

CI-1 : ANALYSER ET DÉCRIRE LES SYSTÈMES INDUSTRIELS

CI-1-1 : DÉCRIRE UN SYSTÈME AVEC L'INGÉNIERIE SYSTÈME

Objectifs

ANALYSER - COMMUNIQUER

A l'issue de la séquence, l'élève doit être capable :

- **A1** Analyser le besoin et les exigences
- **A3** Analyser l'organisation fonctionnelle et structurelle
- **A4** Analyser les performances et les écarts
- **B1** Choisir les grandeurs physiques et les caractériser
- **B3** Valider un modèle
- **E1** Rechercher et traiter des informations

Table des matières

1 Relation Client-Entreprise	2
1.1 Problématique	2
1.2 Entreprise	2
1.3 Performance	2
1.4 Systèmes complexes	2
2 Cahier des Charges Fonctionnelles (CdCF)	3
2.1 Pourquoi faire?	3
2.2 Définition et composition	3
3 Ingénierie Système	4
3.1 Pourquoi l'Ingénierie Système?	4
3.2 Définition	4
3.3 Processus de conception	4
3.4 La modélisation	5
4 SysML	6
4.1 SysML : Pourquoi?	6
4.2 SysML : Qu'est-ce?	6
4.3 SysML : ses avantages	7
4.4 Eléments graphiques des diagrammes	7
5 Analyse et description des systèmes à l'aide de SysML	8
5.1 Besoin du client - Cahier des Charges	8
5.2 Diagramme de définition de blocs (bdd)	8
5.3 Notions de classes, instances et parties	9
5.4 Diagramme de contexte (pas d'identifiant)	10
5.5 Diagramme des cas d'utilisation (uc)	10
5.6 Diagramme des exigences (req)	11
5.7 Diagramme de blocs internes (ibd)	12

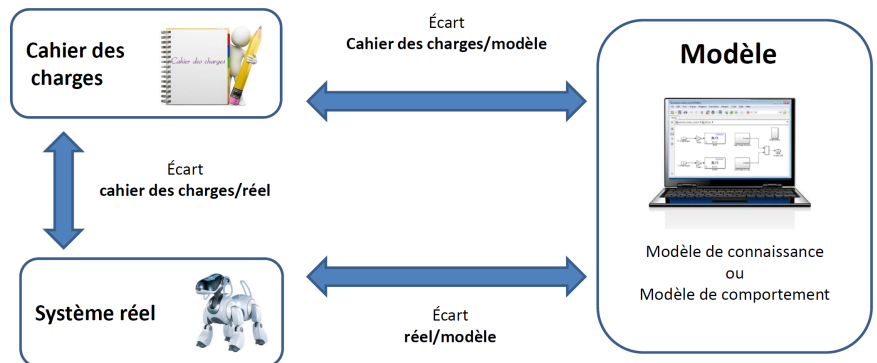
1 Relation Client-Entreprise

1.1 Problématique

PROBLÉMATIQUE :

Pour survivre, une entreprise doit réussir à **satisfaire ses clients**. Ce dernier est donc au cœur du processus de conception.

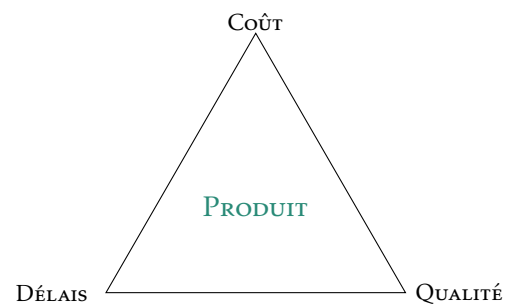
En effet, à cause de la concurrence qui peut proposer des produits similaires, le rôle de l'entreprise est donc de **minimiser l'écart entre le besoin du client et la prestation ou le produit réalisé** :



1.2 Entreprise

Pour satisfaire le besoin du client mais aussi pour survivre et/ou croître, elle doit veiller à :

- un contrôle des **coûts** :
 - d'étude: études de marchés, conception, simulation, conception des outillages
 - d'industrialisation: achat des machines, fabrication...
 - de commercialisation: publicité, transport...
 - d'élimination: destruction, stockage, recyclage...
- une bonne **qualité** du produit
- des **délais** faibles afin d'assurer une bonne réactivité



1.3 Performance

L'intégration des nouvelles technologies dans les nouveaux produits impose de nouvelles approches (dans la conception, la fabrication, la commercialisation). L'ensemble de ces activités d'ingénierie est confié aux chercheurs, *ingénieurs*, techniciens qui **imaginent, conçoivent et réalisent des systèmes modernes pour répondre aux besoins** toujours en évolution des consommateurs.

