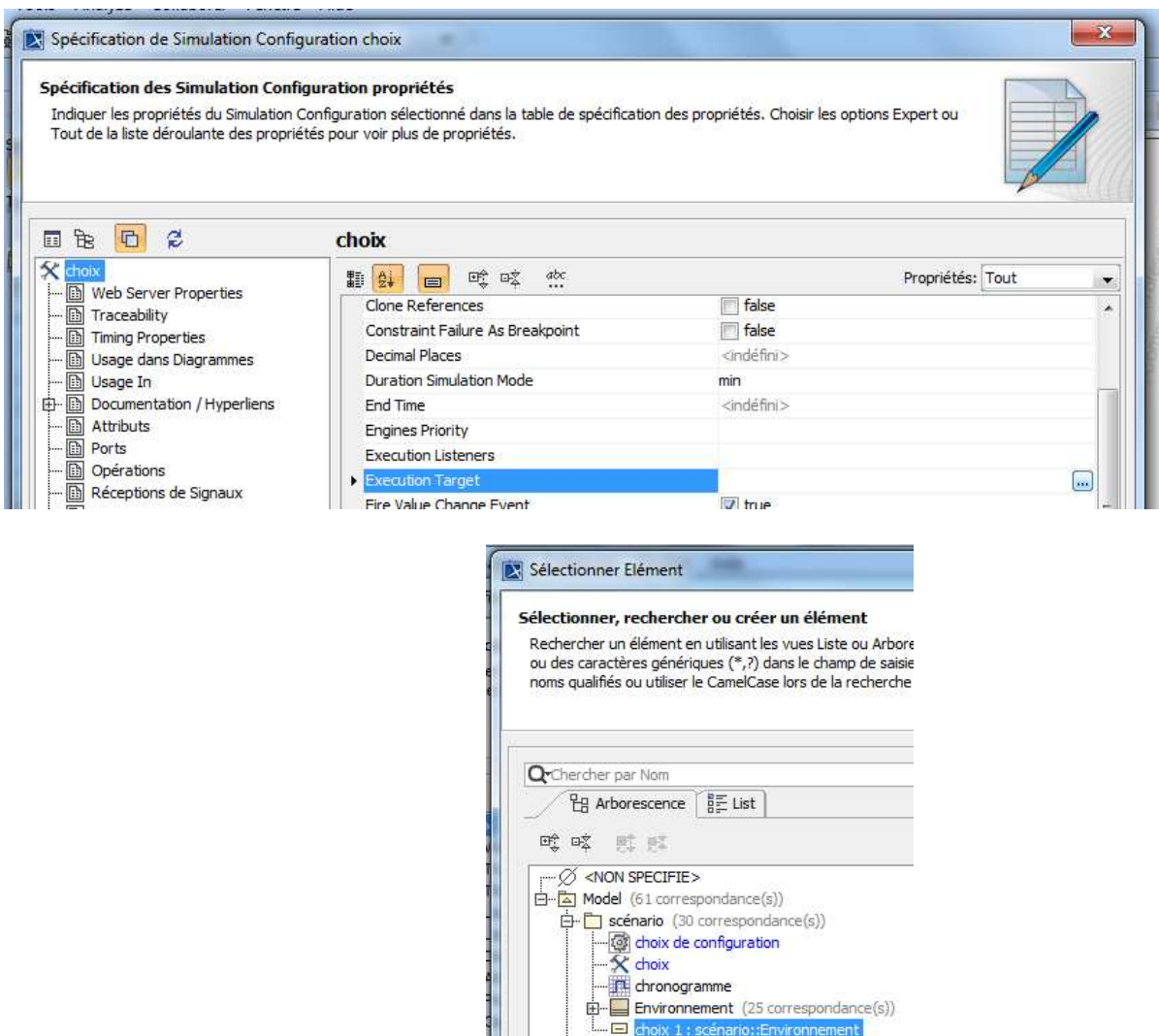


6) Créer le bloc **Environnement**



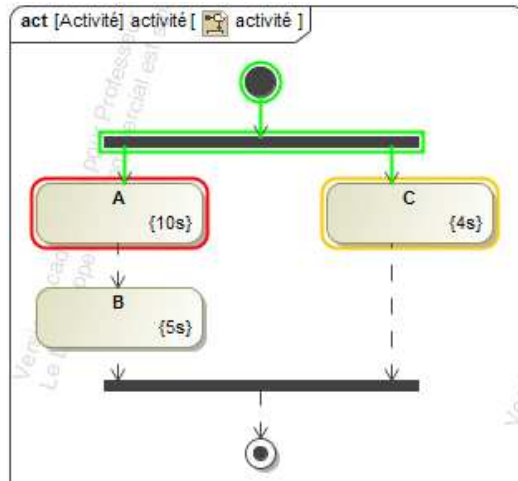
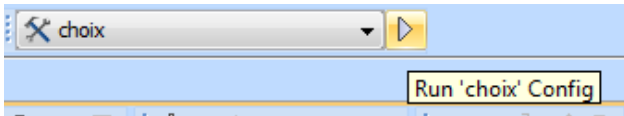
7) Déplacer le diagramme d'activité dans le bloc **Environnement**



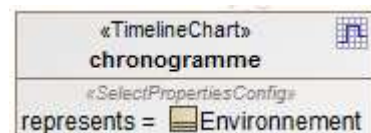
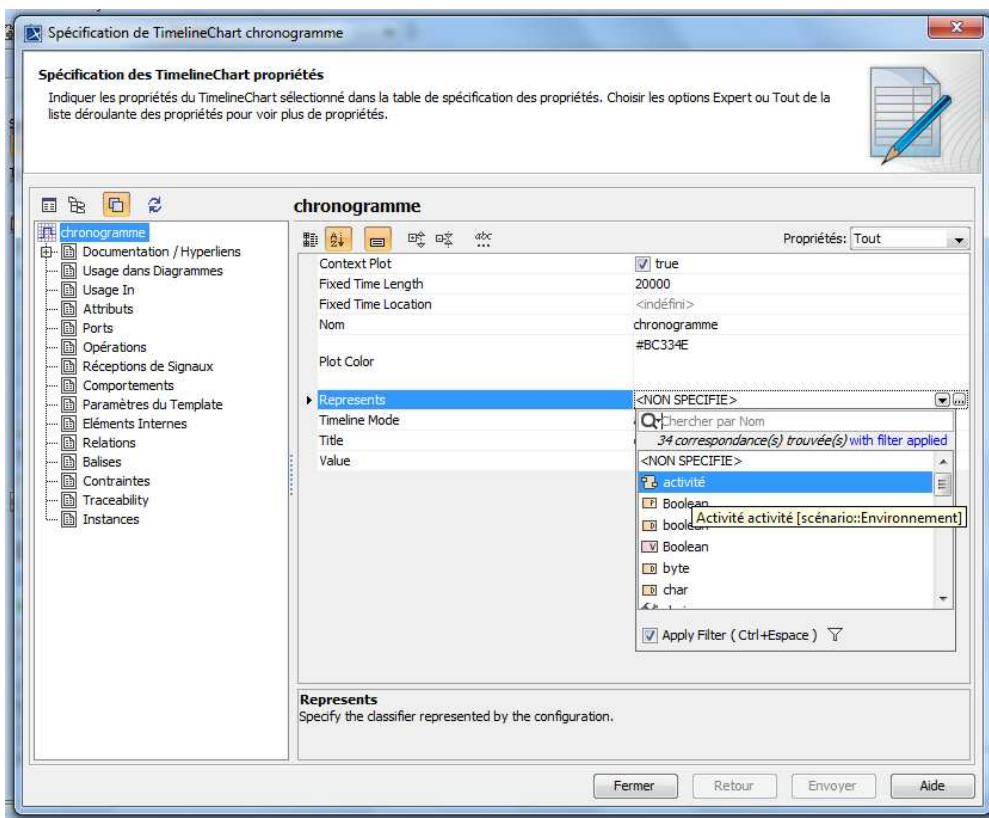
8) Créer une nouvelle instance **choix 1** liée au bloc **Environnement**9) Associer l'instance **choix 1** à la simulation



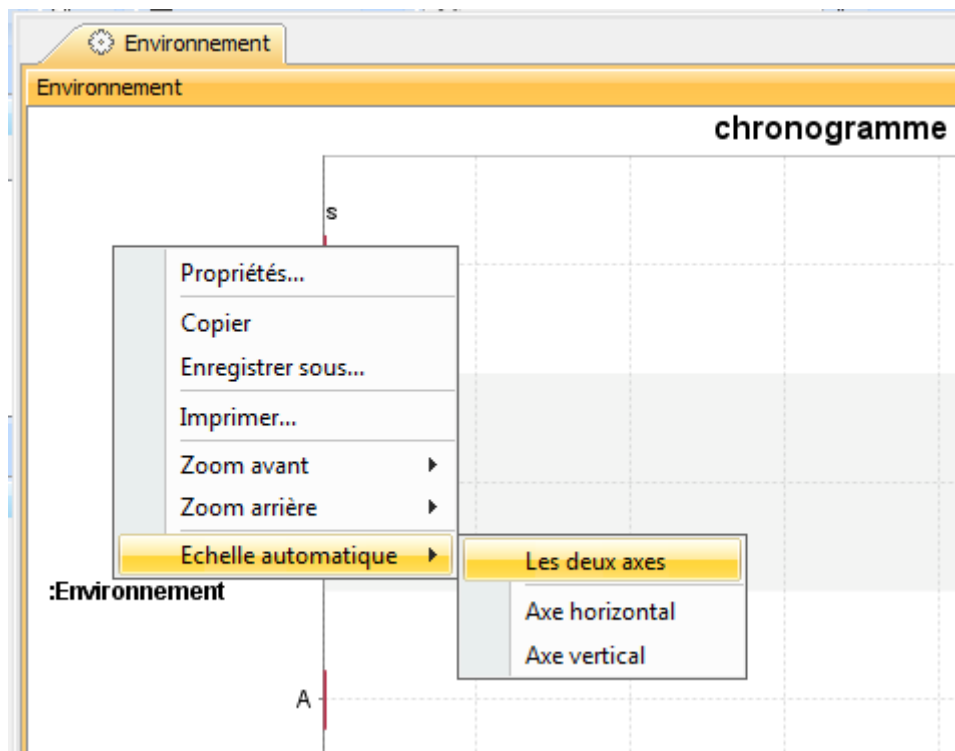
10) Simuler le comportement en temps réel



