

Cours	<b>MODELISATION DE SYSTEMES</b>	S.I.I.
Fonctionnalité, architecture et structure des systèmes	<b>AVEC LE LANGAGE SYSML</b>	CI 1

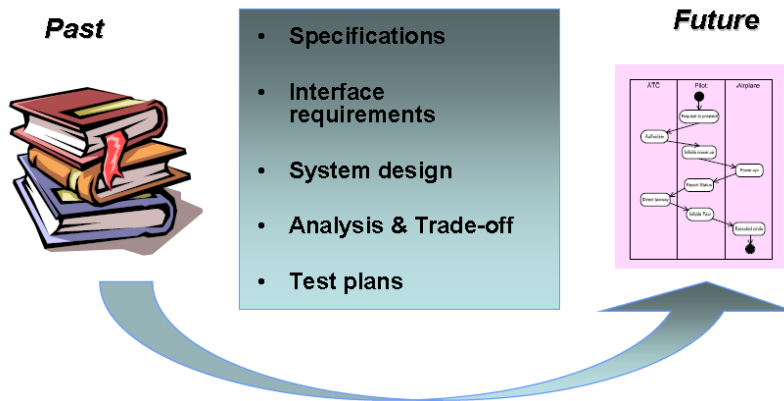
<b>1) INTRODUCTION A L'INGENIERIE SYSTEME.....</b>	<b>2</b>
<b>2) DIFFERENTES ETAPES DU CYCLE DE VIE D'UN SYSTEME.....</b>	<b>3</b>
<b>3) LE LANGAGE SYSML. ....</b>	<b>5</b>
31) DIAGRAMME TRANSVERSAL : .....	5
311) <i>Diagramme d'exigences (SysML Requirements Diagram)</i> .....	5
32) DIAGRAMMES COMPORTEMENTAUX : .....	6
321) <i>Diagramme de contexte</i> .....	6
322) <i>Diagramme de cas d'utilisation (SysML Use Case Diagram)</i> .....	6
323) <i>Diagramme de séquence (SysML Sequence Diagram)</i> .....	7
324) <i>Diagramme d'états / transitions (SysML State Machine Diagram)</i> .....	8
33) DIAGRAMMES STRUCTURELS : .....	8
331) <i>Diagramme de définition de bloc (SysML Block Definition Diagram)</i> .....	8
332) <i>Diagramme de bloc interne (SysML Internal Block Diagram)</i> .....	9
34) DEFINITIONS DES DIFFERENTS TYPES D'ASSOCIATIONS.....	9
35) DEFINITIONS DES RELATIONS DANS LES DIAGRAMMES. ....	9
<b>4) EXEMPLE : SECATEUR ELECTRIQUE PELLENC.....</b>	<b>10</b>
41) ANALYSER LE BESOIN .....	10
42) ETABLIR LE CAHIER DES CHARGES FONCTIONNEL (C.D.C.F.).....	10
43) CONCEVOIR LE SECATEUR. ....	12

# 1) Introduction à l'Ingénierie Système.

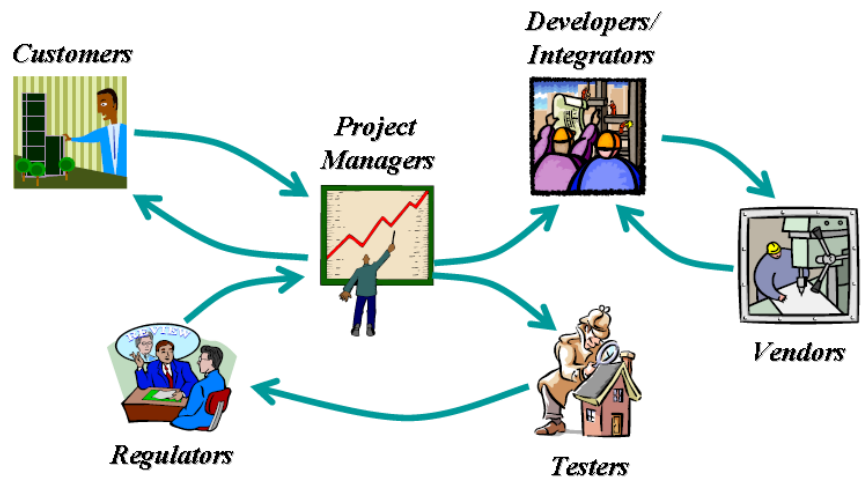
L'ingénierie des systèmes est une approche scientifique interdisciplinaire de formation récente, dont le but est de formaliser et d'appréhender la conception de systèmes complexes avec succès.

L'ingénierie des systèmes se focalise sur la définition des besoins du client et des exigences fonctionnelles, détectés tôt dans le « **cycle de vie du système** », en documentant les exigences, puis en poursuivant avec la synthèse de la conception et la validation du système.

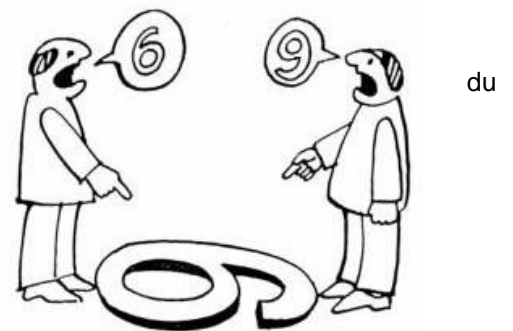
La conception de système donne souvent lieu à une accumulation de documentations qui doivent toutes être croisées et mises à jour pour maintenir la cohérence et respecter les spécifications du système.



Le langage SysML est un moyen de regrouper dans un modèle commun à tous les corps de métiers, les spécifications, les contraintes, et les paramètres de l'ensemble du système.



