|  |  |
| --- | --- |
| **Chapitre** | **2. Outils et méthodes d’analyse et de description des systèmes** |
| **Objectif général de formation** | * identifier les éléments influents d’un système, * décoder son organisation, * utiliser un modèle de comportement pour prédire ou valider ses performances. |
| **Paragraphe** | 2.3 Approche comportementale |
| **Sous paragraphe** | 2.3.6 Comportements informationnels des systèmes |
| **Connaissances** | Modèles de description en statique et en dynamique |
| **Niveau d’enseignement** | Première Terminale |
| **Niveau taxonomique** | **3.** Le contenu est relatif à la **maîtrise d’outils d’étude ou d’action** : utiliser, manipuler des règles ou des ensembles de règles (algorithme), des principes, des démarches formalisées en vue d’un résultat à atteindre. |
| **Commentaire** | *Activités pratiques liées à la mise en œuvre d’un produit industriel ou d’un système permettant l’application des différents modèles de description de l’information (en statique et en dynamique) et la caractérisation des entrées-sorties de ses différents constituants.*  *Les modèles de comportement sont étudiés autour d’un point de fonctionnement. Au niveau de l’expression de l’information on se limite aux grandeurs statistiques usuelles (moyenne et écart type)* |
| **Liens** |  |

## Les diagrammes comportementaux SysML

Le **comportement** des systèmes est décrit par SysML au moyen de 4 types de diagrammes :

* diagramme de cas d’utilisation
* diagramme de séquence
* diagramme d’activité
* diagramme d’état

Le premier (cas d’utilisation) est un diagramme statique, qui décrit l’aspect fonctionnel.

Les trois autres sont des diagrammes qui décrivent l’aspect dynamique du comportement du système.

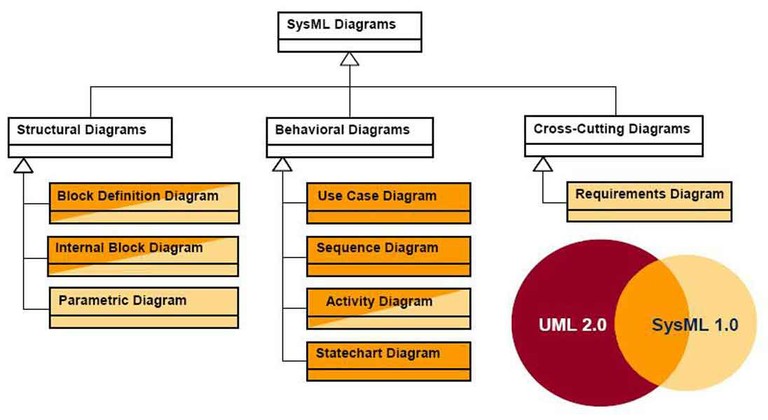


Figure - Les diagrammes comportementaux de SysML (source [1])

## Diagramme de cas d’utilisation

D’après [6] :

* Les cas d’utilisation permettent de décrire **ce que fait le système**, sans spécifier comment il le fait
* Un acteur est un utilisateurs humain ou un d’autre système qui interagit directement avec le système étudié
* Un cas d’utilisation (use case, ou UC) représente un ensemble de séquences d’actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier.
* Les cas d’utilisation peuvent être reliés par des relations :
  + D’inclusion
  + D’extension
  + De généralisation