L'étude de ces systèmes modernes, dans le cadre de l'entreprise, se conduit selon plusieurs points de vue :

- fonctionnel : Quelle fonction le produit remplit-il ?
- structurel : Comment se constitue le produit (composants et constituant) ?
- temporel : Quelles sont les évolutions du comportement du produit au cours du temps ?
- économique : Quels sont les aspects économiques soumis au produit ?

1.4 Systèmes complexes

Un système est un ensemble de composants qui collaborent à la réalisation d'un ensemble de tâches en vue de fournir un ensemble de services. Cet ensemble est soumis à un environnement donné et interagit ainsi avec un sous-ensemble des éléments de cet environnement.

Un système est dit complexe lorsque les inter-relations liant les composants sont multiples, interdépendantes et bouclées : le comportement global n'est donc pas directement prévisible à partir des comportements élémentaires des composants.

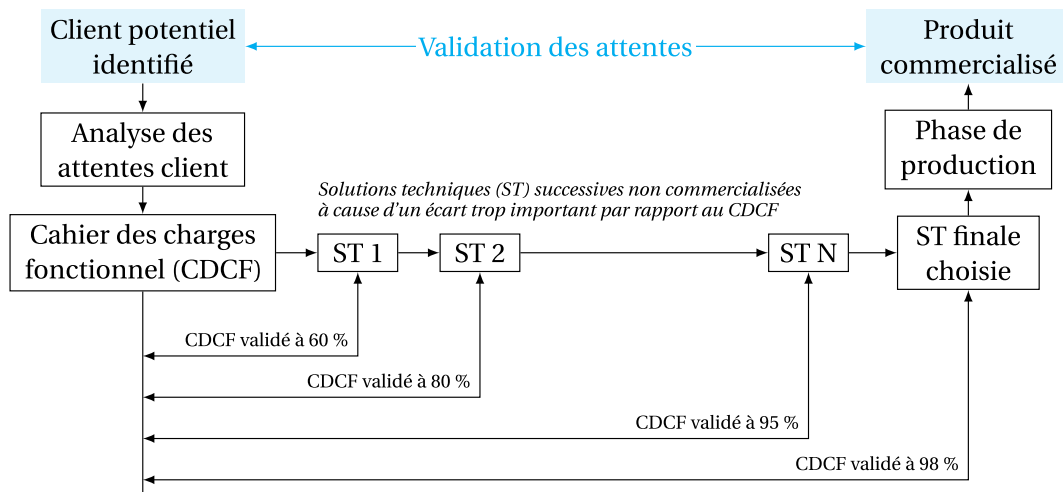
L'analyse des systèmes est un point central en Sciences de l'Ingénieur aux concours. Il n'est pas que demandé de restituer des connaissances apprises au cours de l'année. L'évaluation portera sur les capacités à :

- **présenter** un système réel dans son contexte et dans sa globalité,
- **mobiliser** vos connaissances pour analyser le comportement du système complexe,
- **proposer** des solutions vis-à-vis de problèmes techniques.

Concrètement, il s'agit de faire preuve d'**esprit de synthèse** pour présenter le système et les résultats, d'**esprit d'initiative** et de **créativité** lors des manipulations du système, de proposer des modèles simples.

2 Cahier des Charges Fonctionnelles (CdCF)

2.1 Pourquoi faire ?



La **satisfaction des attentes du client** ou services attendus par celui-ci constitue la **finalité** du processus de conception et par conséquent du système. Une fois le système réalisé, le client juge de sa satisfaction des services attendus de manière subjective grâce au ressenti procuré par ses cinq sens. Une entreprise ne peut pas engager un processus de conception sur des bases subjectives qu'elle ne maîtrise pas pour aboutir à un système qui satisfera de manière très aléatoire le client.

La maîtrise du processus de conception passe par une **description objective du but** à atteindre. Cette description objective se nomme le **Cahier des Charges Fonctionnel (CdCF)**. L'**aspect fonctionnel** du cahier des charges permet de contraindre le processus de conception en **termes de finalités** et **non pas en termes de solutions**. Ceci conduit, en particulier, à produire des systèmes concurrents en termes de services rendus mais dont les solutions techniques sont très différentes.