Il permet ainsi d'éviter tout problème de communication qui engendrerai retard dans l'élaboration du produit.



## 2) Différentes étapes du cycle de vie d'un système.

La notion de « cycle de vie » est indissociable d'un système. Elle exprime les différentes étapes qui vont de l'analyse du besoin jusqu'à l'élimination et/ou le recyclage de ses constituants.



- Analyser le besoin.

« Un besoin est une nécessité ou un désir éprouvé par un utilisateur » (NF X 50-150).

Le client est sensible à l'évolution du contexte économique, social et environnemental ainsi qu'au degré d'innovation, le besoin évolue donc constamment.

Exemple : le vélo à assistance électrique.



EMI/Philips de 1935/37  
simplex



ISD City 3 – 2004  
Batterie 36 V



VAE Tidal Force IO-Cruiser – Matra -2010  
Moteur Matra Brushless DC  
dans le moyeu arrière  
Batterie Ni-MH 36 V 9 Ah  
dans moyeu avant



VAE du futur.... ???  
– Concept Peugeot –



- Etablir le Cahier des Charges Fonctionnel (C.d.C.F.) :

Avant d'imposer une solution, il faut se tourner vers le demandeur, pour aboutir de manière structurée à la solution. En effet, le but d'un projet est de satisfaire le besoin. Il faut exprimer clairement les objectifs à atteindre d'un projet, afin d'éviter toute confusion entre vous et le demandeur.

Ce besoin doit être exprimé dès le lancement du projet. Il s'agit d'explicitier quelle est l'exigence fondamentale qui justifie la conception du produit.

Le cahier des charges fonctionnel est un document qui permet de formaliser avec précision le besoin du demandeur. En effet, le C.d.C.F. est un tableau de bord qui définit le projet et détaille les conditions dans lesquelles il doit être réalisé. C'est le lien de compréhension entre l'entreprise et le client.

La partie technique d'un C.d.C.F. doit se limiter à *énumérer les contraintes techniques avérées*.

Les contraintes de base sont économiques (les contraintes monétaires comme le budget de fonctionnement), environnementales (le caractère recyclable du produit, etc.), humaines (par exemple, dans le cas d'un jouet pour enfant, il doit être léger, ne pas contenir de petites pièces, etc.), industrielles (par exemple, il doit être fabriqué au Canada) et matérielles (par exemple, il doit spécifier les morceaux qui peuvent être remplacés, comme des piles, il doit préciser le recours à tel ou tel serveur d'applications).

- Concevoir :

A partir du cahier des charges fonctionnel, le bureau d'études modélise le produit sous forme de maquette numérique.

- Matières premières :

Le choix des matières premières est important dans le cycle de vie puisqu'il conditionne le prix du produit. Il faut également penser au recyclage du produit donc aux choix des matériaux.

- Industrialiser :

L'industrialisation est le processus de fabrication de produits avec des techniques permettant une forte productivité du travail.

- Homologuer :

L'homologation est la certification conforme d'un produit à une norme, ou une réglementation. En d'autres termes l'homologation garantit au consommateur que le produit qu'il achète correspond à ce qu'il est en droit d'en attendre.

- Transporter :

Le transport du produit est un point important dans l'analyse du cycle de vie du produit.

- Commercialiser, utiliser le produit :

C'est là que l'on peut constater si le client est satisfait par le produit.

- Eliminer/Recycler :

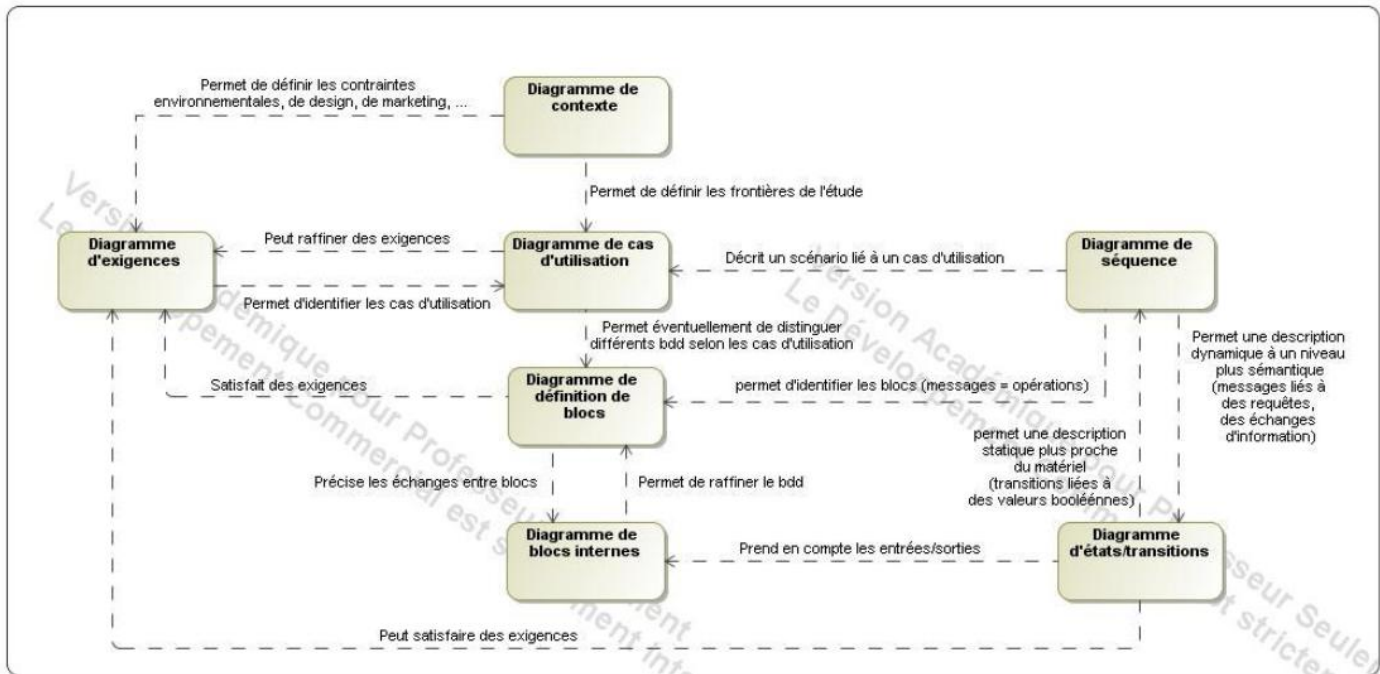
L'objectif actuel est de recycler un maximum de composants mais beaucoup de matériaux ne peuvent pas être recyclés.