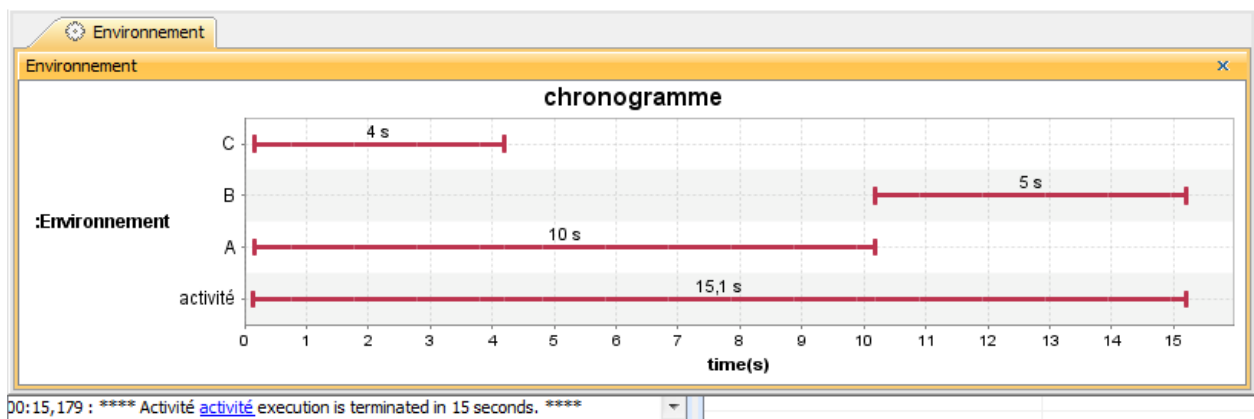
11) Simuler le comportement en temps réel avec le chronogramme



Choix d'une échelle automatique :



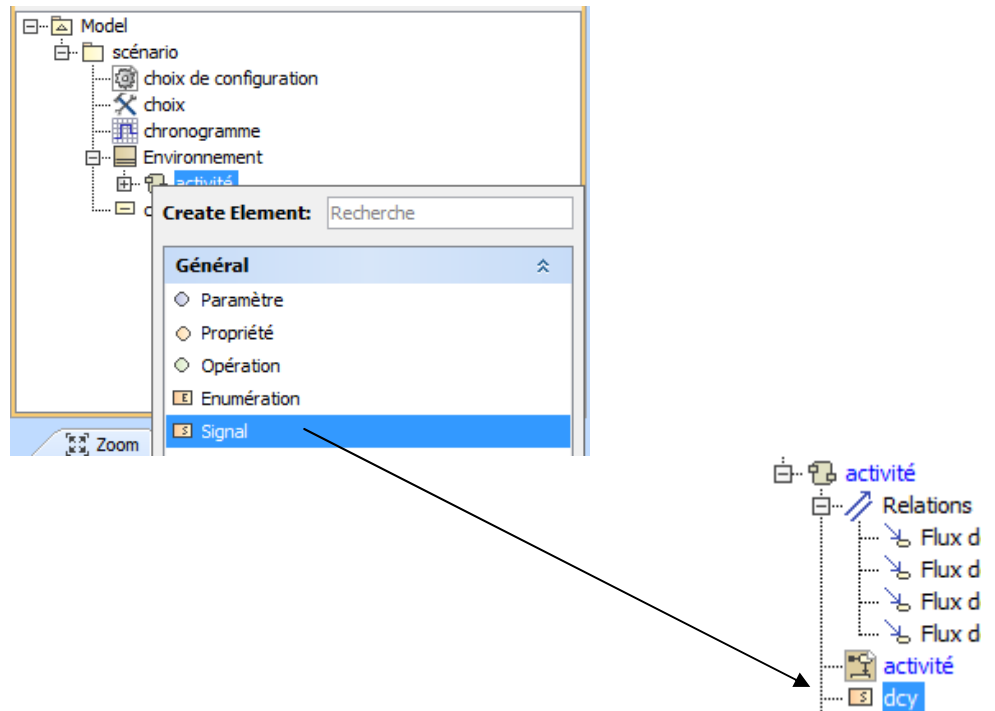
La représentation temporelle :



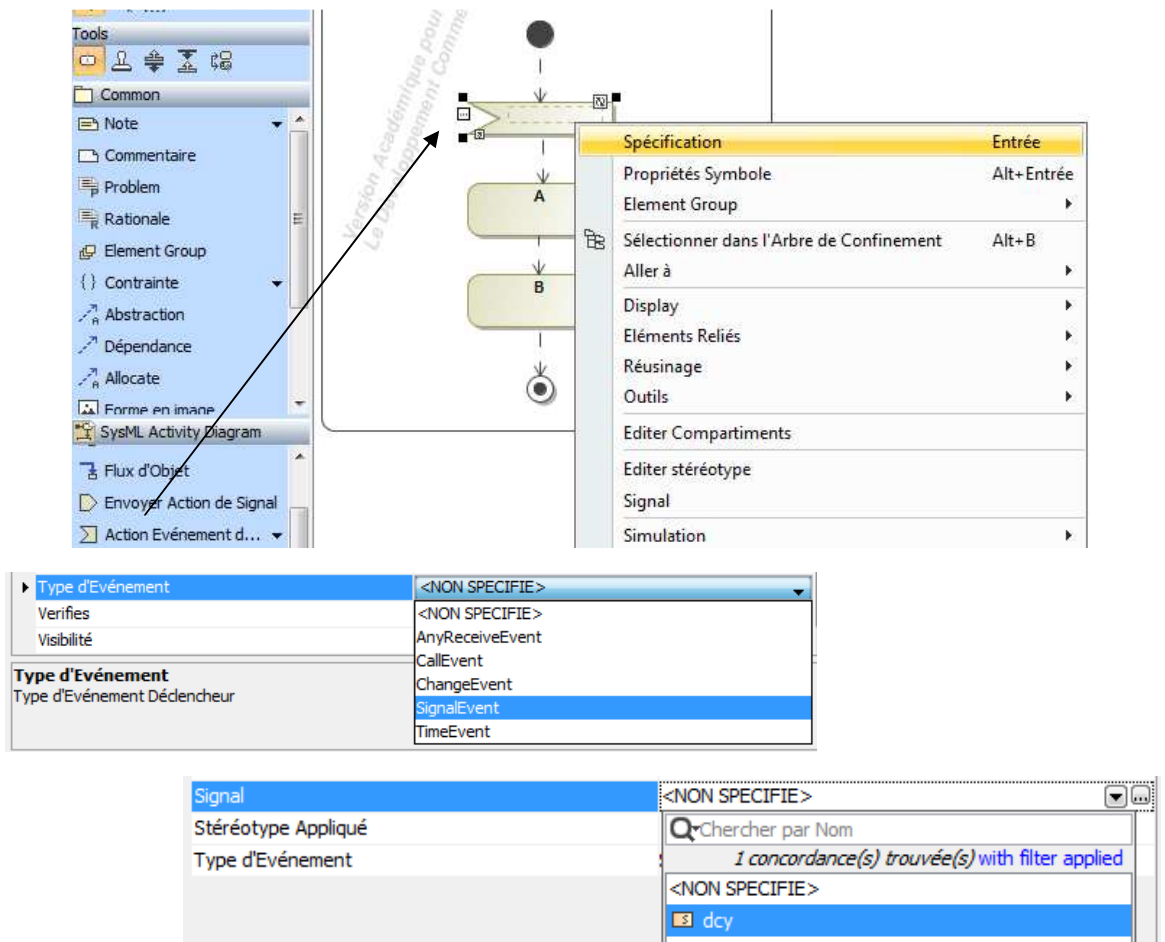
Exercice à simuler : **TP04 simulation 6.mdzip** (les durées -> **controleSysML\_2.pdf**)

**Simulation 7 : utilisation d'évènement extérieur au système (consigne de l'opérateur par exemple)**

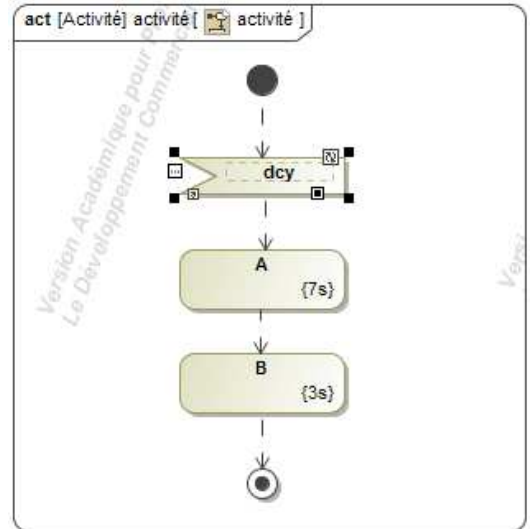
Création du signal **dcy** :



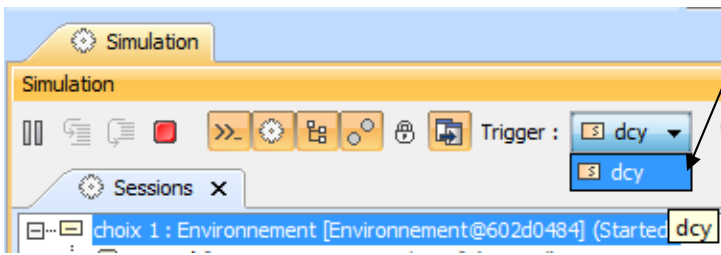
Edition du nœud d'action Evènement **dcy** :



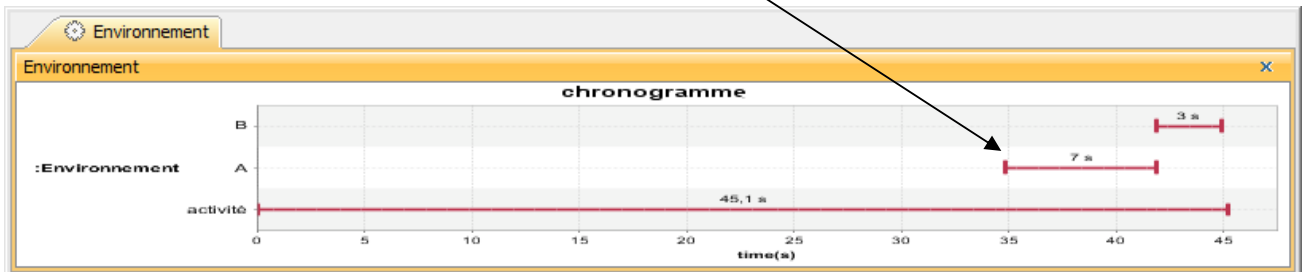
Finir l'édition du diagramme d'activité :



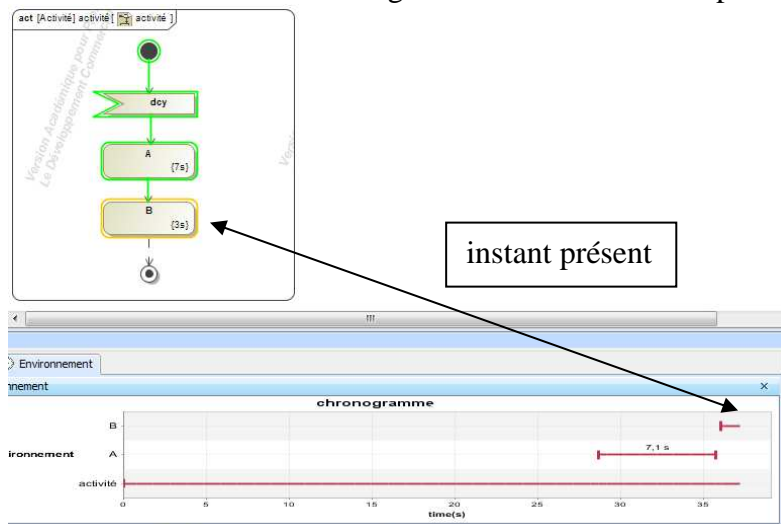
Lancer la simulation et commander l'évènement **dcy** :