2.2 Définition et composition

DÉFINITION : Cahier des Charges [IEEE Std 1220-1994 et INCOSE 1996]

Document identifiant une performance, une caractéristique physique, ou un niveau de qualité, définissant un produit, ou un procédé, pour lesquels une action sera développée.

FONCTIONS DE SERVICE	CRITÈRES	NIVEAUX	FLEXIBILITÉS
FS1 : Orienter la tuyère par rapport au corps central	Précision	Ecart nul pour les réponses impulsionnelle et indicielle	0
	Débattement angulaire	$\pm 60^\circ$	$\pm 3^\circ$
	Vitesse de déplacement	$t_{5\%} \leq 0.15 \text{ s}$	1
	Sensibilité aux résonances	fréquences propres $\neq 20 \text{ Hz}$	2
	Stabilité	Marge de gain $\geq 6 \text{ dB}$ Marge de phase $\geq 45^\circ$	0 1

3 Ingénierie Système

3.1 Pourquoi l'Ingénierie Système ?

Face à la complexification de notre environnement, et en regard des objectifs nouveaux que l'on cherche à atteindre, les systèmes à faire fonctionner, comme les organisations pour les réaliser et les exploiter, deviennent plus complexes.

De plus, des contraintes de tous ordres à prendre en compte s'accroissent. La maîtrise des systèmes apparaît donc de plus en plus comme un enjeu majeur, tant pour les organismes que pour les nations, dans leurs interactions ou coopérations.

Cette approche, très ancienne dans sa démarche, est devenue nécessaire avec l'accroissement de la complexité des systèmes. Elle a été formalisée de manière rigoureuse à partir de 2001 par les normes IEEE1220, EIA632 et ISO15288.

3.2 Définition

L'ingénierie système (IS) est une démarche méthodologique pour maîtriser la conception des systèmes et produits complexes. On peut aussi la définir ainsi :

DÉFINITION : Ingénierie Système

Approche interdisciplinaire rassemblant tous les efforts techniques pour faire évoluer et vérifier un ensemble intégré de systèmes, de gens, de produits et de solutions de processus de manière équilibrée au fil du cycle de vie pour satisfaire aux besoins client.

Les pratiques de cette démarche sont aujourd'hui répertoriées dans des normes, réalisées à l'aide de méthodes et supportées par des outils.

3.3 Processus de conception

En Ingénierie Système la définition du système comporte :

- celle de ses sous-systèmes et constituants (matériels, logiciels, organisations et compétences humaines) et de leurs interfaces, sièges des interactions recherchées.
- celles des processus de leurs cycles de vie permettant de les concevoir, produire, vérifier, distribuer, déployer, exploiter, maintenir en condition opérationnelle et retirer du service, et donc des produits contributeurs nécessaires à ces processus.

Cette approche de la définition induit une démarche descendante d'ingénierie s'appuyant sur une décomposition itérative du système en blocs constitutifs. Les constituants sont alors définis avec leurs interfaces ainsi que les produits contributeurs à leur cycle de vie.

L'aspect pluri-technique de tels systèmes implique :

- la participation de spécialistes de cultures différentes : il faut donc des outils de communications communs. Le langage SysML¹ est un de ses outils (ce n'est pas une méthode).
- les délais de conception courts nécessitent un travail parallèle des équipes, ce qui rend difficile la mise en place d'une organisation du travail efficace.
- les inter-relations entre les composants et les performances à atteindre nécessitent une adaptation permanente des paramètres.