### 3) Le langage SysML.

Le langage **SysML** - **S**ystems **M**odeling **L**anguage - est un langage de modélisation permettant de décrire tout ou partie d'un système technique, d'un point de vue transversal, comportemental ou structurel.



Le langage **SysML** s'articule autour de neuf types de diagrammes, seuls sept diagrammes seront étudiés dans ce cours, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système.



### 31) Diagramme transversal :

#### 311) Diagramme d'exigences (SysML Requirements Diagram)

- Rôle :

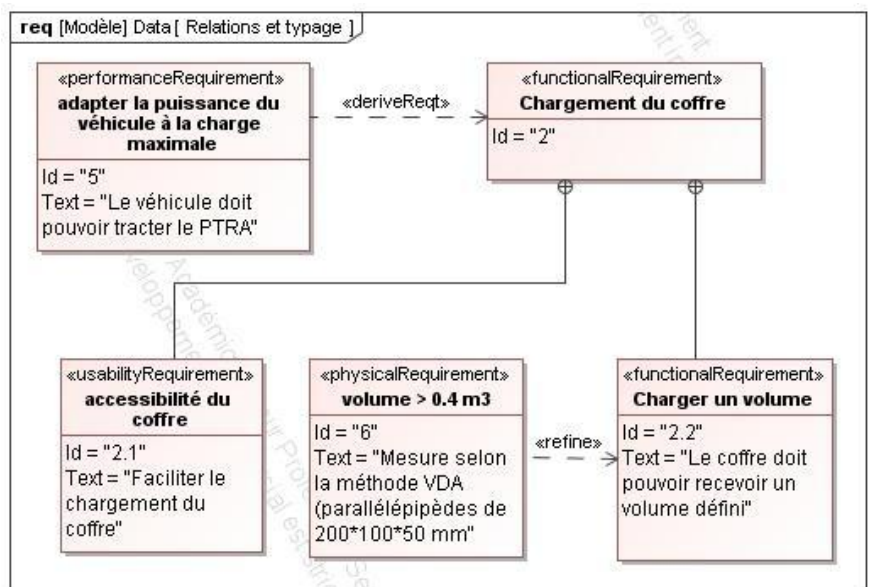
Représente toutes les exigences du système.

- Exigences environnementales,
- Exigences économiques,
- Exigences fonctionnelles,
- Exigences techniques,
- etc.

- Limites et préconisation :

Ne pas chercher à poser toutes les exigences sinon illisible.  
Réaliser plusieurs diagrammes d'exigences si nécessaire.

- Regrouper les exigences techniques sur un seul diagramme par exemple.
- puis les autres groupes d'exigences sur d'autres diagrammes.



## 32) Diagrammes comportementaux :

### 321) Diagramme de contexte.

- Rôle :

Il permet de définir les frontières de l'étude, et en particulier de préciser la phase du cycle de vie dans laquelle on situe l'étude (généralement la phase d'utilisation).

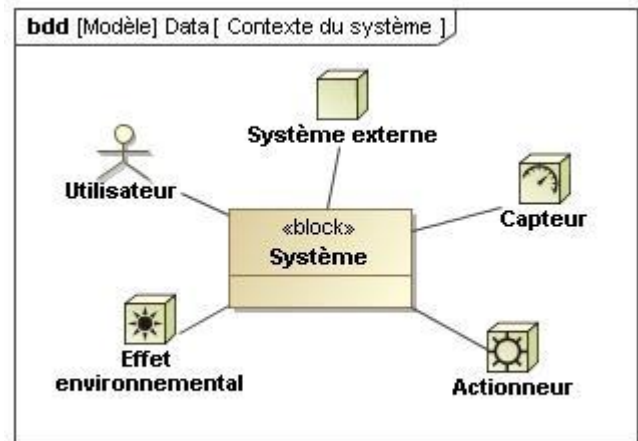
Il répond à la question : "Quels sont les acteurs et éléments environnants du système ?".

- Limites et préconisation :

Ce diagramme devra bien sûr faire apparaître tous les acteurs intervenants dans le diagramme de cas d'utilisation, mais il fera aussi apparaître les différents acteurs ou éléments intervenant dans une exigence.

Il n'y a aucune recommandation spécifique sur la manière dont il sera établi. Il pourra se faire par :

- Une carte mentale,
- Un bdd (diagramme de définitions de blocs) SysML (le plus souvent),
- Un ibd (diagramme de blocs internes) SysML.