Visualisation de l'instant où a eu lieu l'évènement **dcy** lors de la simulation :



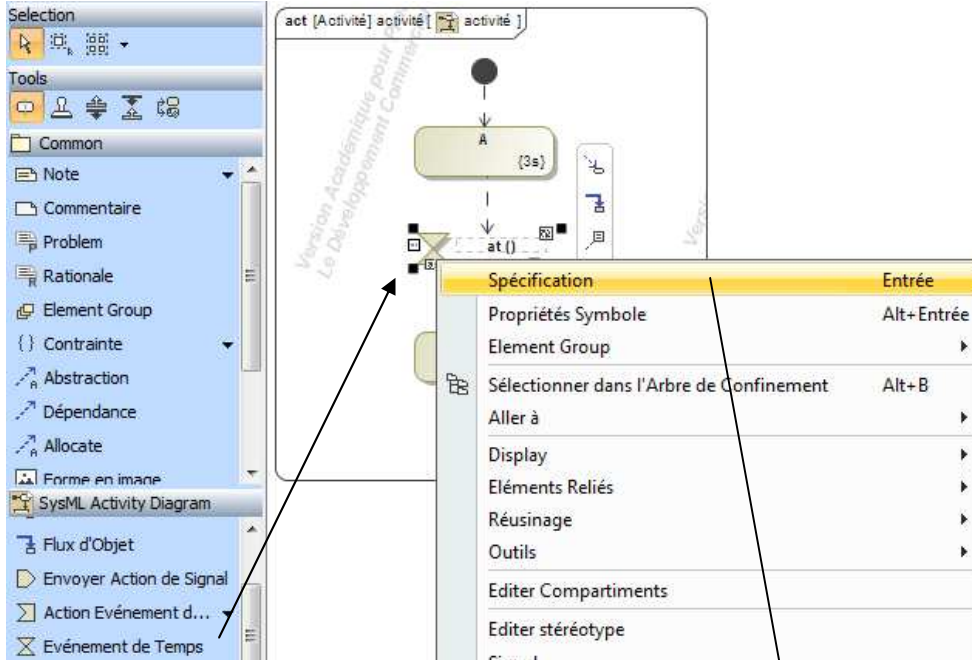
Visualisation de la concordance entre le diagramme d'activité et la représentation temporelle :



Exercice à simuler : TP05 simulation 7.mdzip (les durées -> 1.5s pour sortir ou rentrer la tige du vérin de transfert)

## Simulation 8 : utilisation d'évènement de temps

Edition d'un évènement de temps relatif :



Spécification de Action d'Évènement d'Appel <>

**Spécification des Action d'Évènement d'Appel propriétés**  
Indiquer les propriétés du Action d'Évènement d'Appel sélectionné dans la table de spécification des propriétés. Choisir les options Expert ou Tout de la liste déroulante des propriétés pour voir plus de propriétés.

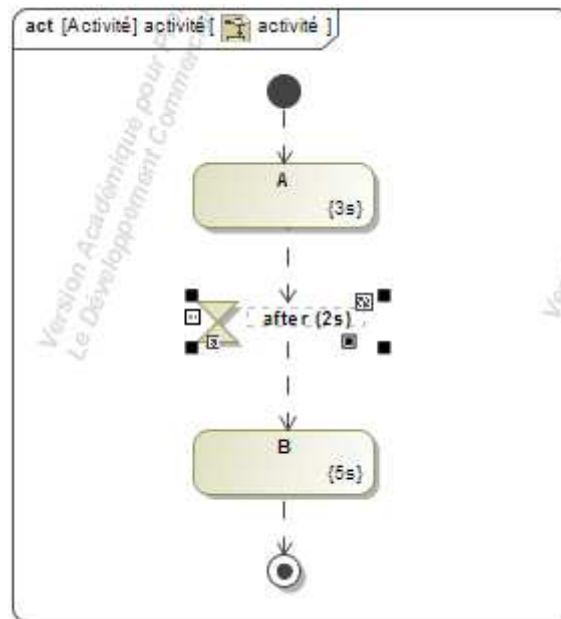
A Faire	
Déclencheur	<input checked="" type="checkbox"/> Déclencheur:at () [scénario::Environnement::activi...
Élément ID	_18_5_2_dba0322_1516439785884_576976_16246
Élément d'Évènement	<input checked="" type="checkbox"/> TimeEvent at () [scénario::Environnement::activité]
Est Converti	<input type="checkbox"/> false
<b>Est Relatif</b>	<input type="checkbox"/> false

Déclencheur	<input checked="" type="checkbox"/> Déclencheur:after ()
Élément ID	_18_5_2_dba0322_1516
Élément d'Évènement	<input checked="" type="checkbox"/> TimeEvent after () [s
Est Converti	<input type="checkbox"/> false
<b>Est Relatif</b>	<input checked="" type="checkbox"/> true

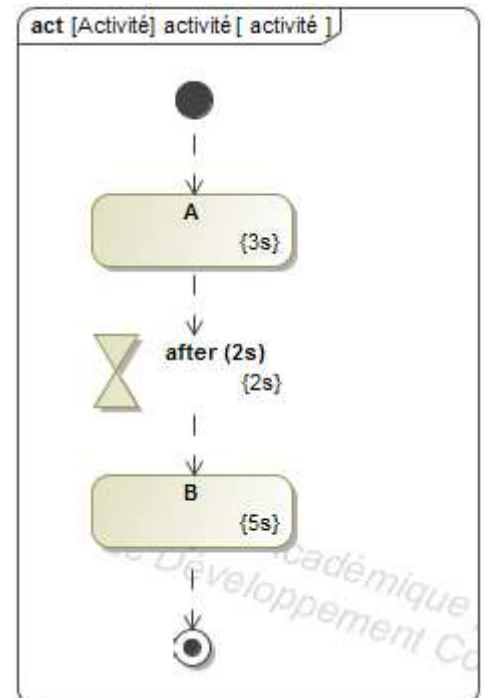
Saisir la durée de la temporisation :

Est Relatif	<input checked="" type="checkbox"/> true
Noeud Redéfini	
Nom	
Possesseur	activité [scénario::Environnement]
<b>Quand</b>	2s

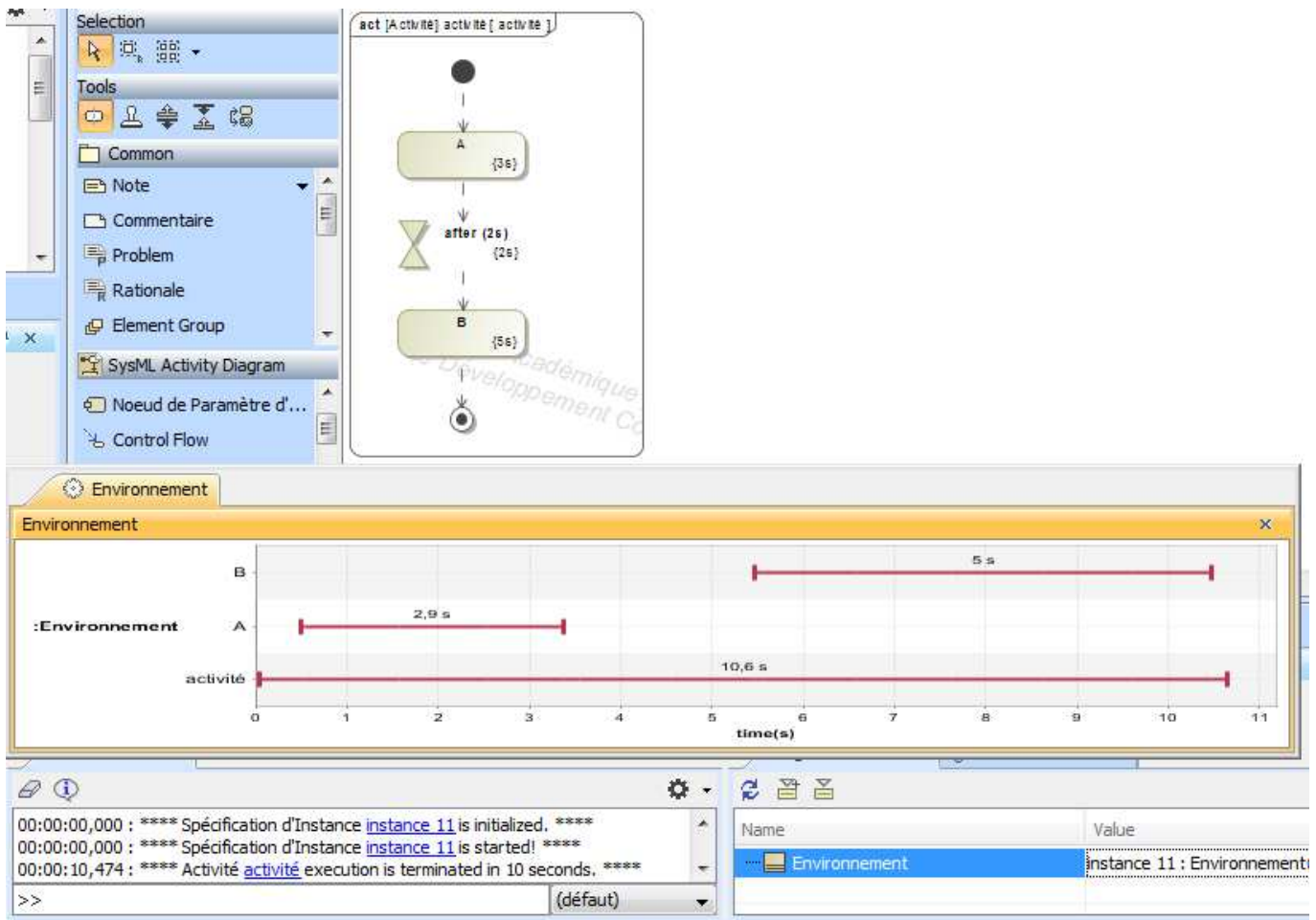
Visualisation du résultat :



Prise en compte de la durée de 2 secondes (contrainte de durée mini) :