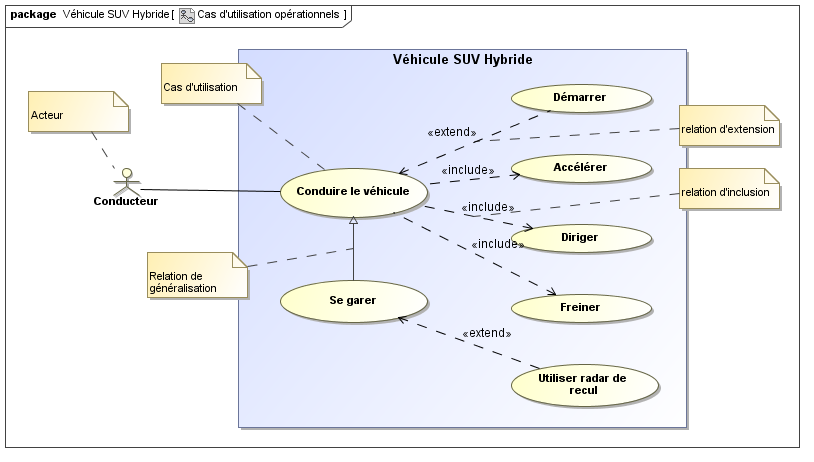


Figure : cas d'utilisation opérationnels du véhicule SUV Hybride

## Diagramme de séquence

D’après [6] :

* Le diagramme de séquence montre la séquence verticale des **messages** passés entre **éléments** (lignes de vie) au sein d’une interaction
* La **ligne de vie** représente l’existence d’un élément participant dans un diagramme de séquence
* Un **message**, représenté par une flèche, déclenche une **activité** dans le destinataire



Figure : Notation de base du diagramme de séquence (source [6])

Les diagrammes de séquence peuvent inclure des « **fragments combinés** ». Les fragments combinés permettent notamment :

* Des références à d’autres séquences (type « ref »)
* Des alternatives (type « alt »)
* Des exécutions parallèles (type « par »)

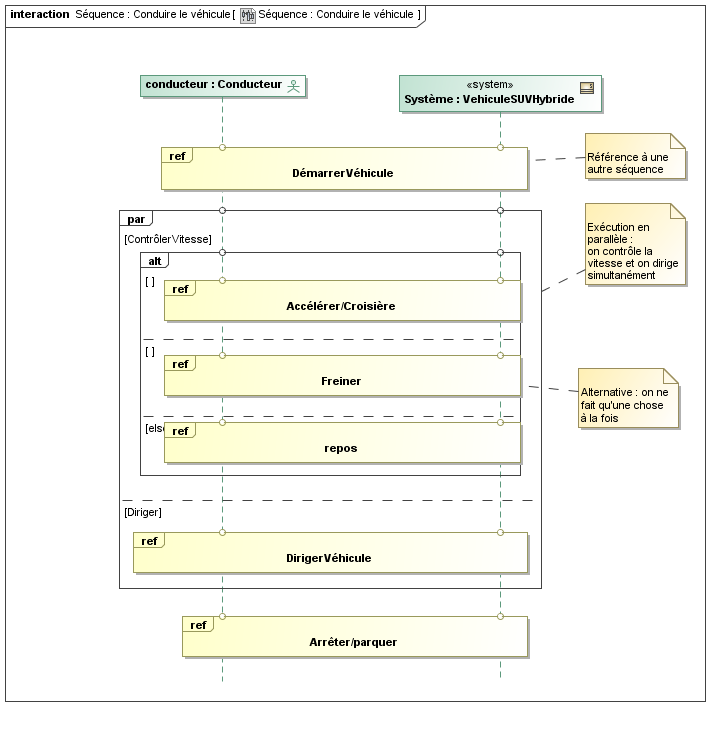


Figure : Diagramme de séquence utilisant des fragments combinés

## Diagramme d’état

D’après [5] :

Le diagramme d’état décrit comment un **bloc** réagit à des **événements** en fonction de son **état** courant et comment il passe dans un nouvel état.

* Un **état** représente une situation durant la vie d’un **bloc** pendant laquelle :
  + il satisfait une certaine condition
  + il exécute une certaine activité
  + ou bien il attend un certain événement.
* Une **transition** décrit la réaction d’un bloc lorsqu’un **événement** se produit

|  |  |
| --- | --- |
| Figure  : diagramme d’états simple | Figure  : conditions dans un diagramme d’état |

* Une **condition** (ou condition de garde) est une expression booléenne qui doit être vraie lorsque l’événement arrive pour que la transition soit déclenchée
* En plus de la succession d’états « normaux » correspondant au cycle de vie d’un bloc, le diagramme d’états comprend également deux pseudo-états :
  + **l’état initial** du diagramme d’états correspond à la création d’une instance ;
  + **l’état final** du diagramme d’états correspond à la destruction de l’instance.
* Un **état composite** (aussi appelé super-état) permet d’englober plusieurs sous-états exclusifs.