### 322) Diagramme de cas d'utilisation (SysML Use Case Diagram)

- Rôle :

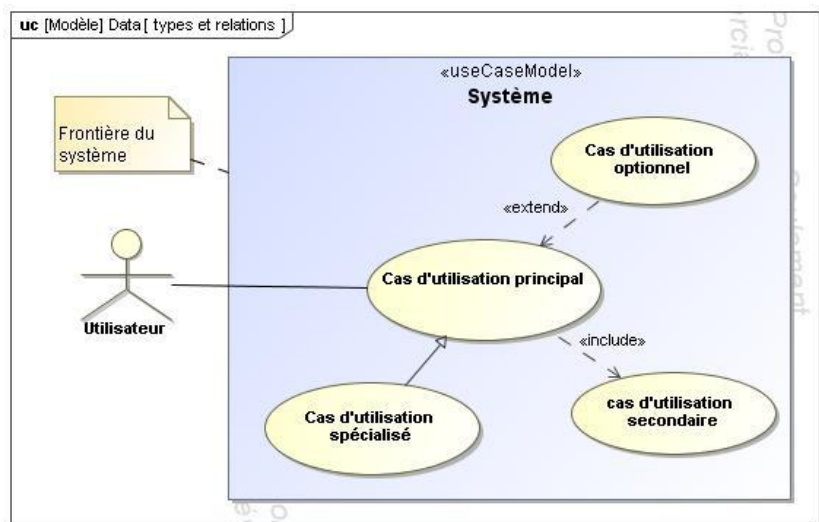
Montre les fonctionnalités offertes par le système. Il répond à la question : "quels services rend le système ?".

Fonctionnalité = cas d'utilisation = service rendu en autonomie d'un bout à l'autre par le système.

Le résultat est visible par l'acteur (entité extérieure en interaction avec le système).

- Limites et préconisation :

Ce diagramme devra bien sûr faire apparaître tous les acteurs intervenants dans le diagramme de cas d'utilisation, mais il fera aussi apparaître les différents acteurs ou éléments intervenant dans une exigence.





### 323) Diagramme de séquence (SysML Sequence Diagram).

- Rôle :

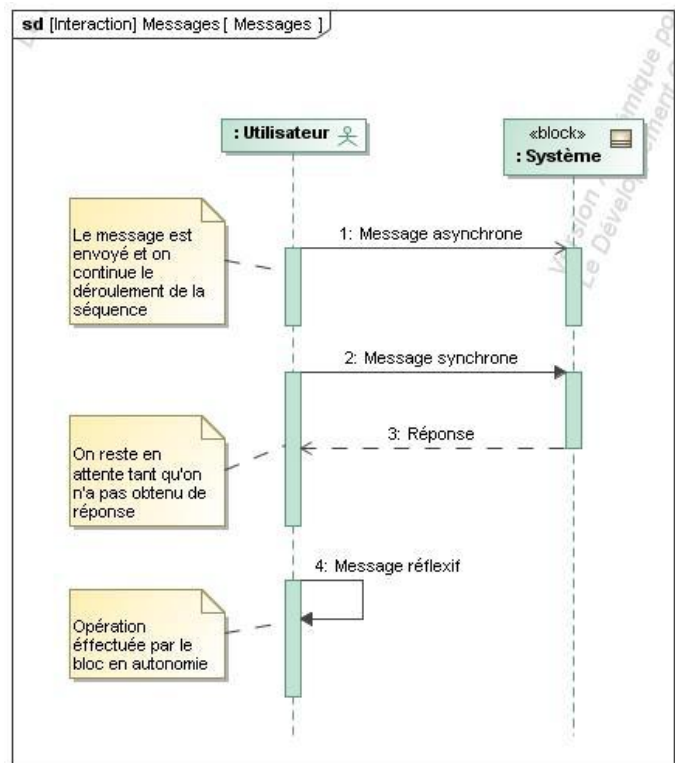
Décrit les scénarios correspondant aux cas d'utilisation, un cas d'utilisation est décrit par au moins un diagramme de séquence.

Il répond à la question : "Comment est réalisé ce cas d'utilisation ?".

Montre les interactions entre différents éléments d'un point de vue séquentiel, enchaînement et nature des échanges.

Dans un premier temps il est préférable de faire un diagramme de séquence « système », celui-ci étant vu comme une boîte noire.

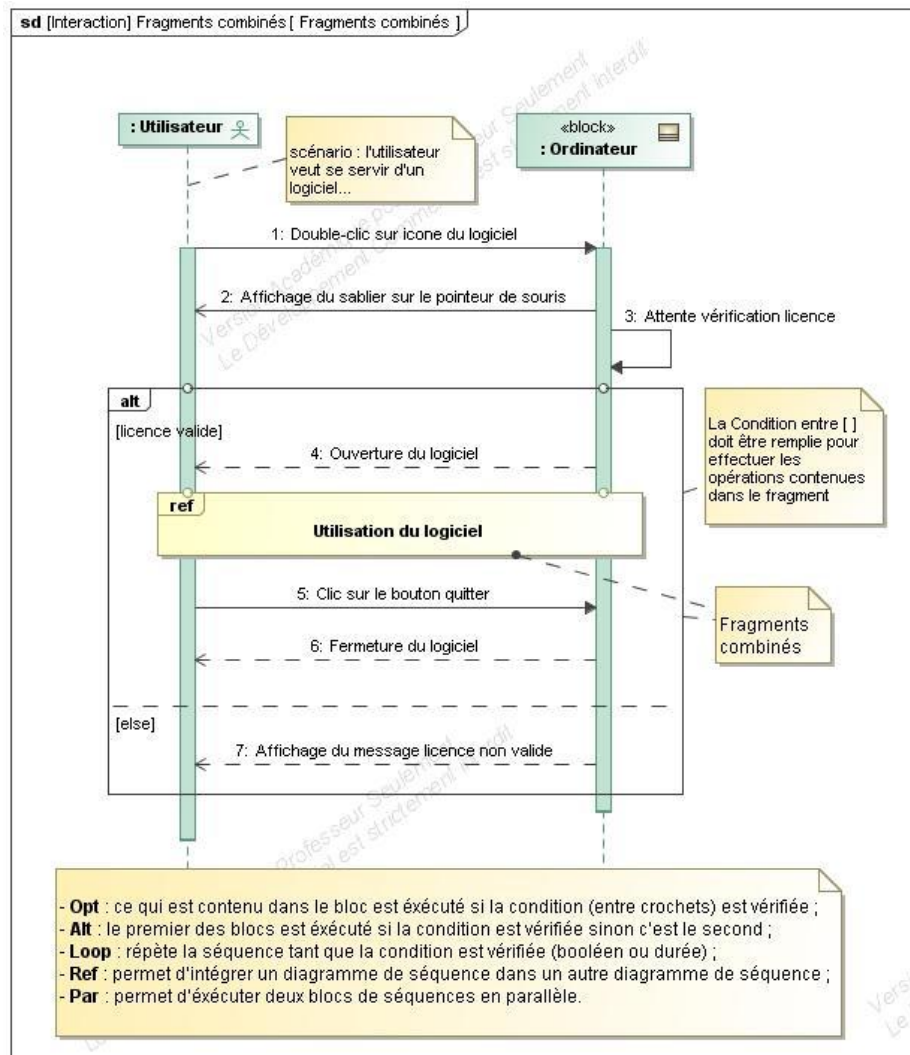
Pour des détails vous pouvez ensuite montrer les interactions au sein du système (décomposé en ses différents éléments).