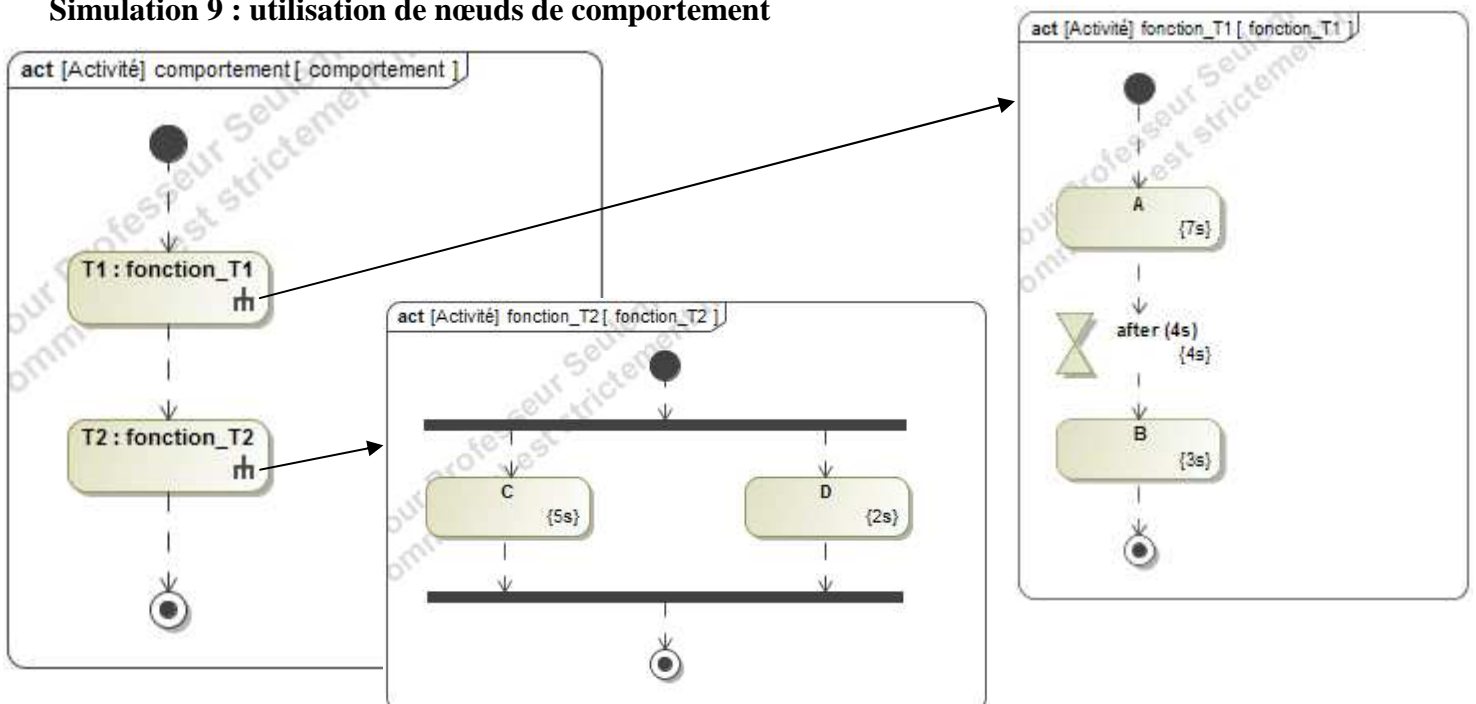


La simulation est terminée. L'exécution a bien duré 10 secondes.

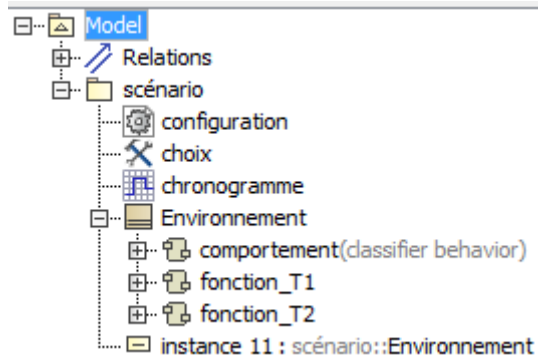


Exercice à simuler : TP03 EX2 simulation 8.mdzip (les durées -> 0.5s pour sortir ou rentrer la tige du vérin)

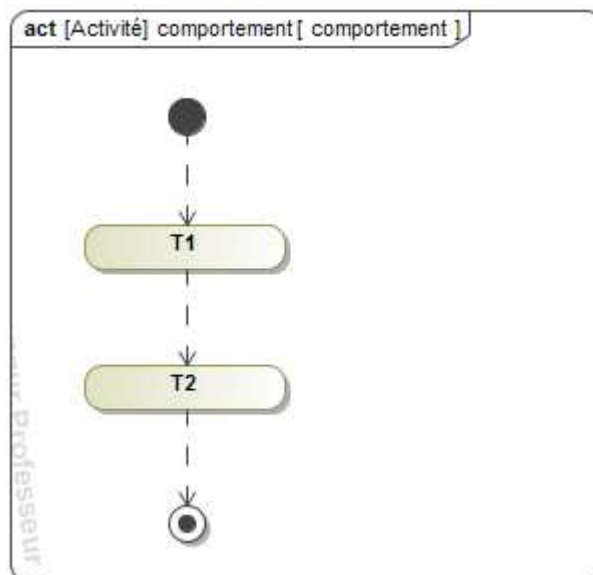
### Simulation 9 : utilisation de nœuds de comportement



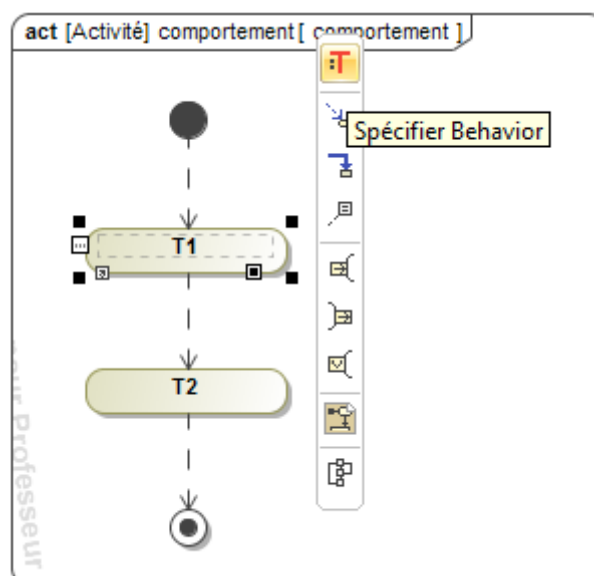
La structure du projet :



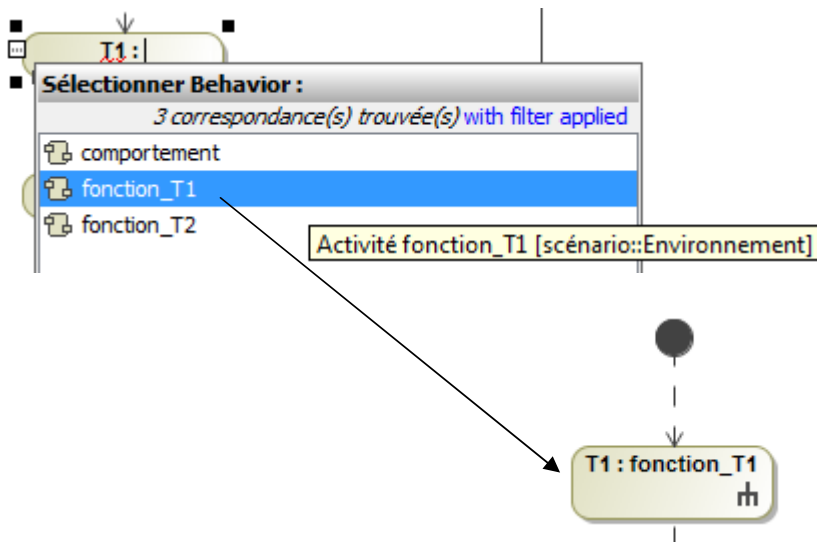
- 1) Editer les diagrammes d'activité **fonction\_T1** et **fonction\_T2**
- 2) Editer le diagramme **comportement** comme ci-dessous



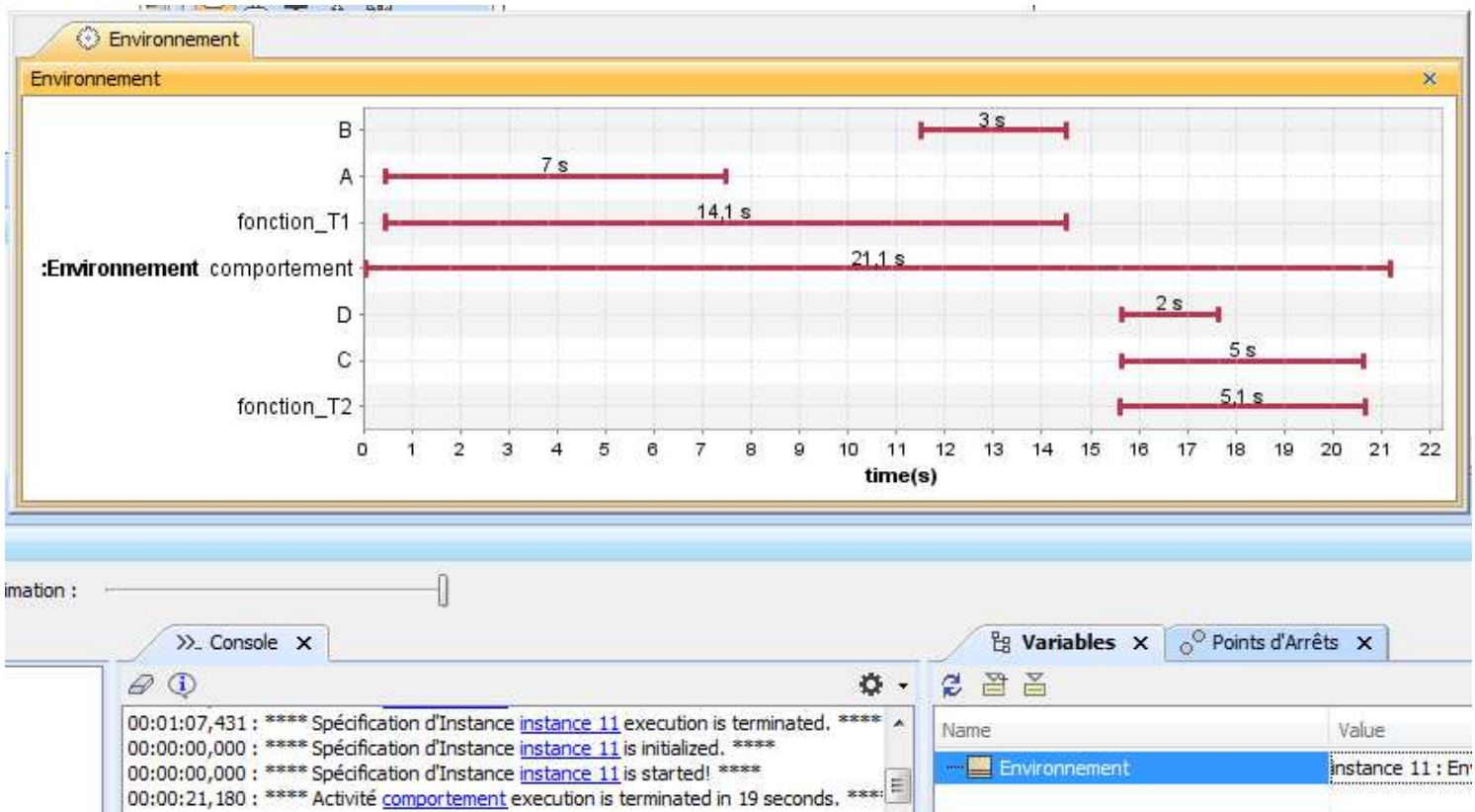
- 3) Associer les comportements aux nœuds