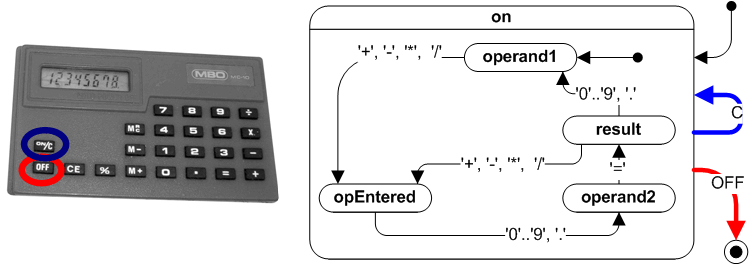


Figure : diagramme d'état d'une calculette – utilisation d’un état composite ON (source [4])

## Diagramme d’activité

Le diagramme d’activité ressemble au diagramme d’états (D.E.), mais au lieu d’états il organise des activités, ou actions.

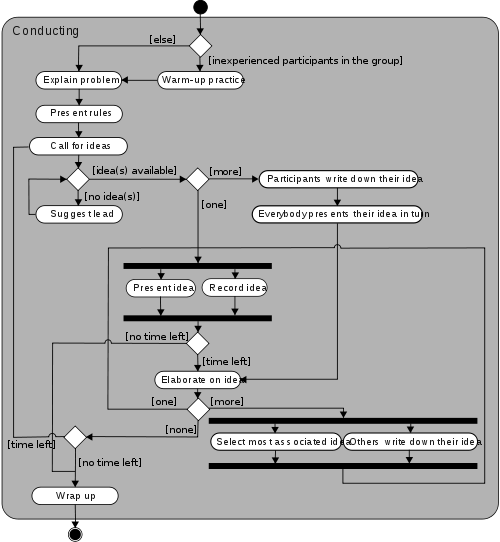


Figure – Diagramme d’activité (source [5])

## Utilisation en Sti2D

* Outils de modélisation :
  + En général beaucoup trop complexes pour envisager une utilisation par les élèves.  
    Bien que le niveau taxonomique soit 3, les outils existants nous semblent inadaptés.
* Diagramme de cas d’utilisation :
  + Le choix des C.U. effectué l’analyste système est subjectif. A ce niveau, les élèves doivent être uniquement lecteurs.
  + Un travers fréquent est de faire de la décomposition fonctionnelle avec les C.U. (attention à rester centré sur les besoins et non sur les fonctionnalités internes).
* Diagramme de séquence :
  + L’utilisation des fragments combinés est déconseillée en première approche.
  + Considérer qu’un diagramme représente uniquement un scénario possible, et ne pas chercher à représenter tous les scénarios sur un unique diagramme
* Diagrammes d’activités :
  + la richesse de la syntaxe est très importante. Se limiter à des enchaînements simples de tâches.
  + Risque fort de confusion avec le diagramme d’état. A éviter si possible (préférer le diagramme d’état).

## Références

1. OMG SyML - <http://www.omgsysml.org/>
2. SyML France - <http://www.sysml-france.fr/>
3. Introduction à SysML – Olivier Servat – Académie d’Orléans-Tours  
   <http://www.jampez.net/index.php?option=com_content&view=article&id=66:intro-sysml&catid=43:category-modelisation&Itemid=28>
4. Wikipedia – UML state machine (en Anglais) : <http://en.wikipedia.org/wiki/UML_state_machine>
5. Wikipedia – UML activity diagram : <http://en.wikipedia.org/wiki/UML_activity_diagram>
6. SysML par l’exemple – Pascal Roques – Ed. Eyrolles 2009 - ISBN : 978-2-212-85006-2