- Limites et préconisation :

Il existe les **fragments combinés** pour montrer des variantes dans un scénario. Cela doit être utilisé avec parcimonie car les diagrammes de séquence ne sont pas des algorithmes.

Un scénario se décrit dans un cadre bien précis. Vous aurez donc des scénarios de réussite et des scénarios d'échecs (gestion des problèmes).



### 324) Diagramme d'états / transitions (SysML State Machine Diagram).

- Rôle :

Décrit le comportement d'un programme sous forme de machine d'états.

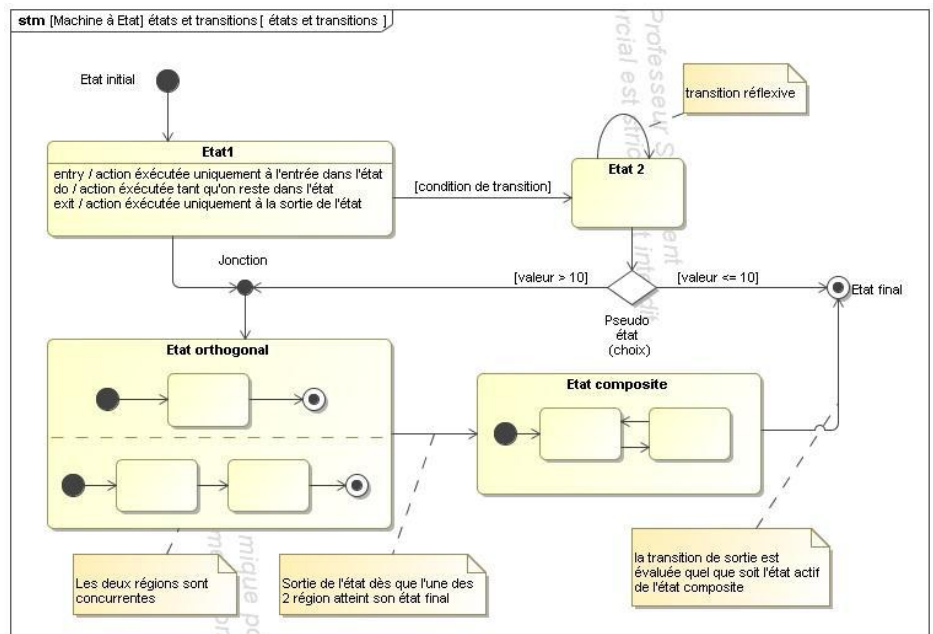
Montre les différents états pris par le système (ou un sous-système) en fonction des interactions

Il répond à la question : "Comment représenter les différents états du système ?"

- Limites et préconisation :

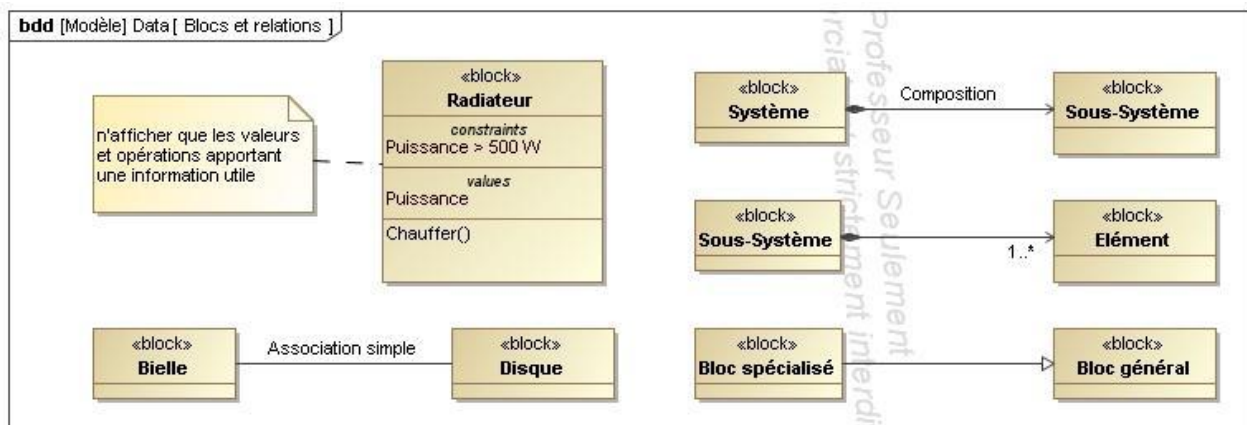
Ce diagramme servira avant tout à décrire le fonctionnement d'un programme.