Visualisation de la simulation dont l'activité a duré **19 secondes** :



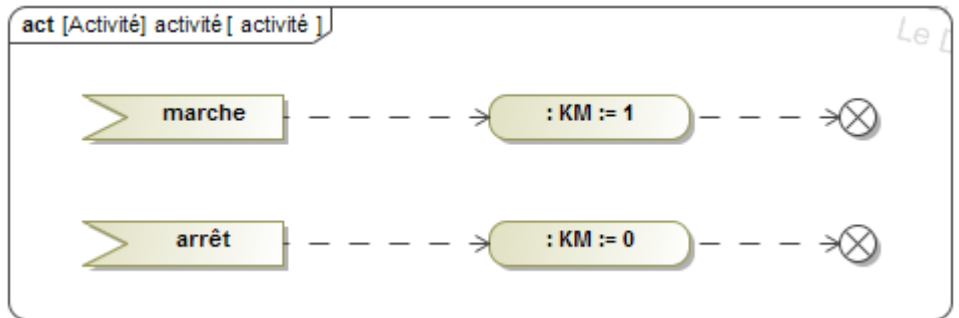
*Exercice à simuler : TP04 simulation 9.mdzip*

(les durées -> **controleSysML\_2.pdf**,

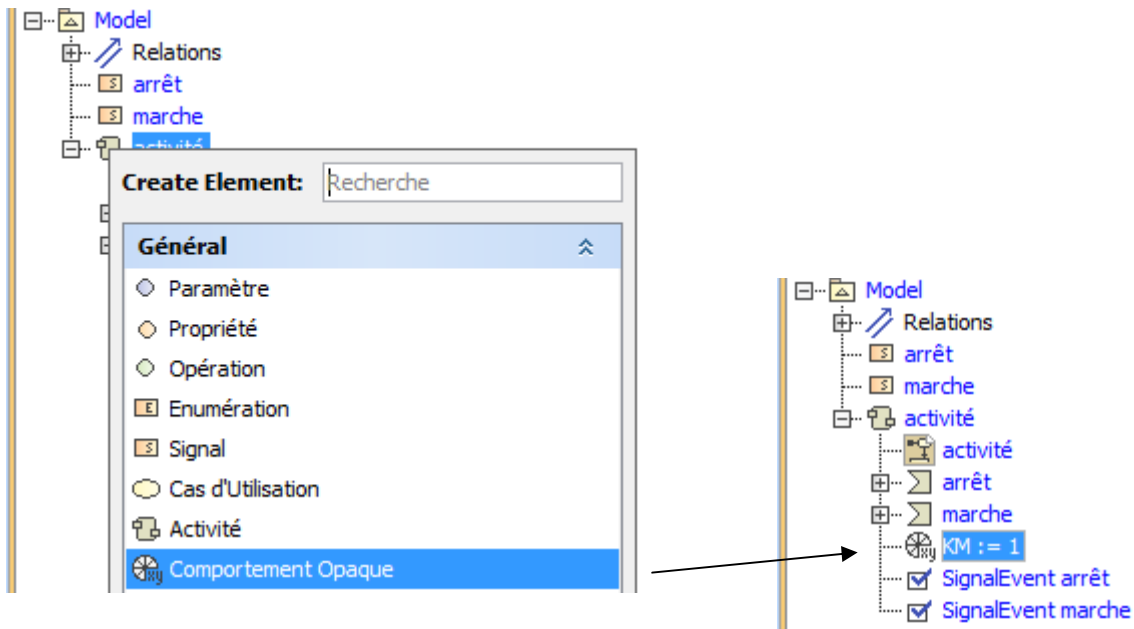
la coordination des tâches opératives et les tâches -> **4\_TP Siemens\_SysML\_corrige.pdf** )

**Simulation 10 : affectation d'une variable**

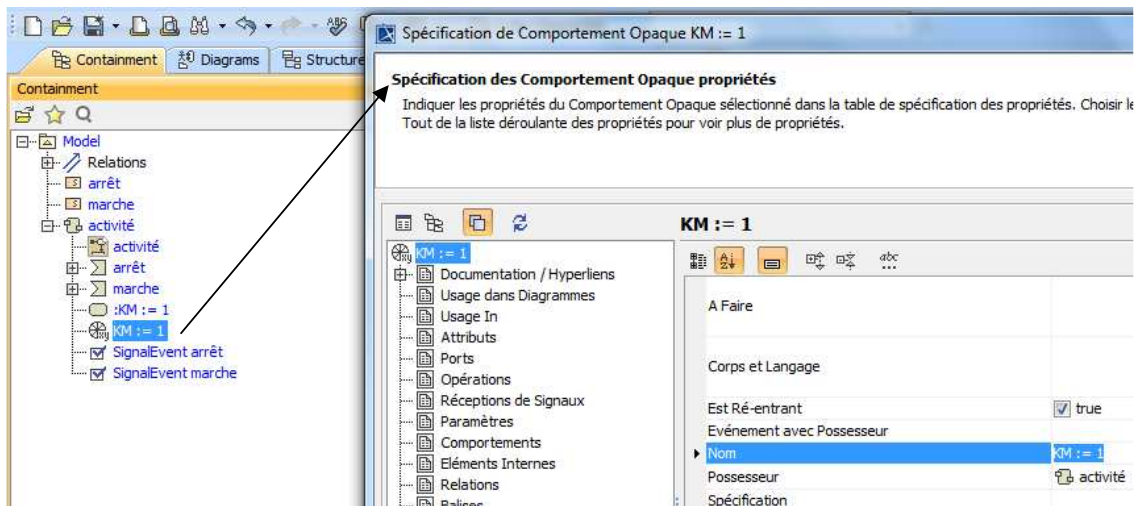
Exemple :



1) Edition de l'affectation de la variable KM de type booléen (KM := 1)

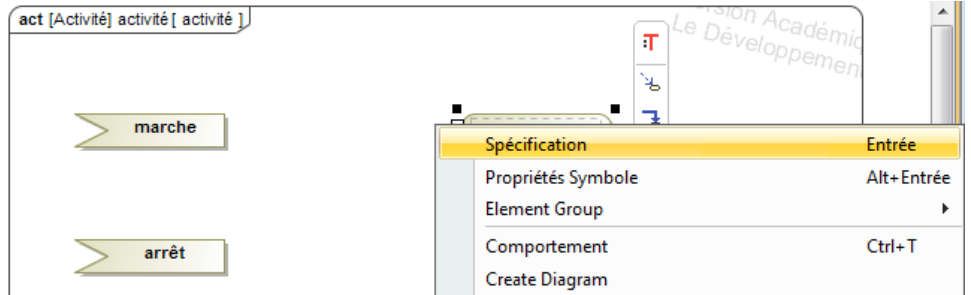


Traitement associé (KM = true ;)

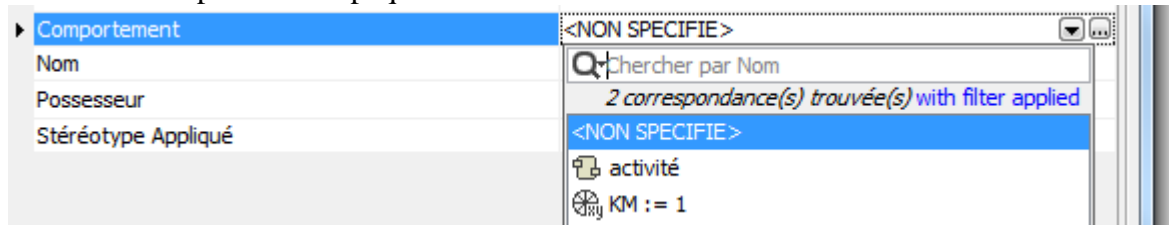




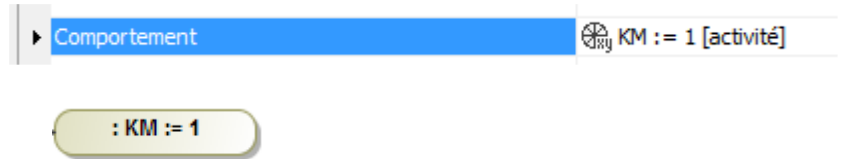
2) Edition du nœud d'action ( : KM := 1 )



Choisir le comportement opaque KM := 1

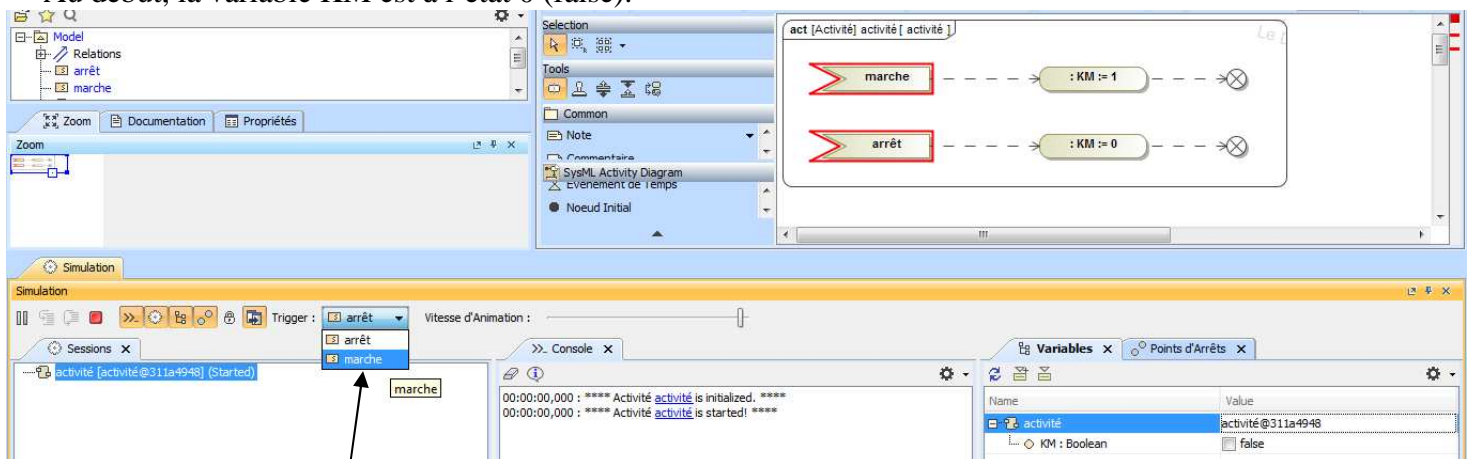


Résultat obtenu :