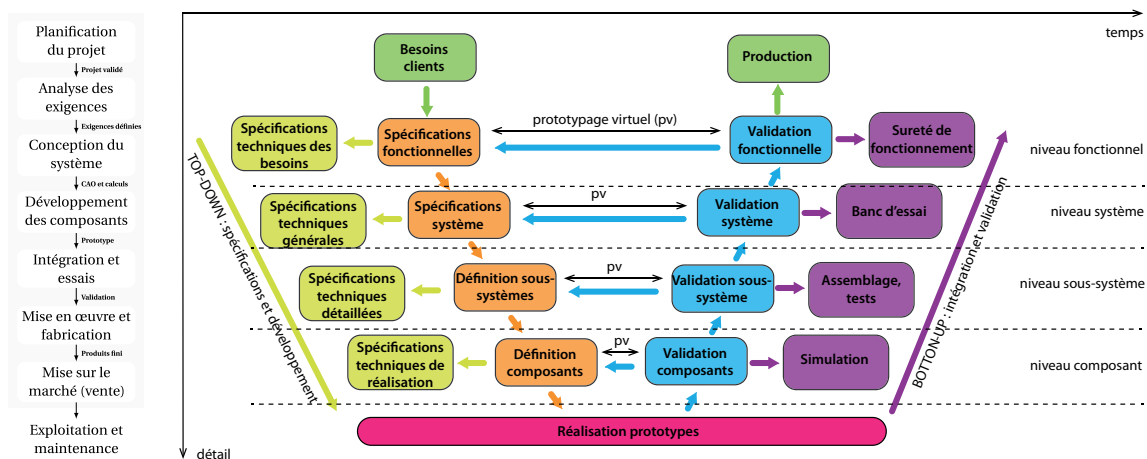
Une méthode classiquement répandue dans le milieu industriel est le **cycle en V**. Le cycle en V décline deux phases dans la conception :

- Dans la **phase descendante (à gauche)**, le problème global est morcelé en sous problèmes et des choix technologiques sont proposés, ce qui aboutit à la définition de chaque composant.
- Dans la **phase ascendante (à droite)**, la solution technique précise est vérifiée à l'aide de calculs numériques ou d'essais expérimentaux, d'abord localement puis au sein d'ensemble plus globaux, jusqu'au produit final.

À chaque étape, si les tests de validation sont négatifs, il y a itération, c'est-à-dire modification des paramètres de la solution technique et test à nouveau.

Cette démarche permet de diviser le système complexe en sous-composants, en définissant clairement le périmètre de chaque composant et ses contraintes vis-à-vis de son environnement.

Il est ainsi possible aux équipes de travailler en parallèle au niveau inférieur du V et assurer la cohérence dans les phases d'intégration des composants car **les itérations aux niveaux hauts sont beaucoup plus coûteuses que les itérations aux niveaux bas**.



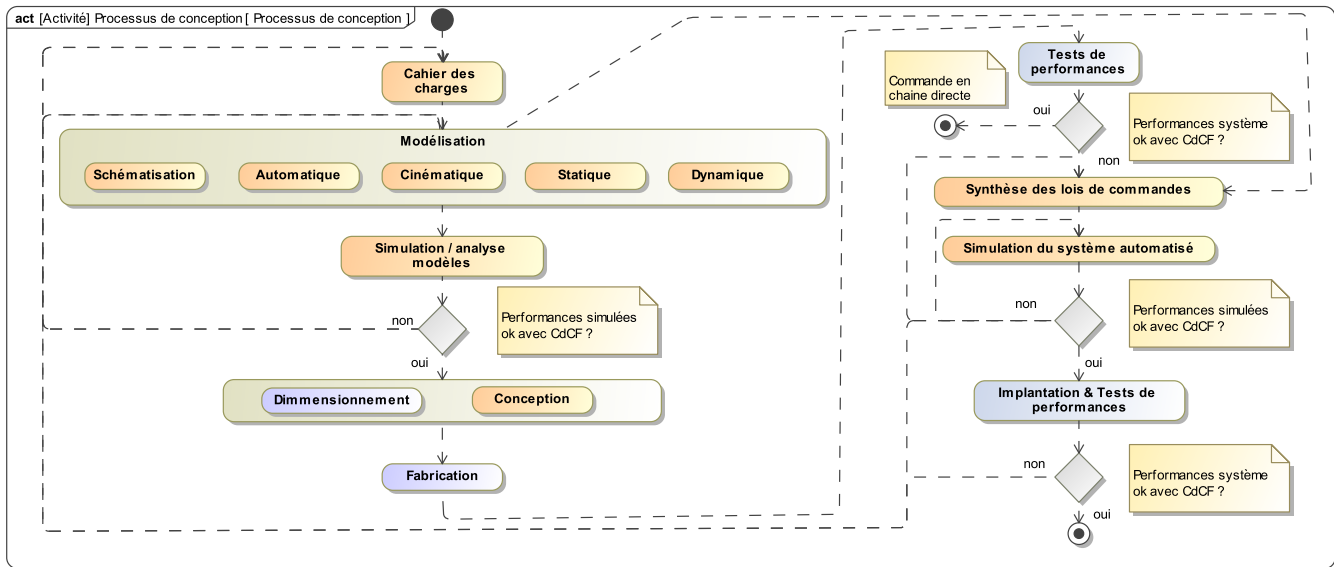
Processus de conception en cascade et en V