Ce diagramme trouve toute sa place en lien avec les logiciels de simulation comportementale (Matlab, LabView, Modelica,...) .



### 33) Diagrammes structurels :

#### 331) Diagramme de définition de bloc (SysML Block Definition Diagram).



- Rôle :

Montre le système d'un point de vue composé/composant

Il répond à la question "qui contient quoi ?".

Il peut aussi montrer les caractéristiques principales de chaque bloc en faisant apparaître les opérations (rôles) et les propriétés (caractéristiques).

Permet de représenter les liens entre les blocs de même niveau par une association (simple trait entre 2 blocs).

- Limites et préconisation :

La question du zoom est importante. Même si on peut descendre assez bas dans les détails, il ne sera pas pertinent en général de le faire. Ce diagramme est utile pour montrer les grosses briques du système.

Il n'est pas obligatoire de faire apparaître les propriétés et les opérations dans chaque bloc.

Dans ce cas le diagramme est relativement pauvre en informations, mais il offre d'un coup d'œil la structure du système.



### 332) Diagramme de bloc interne (SysML Internal Block Diagram).

- Rôle :

Permet de représenter les échanges de matière/information/énergie entre blocs de même niveau grâce aux ports de flux (petit carré avec une flèche).

Permet de représenter les services invoqués par un autre bloc grâce aux ports standards (petit carré sans flèche), et par extension toute entrée/sortie de contrôle/commande.

Permet de représenter les liens entre les blocs de même niveau.

- Limites et préconisation :

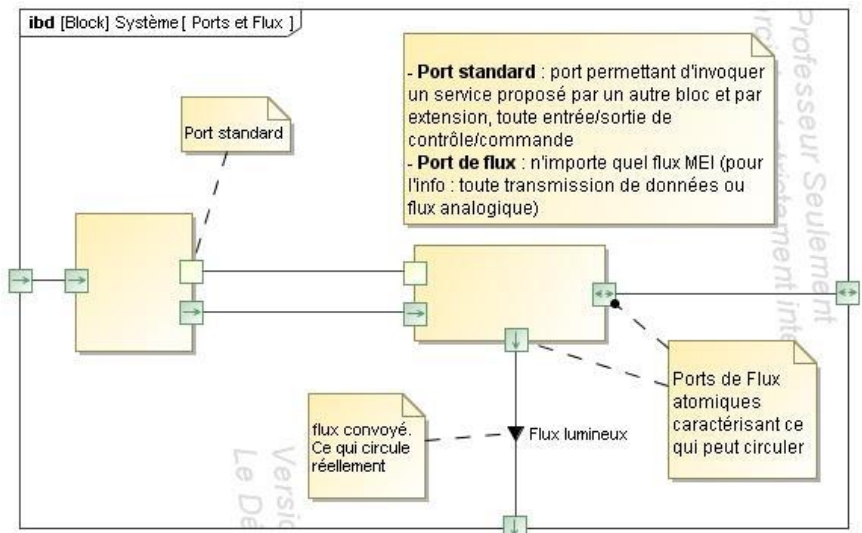
Il faut bien retenir que les liens se représentent entre blocs de même niveau, ils ne se contiennent pas.

Chaque bloc du BDD contenant d'autres blocs peut être représenté par un IBD.

Attention à bien faire la différence entre port de commande et port de flux.

**Port standard** : désigne une interface permettant d'invoquer un service/une opération

**Port de flux** : canal d'Entrée/Sortie par lequel transite de la matière, de l'énergie ou de l'information (MEI).



### 34) Définitions des différents types d'associations.

**Extend** : le cas d'utilisation source est une extension possible du cas d'utilisation destination.

**Include** : le cas d'utilisation source comprend obligatoirement le cas inclus.

**Derive** : une ou plusieurs exigences sont dérivées d'une exigence.

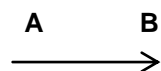
**DeriveReq** : permet de relier une exigence d'un niveau général à une exigence d'un niveau plus spécialisée mais exprimant la même contrainte.

**Satisfy** : un ou plusieurs éléments du modèle permettent de satisfaire une exigence.

**Verify** : un ou plusieurs éléments du modèle permettent de vérifier et valider une exigence.

**Refine** : un ou plusieurs éléments du modèle redéfinissent une exigence.

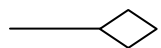
### 35) Définitions des relations dans les diagrammes.



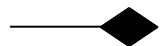
**Association** : relation d'égal à égal entre 2 éléments. **A utilise B**  
- 3 diagrammes : cas d'utilisation, définition des blocs, blocs internes.



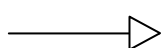
**Dépendance** : l'un des deux éléments dépend de l'autre. **A dépend de B**  
- 3 diagrammes : cas d'utilisation, exigences, définition des blocs.



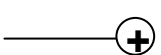
**Agrégation** : Un élément est une composante facultative d'un autre. **A entre dans la composition de B sans être indispensable à son fonctionnement.**  
- 2 diagrammes : exigences, définition des blocs.



**Composition** : Un élément est une composante obligatoire de l'autre. **A entre dans la composition de B et lui est indispensable à son fonctionnement.**  
- 2 diagrammes : exigences, définition des blocs.



**Généralisation** : Dépendance de type 'filiation' entre 2 éléments. **A est une sorte de B.**  
- 3 diagrammes : exigences, définition des blocs, blocs internes.



**Conteneur** : relation d'inclusion entre 2 éléments. **B contient A**  
- 2 diagrammes : exigences, définition des blocs.