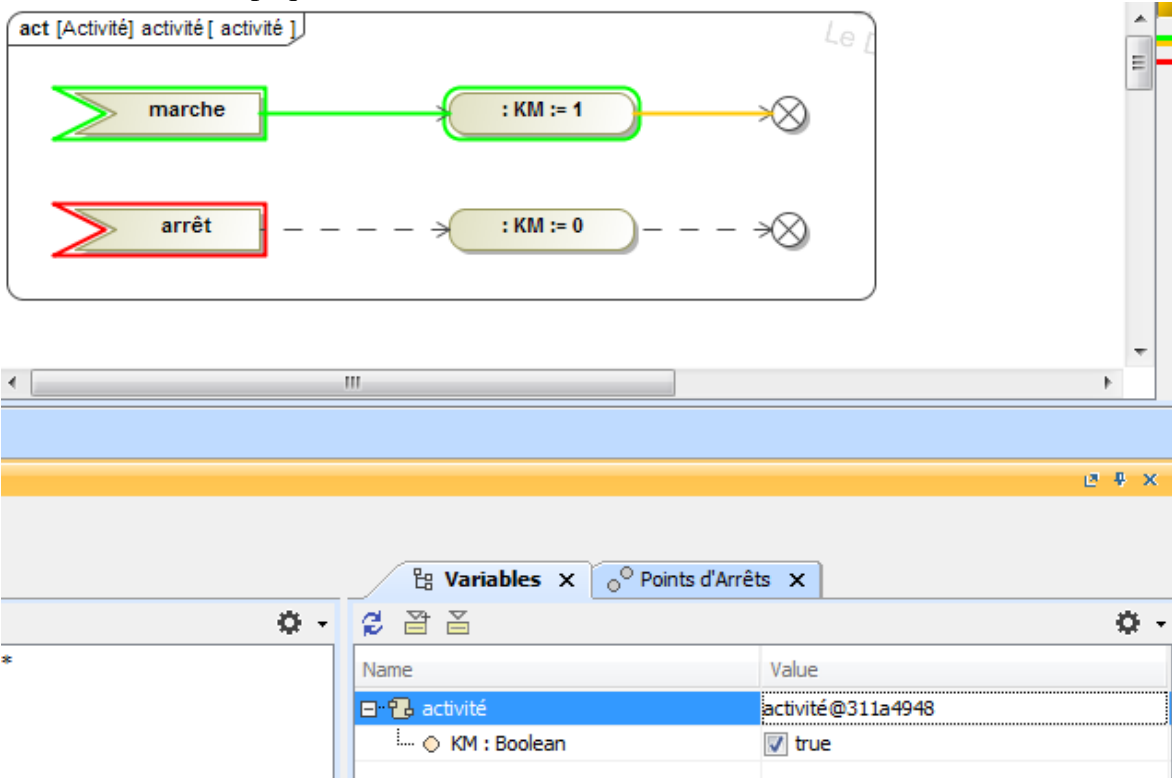
Visualisation de la simulation :

Au début, la variable KM est à l'état 0 (false).



On envoie le signal **marche**.

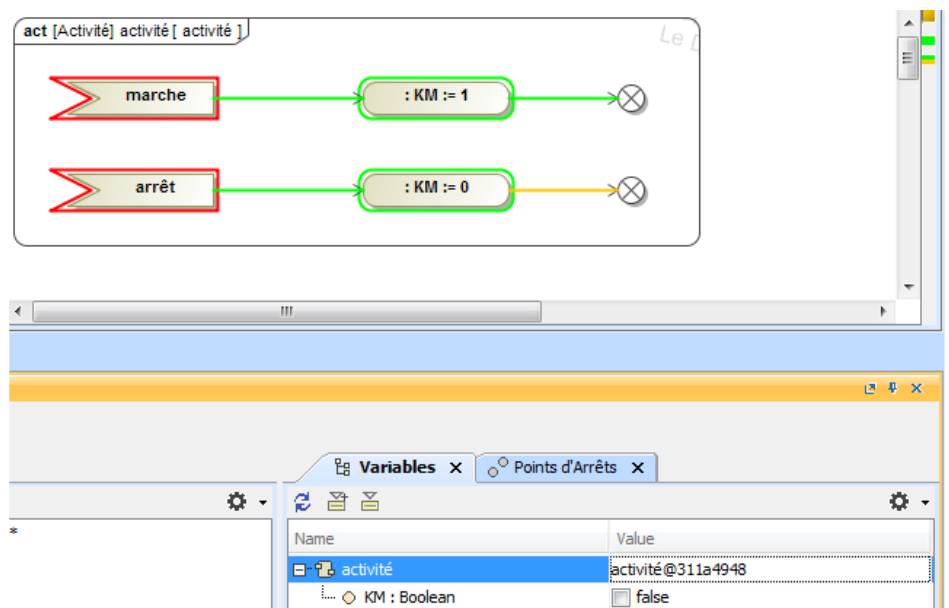
Le nœud d'action opaque **KM := 1** a été activé. La variable KM est mise à 1 (**true**).



On envoie le signal **arrêt**.



Le nœud d'action opaque **KM := 0** a été activé. La variable KM est mise à 0 (**false**).



Exercice à simuler : TP03 EX1\_2 simulation 10.mdzip

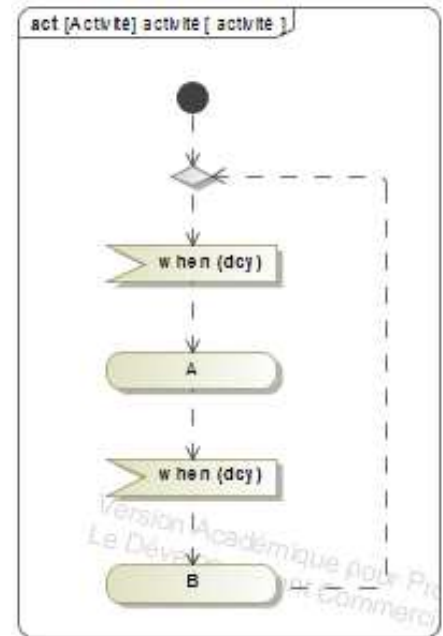
### Simulation 11 : fronts montant et descendant

Exemple :

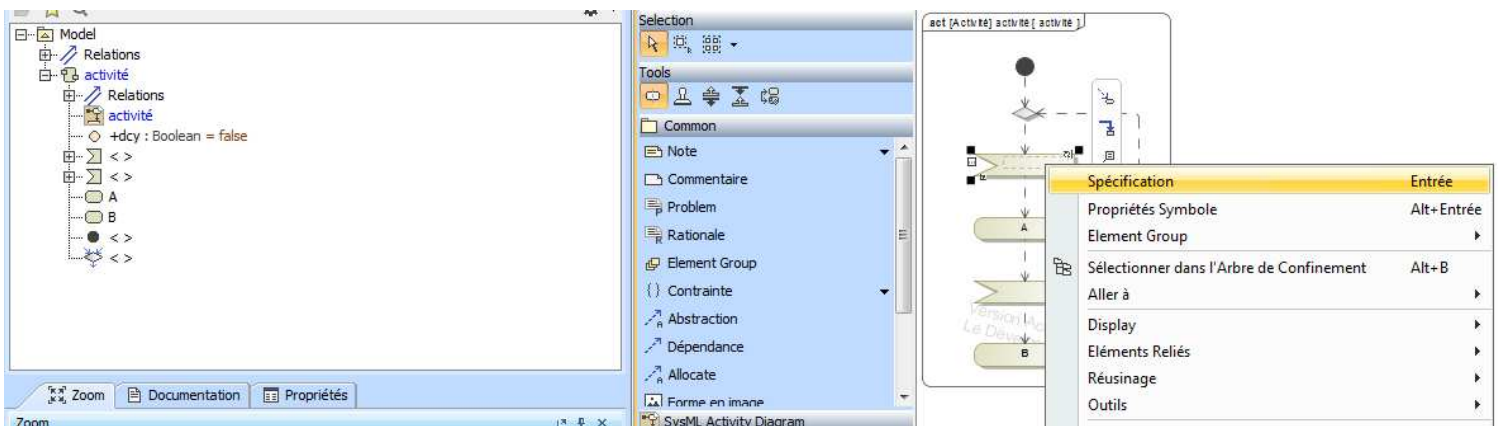
Explication :

Au début de l'activité, au front montant de dcy l'action A s'exécute.

Il faut attendre un nouveau front montant de dcy pour exécuter l'action B.

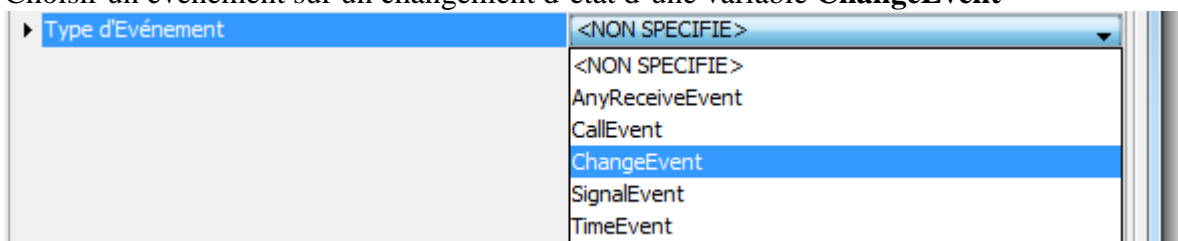


Edition du front montant de la variable dcy :



Spécification	Entrée
Propriétés Symbole	Alt+Entrée
Element Group	
Sélectionner dans l'Arbre de Confinement	Alt+B
Aller à	
Display	
Éléments Reliés	
Réusinage	
Outils	

Choisir un évènement sur un changement d'état d'une variable **ChangeEvent**



Type d'Évènement	<NON SPECIFIE>
	<NON SPECIFIE>
	AnyReceiveEvent
	CallEvent
	<b>ChangeEvent</b>
	SignalEvent
	TimeEvent

Associer la variable dcy au champ **Changer l'Expression**

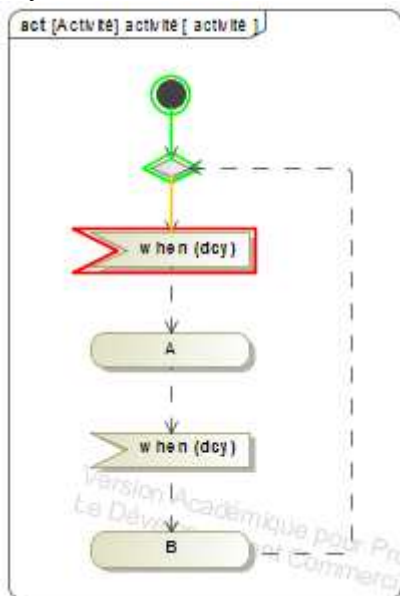
Changer l'Expression	dcy
Déclencheur	Déclencheur:when () [activité::]
Élément ID	_18_5_2_dba0322_1532938060410_463385_14177
Élément d'Événement	<input checked="" type="checkbox"/> ChangeEvent when () [activité]
Est Converti	<input type="checkbox"/> false
Noeud Redéfini	
Nom	
Possesseur	activité
Stéréotype Appliqué	
Type d'Événement	ChangeEvent

Pour obtenir un front descendant, on écrit le NON de dcy ( !dcy).