3.4 La modélisation

La transformation d'un besoin émergent en la définition d'un système lui apportant une solution met en œuvre de multiples activités intellectuelles faisant passer progressivement de concepts abstraits à la définition rigoureuse de produits.

1. Systems Modeling Language

Il est nécessaire de s'appuyer sur des représentations tant du problème que de ses solutions possibles à différents niveaux d'abstraction pour **appréhender, conceptualiser, concevoir, estimer, simuler, valider, justifier des choix, communiquer**. C'est le rôle de la **modélisation**.



Les métiers mis en œuvre en Ingénierie Système ont, de tous temps, utilisé des modèles allant de représentations des plus concrètes, telles que les plans ou modèles réduits, aux plus abstraites, telles que les systèmes d'équations.

Dans un souci d'uniformisation des représentations, le langage SysML a été créé.

4 SysML

4.1 SysML : Pourquoi ?

Dans un système complexe, les flux de matière, d'énergie ou d'information (MEI) échangés entre les composants, les relations orientées ou non et les bouclages ne permettent pas de décrire un système simplement sous la forme d'un texte ou d'un discours et l'utilisation d'un **support graphique** devient rapidement **indispensable**.

L'idée est donc née au début des années 2000 d'**unifier les langages de modélisation**.

Cette réflexion a donné naissance au **langage de modélisation des systèmes SysML**² (Systems Modeling Language) qui a une structure standardisée depuis septembre 2007 (version 1.0a, la dernière en date étant la 1.3 de juin 2012).

4.2 SysML : Qu'est-ce ?

Basé sur le langage UML³, SysML est un langage de modélisation pour l'Ingénierie Système. Il prend en charge la spécification, l'analyse, la conception, la vérification et la validation des systèmes qui comprennent le matériel, les logiciels, les données, le personnel, les procédures et les installations.

C'est un langage de modélisation qui fournit :

2. Systems Modeling Language
3. Unified Modeling Language

- **une sémantique** : qui donne une signification et une relation entre les signes et leurs référents
- **une notation** : qui est un ensemble de signes conventionnels qui servent à fixer par écrit leur interprétation.

SysML permet d'approcher un modèle par des vues (fenêtre ayant un angle de vision déterminé). Une vue est un élément du modèle. Trois points de vue ont été privilégiés dans le langage SysML :

- Un point de vue **comportemental**, auxquels sont associés quatre diagrammes :
 - Le diagramme des cas d'utilisation (*Use Case Diagram*, indicateur **uc** ou **ucd**)
 - Le diagramme de séquence (*Sequence Diagram*, indicateur **seq**)
 - Le diagramme d'états (*State Machine Diagram*, indicateur **stm**)
 - Le diagramme d'activités (*Activity Diagram*, indicateur **act**)
- Un point de vue **structurel**, auxquels sont associés quatre diagrammes :
 - Le diagramme de définition de blocs (*Block Definition Diagram*, indicateur **bdd**)
 - Le diagramme de bloc interne (*Internal Block Diagram*, indicateur **ibd**)
 - Le diagramme paramétrique (*Parametric Diagram*, indicateur **par**)
 - Le diagramme de paquetages (*Package Diagram*, indicateur **pkg**)
- Un point de vue **transversal**, spécifique au langage SysML, a été rajouté : le diagramme d'exigences (*Requirement Diagram*, indicateur **req**)

Ce n'est pas une méthode, il n'y a pas obligatoirement d'ordre dans l'établissement des diagrammes. Cependant, il y a tout de même une manière *naturelle* de procéder.

4.3 SysML : ses avantages

Grâce à la richesse de sa notation, le langage SysML permet :

- l'expression des **besoins** et des **contraintes** ;
- la représentation de l'**organisation structurée** des composants et leur définition précise ;
- la représentation des **modes de fonctionnement**, des processus internes et externes au système ainsi que les interactions avec son environnement.