## 4) Exemple : Sécateur électrique PELLENC.

Pour appréhender le langage SysML, nous nous appuyerons sur un exemple concret : **un sécateur électrique**.

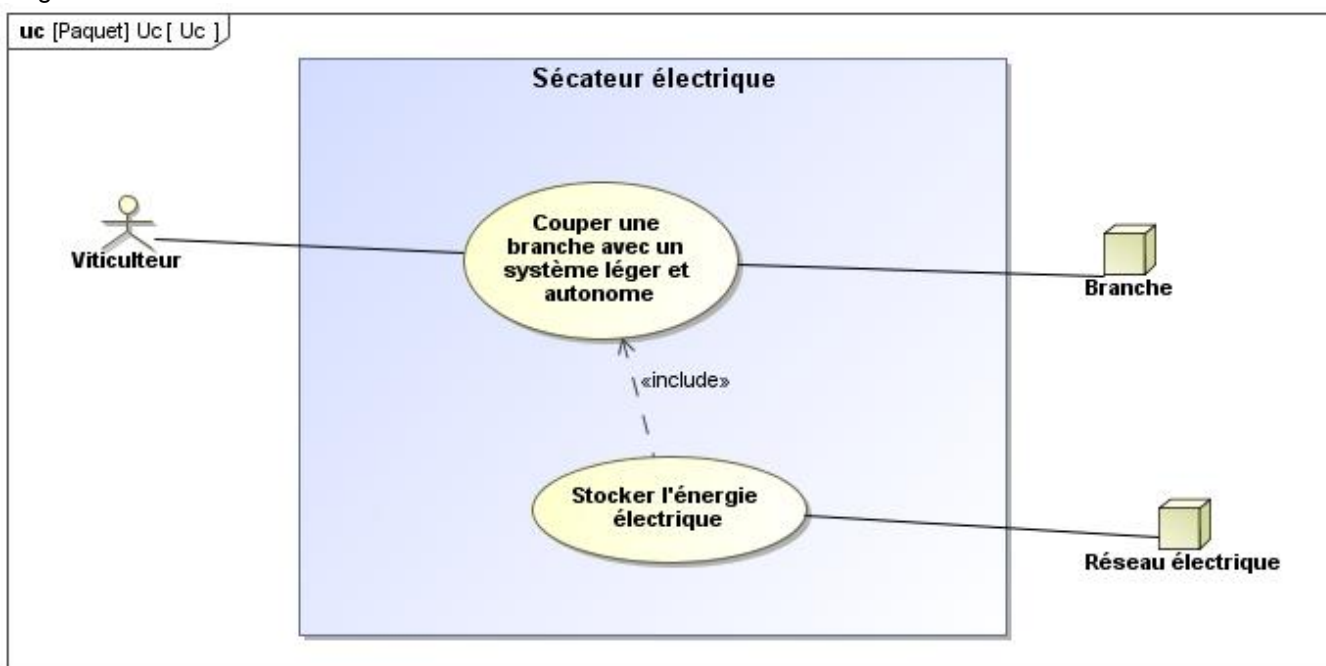
### 41) Analyser le besoin

La période de la taille de la vigne dure environ 2 mois. Les viticulteurs coupent 8 à 10 heures par jour.

*Pour réduire la fatigue de la main et du bras, la société PELLENC commercialise un sécateur électrique à commande électronique. Ce système se compose d'une valise contenant la partie commande PC (portée sur le dos de l'utilisateur) et alimentant un sécateur (tenu par la main de l'utilisateur) par un câble.*



Diagramme de cas d'utilisation :