Changer l'Expression	!dcy
----------------------	------

Visualisation de la simulation :

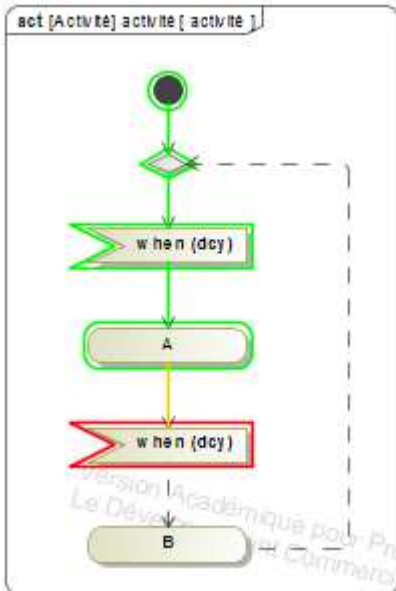
Au début de l'exécution de l'activité, la variable dcy est à l'état logique 0.  
Il y a attente du front montant de dcy.



Name	Value
activité	activité@5ae81c23
dcy : Boolean	<input type="checkbox"/> false

A partir de la fenêtre Variables, il faut changer l'état de dcy.

L'évènement (passage de l'état 0 à l'état 1 de dcy) a été pris en compte afin d'exécuter l'action A.  
On est en attente d'un nouveau front montant de dcy.

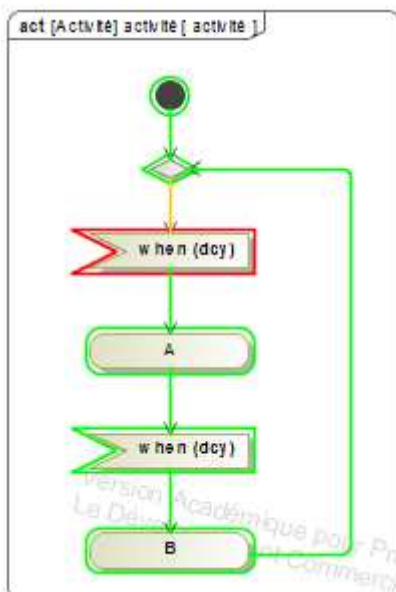


Name	Value
activité	activité@5ae81c23
dcy : Boolean	<input checked="" type="checkbox"/> true

A partir de la fenêtre Variables, passez dcy à l'état 0 logique.

Name	Value
activité	activité@5ae81c23
dcy : Boolean	<input type="checkbox"/> false

L'évènement (passage de l'état 0 à l'état 1 de dcy) a été pris en compte afin d'exécuter l'action B.  
On est en attente d'un nouveau front montant de dcy.



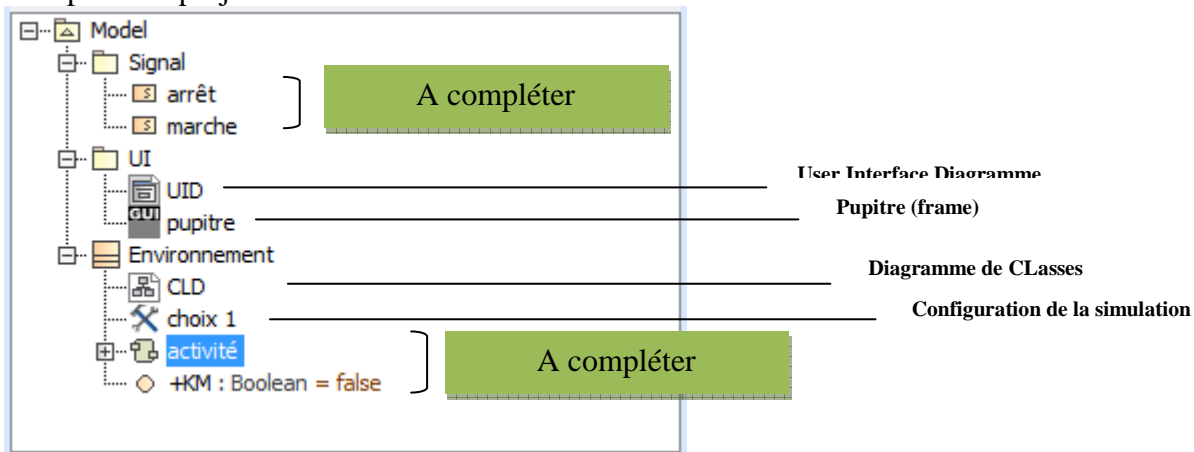
Name	Value
activité	activité@5ae81c23
dcy : Boolean	<input checked="" type="checkbox"/> true

Exercice à simuler : TP02 EX2 simulation 11.mdzip

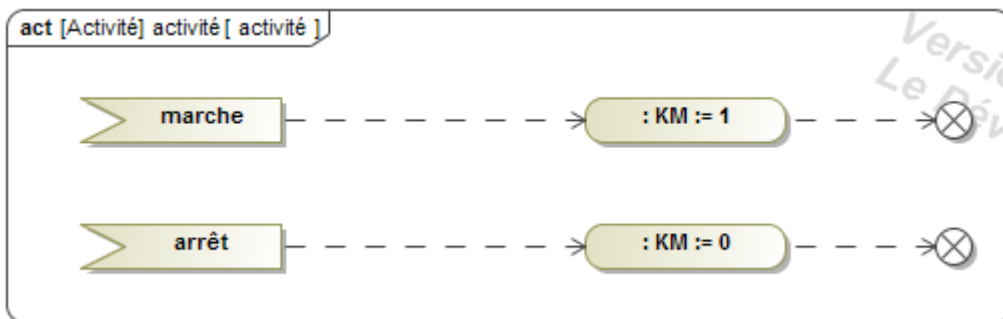
**Simulation 12 : utilisation d'un pupitre**

L'exemple est à faire en complétant le fichier magicDraw **fichier\_simulation\_12\_à compléter.mdzip** et le renommer pour garder le fichier initial.  
Le diagramme d'activité proposé est celui de l'exemple de la simulation 10.

Compléter le projet comme ci-dessous :

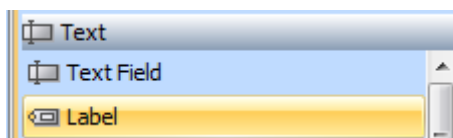


Le diagramme d'activité (act) :



Réalisation du pupitre :

- 1) Ecrire du texte (Label)

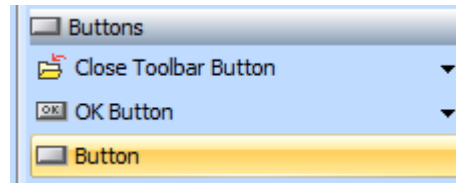


Résultat à obtenir :

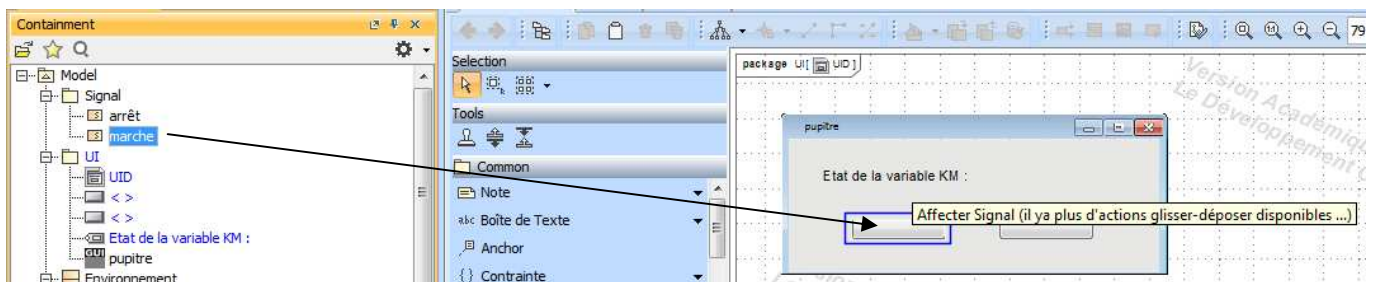




## 2) Insérer un bouton (Button)



Résultat à obtenir :

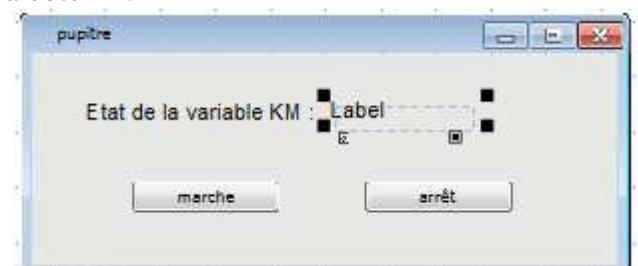
Associer un signal : **action glisser-déposer**

Résultat à obtenir :

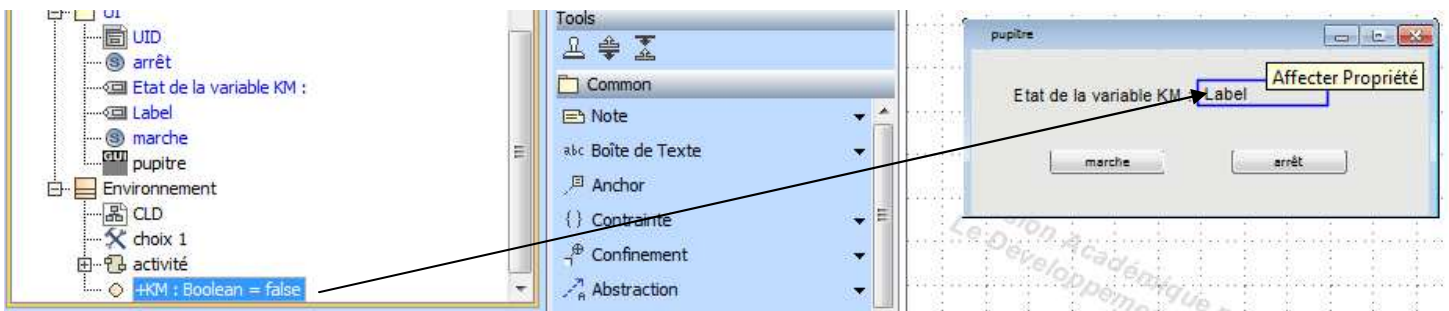


## 3) Afficher l'état d'une variable (Label)

Résultat à obtenir :