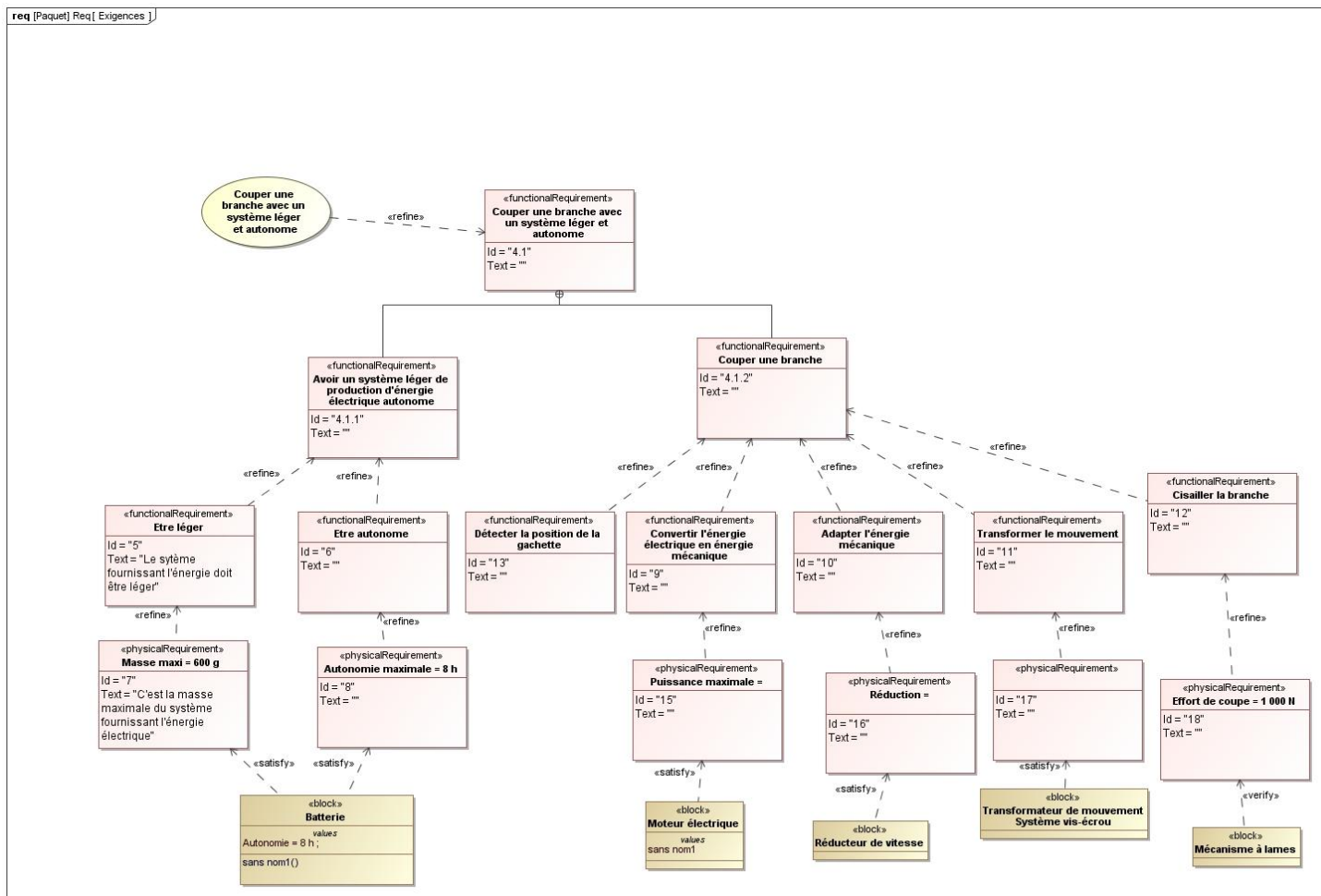


### 42) Etablir le Cahier des Charges Fonctionnel (C.d.C.F.).

Il est extrait du tableau des exigences (SysML Requirements Table).

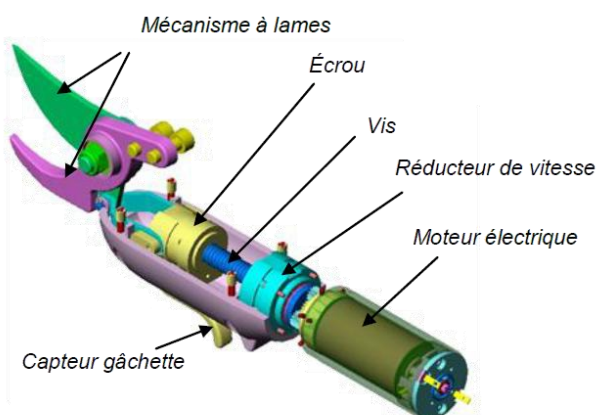
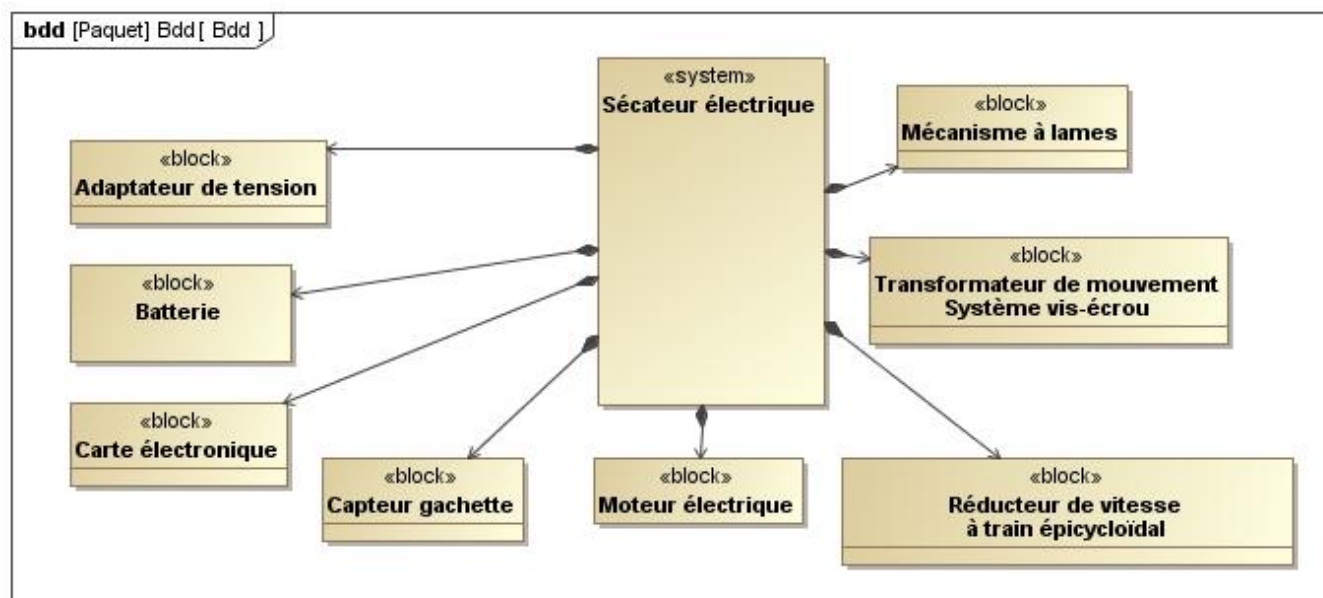
#	ID	Name	Text
1	4.1.2	Couper une branche	Diamètre maximal de la branche = 22 mm; Effort de coupe = 1 000 N; Cadence de coupe = 60 coupes/min
2	6	Etre autonome	Durée d'autonomie = 8 h
3	4.1.3	Etre intégralement portable	Masse totale < 3,5 kg

## Diagramme d'exigences :



### 43) Concevoir le sécateur.

- Diagramme de définition de bloc



- Diagramme de bloc interne