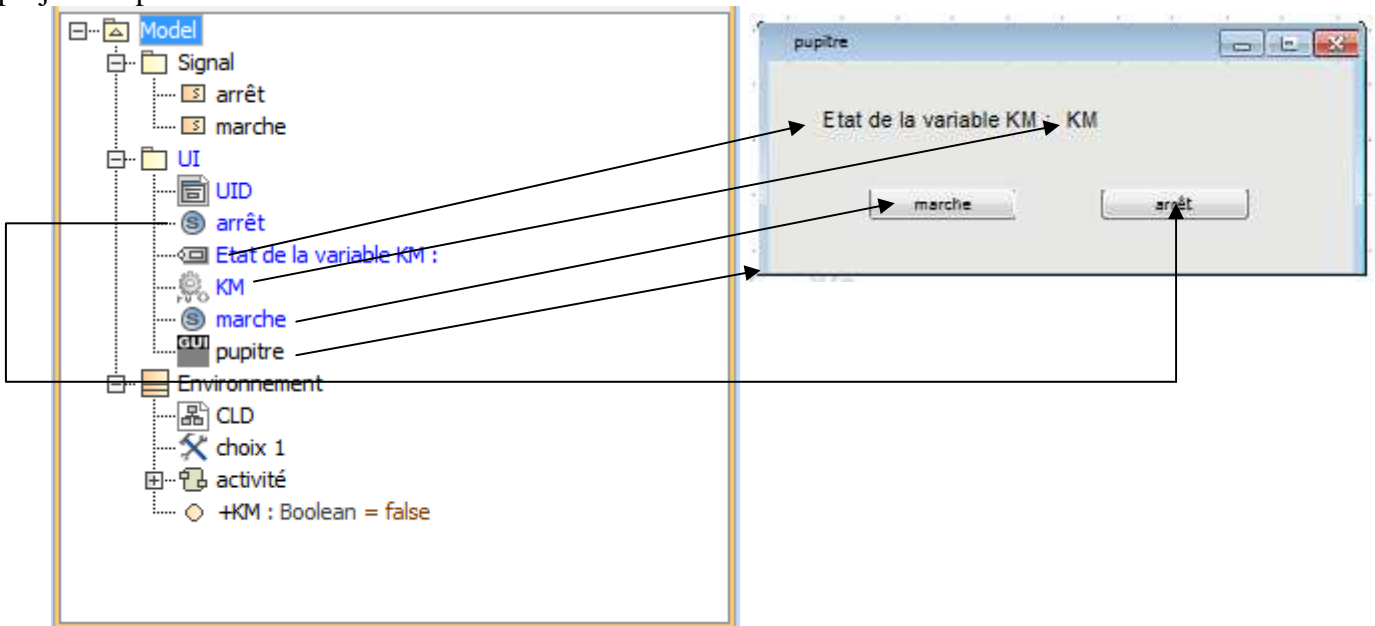
Associer une variable : **action glisser-déposer**



Résultat à obtenir :

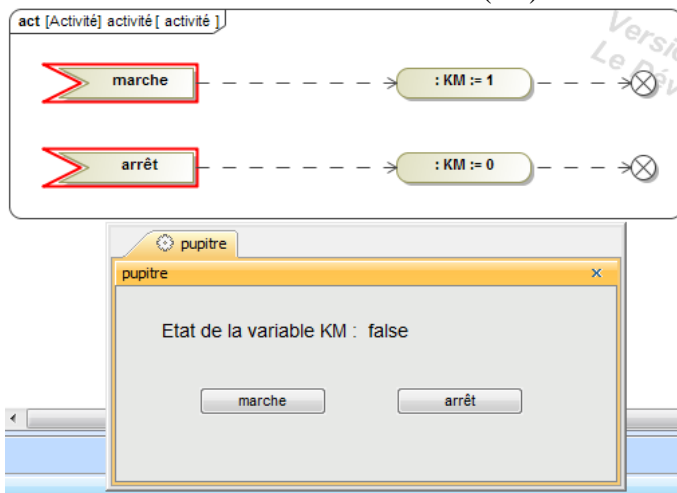


Le projet complet :



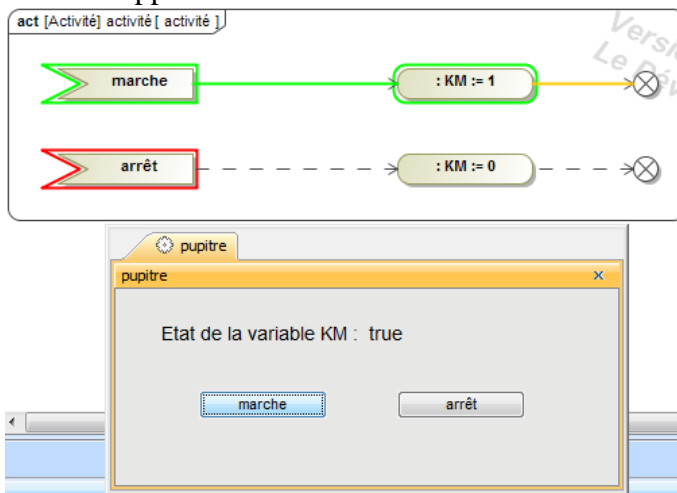
Visualisation de la simulation :

Au début de l'exécution de l'activité (act) :



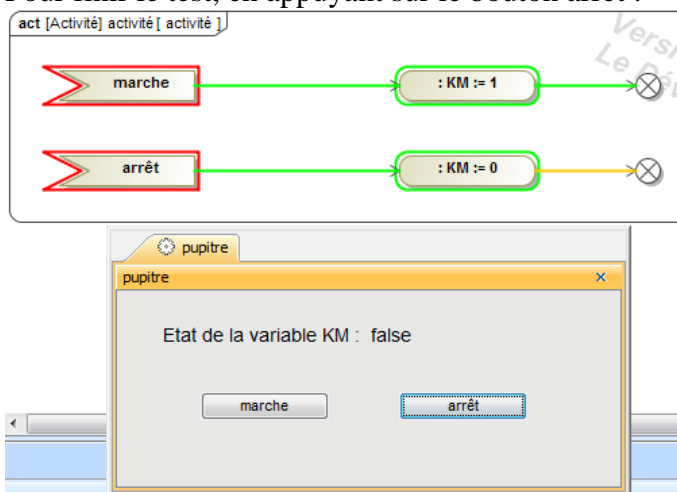
La variable KM est à l'état 0 logique (état initial déclaré).

Suite à l'appui sur le bouton marche :



La variable KM est à l'état 1 logique (true).

Pour finir le test, en appuyant sur le bouton arrêt :



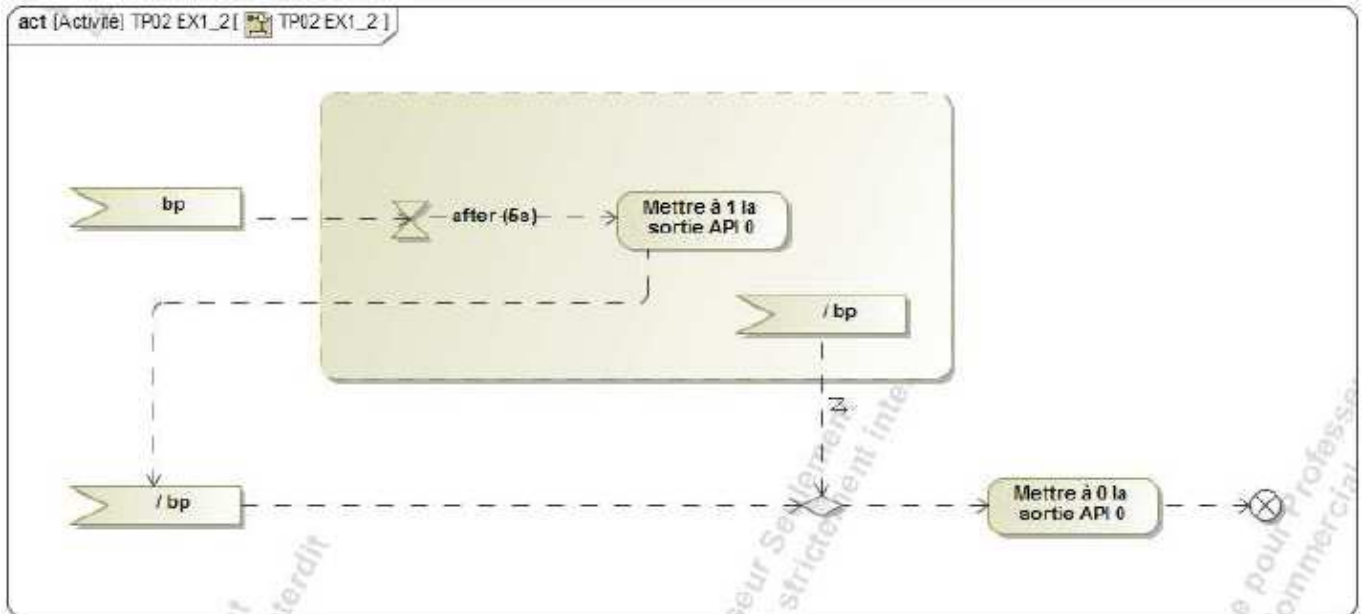
La variable KM est revenue à l'état 0 logique (false).

Exercice à simuler : TP03 EX1 simulation 12.mdzip

### Simulation 13 : région interruptible (équivalence)

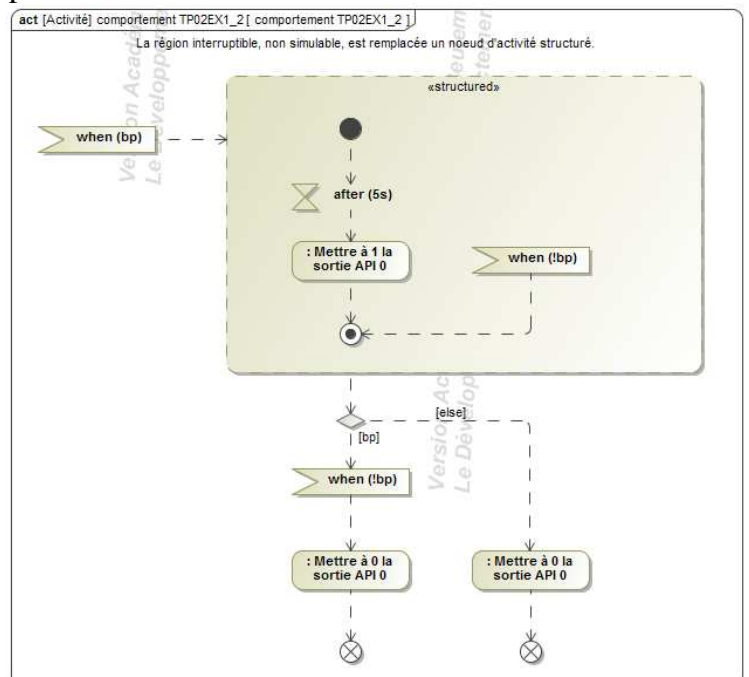
La région interruptible n'est actuellement pas simulable (réponse de la société No Magic).  
Je vous propose une structure équivalente qui permettra de simuler son comportement.

Exemple :



Pour la simulation de la région interruptible, vous allez utiliser un **nœud d'activité structuré** et un **nœud de décision**.

Représentez l'activité (act) équivalente suivante pour la simuler:



Exercice à simuler : TP02 EX5\_2 simulation 13.mdzip