Sa structure autorise également des analyses très intéressantes pour les concepteurs telles que :

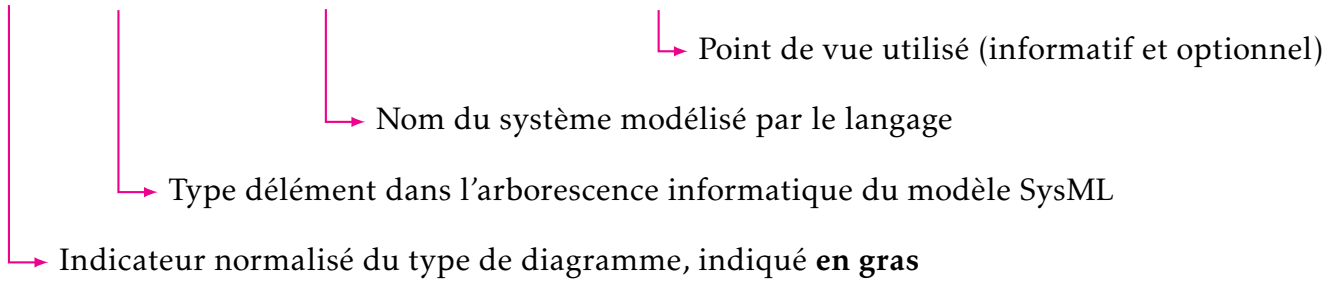
- la facilitation de la **collaboration de tous les spécialistes** des corps de métier concernés, en proposant un ensemble lié d'**outils de représentation universels et expressifs** ;
- la réalisation de la **mise à jour**, du **stockage** et du **partage** ainsi que l'interprétation des informations issues des analyses des travaux des différents intervenants ;
- l'**intégration** et la **mise en relation** des différentes composantes techniques, par exemple les liaisons entre un programme informatique et des actionneurs mécaniques ;
- la modélisation du système à toutes les étapes de son **cycle de développement** et dans sa **phase de vie** en représentant les éléments du modèle selon différents points de vue ;
- la **validation des différentes solutions** par une ou plusieurs simulations basées sur les diagrammes d'états, d'activités et paramétrique présentés dans la suite.

4.4 Éléments graphiques des diagrammes

4.4.1 Cadre du diagramme

Chaque diagramme SysML représente un élément particulier du modèle selon un certain point de vue. Afin de le repérer, chaque diagramme comporte un **cartouche**, positionné sur la partie supérieure gauche du cadre.

req [Paquet] Exigences Drone [Point de vue production]



Le type de diagramme, repéré par son identifiant (**req**, **bdd**, **ibd**, **pkg**, **par**, **uc**, **seq**, **stm**, **act**), est obligatoirement indiqué dans ce cartouche. L'ajout des autres éléments est optionnel selon la norme mais les logiciels dédiés à la modélisation par ce langage ajoutent par défaut le type d'élément (bloc ou paquet par exemple) et proposent d'indiquer le nom du modèle mis en place et le point de vue utilisé.

4.4.2 Formes géométriques et liens

Les neuf diagrammes du langage SysML sont composés des mêmes types de formes géométriques : des rectangles à coins droits ou arrondis, des ellipses et des lignes. Selon les diagrammes, tout ou partie de ces formes géométriques seront utilisées.

Plusieurs types de relations sont représentés par des formes géométriques dans les diagrammes SysML. A force de côtoyer les documents, vous apprendrez à les reconnaître.

5 Analyse et description des systèmes à l'aide de SysML

5.1 Besoin du client - Cahier des Charges

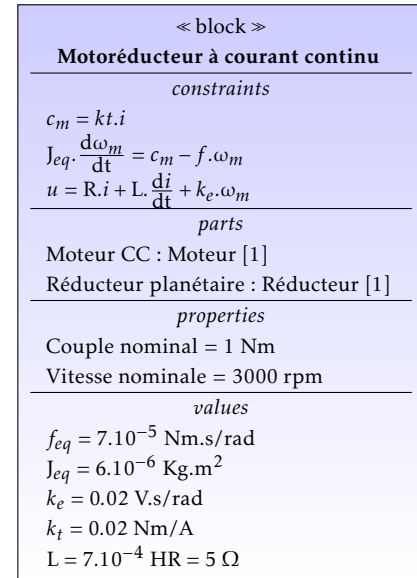
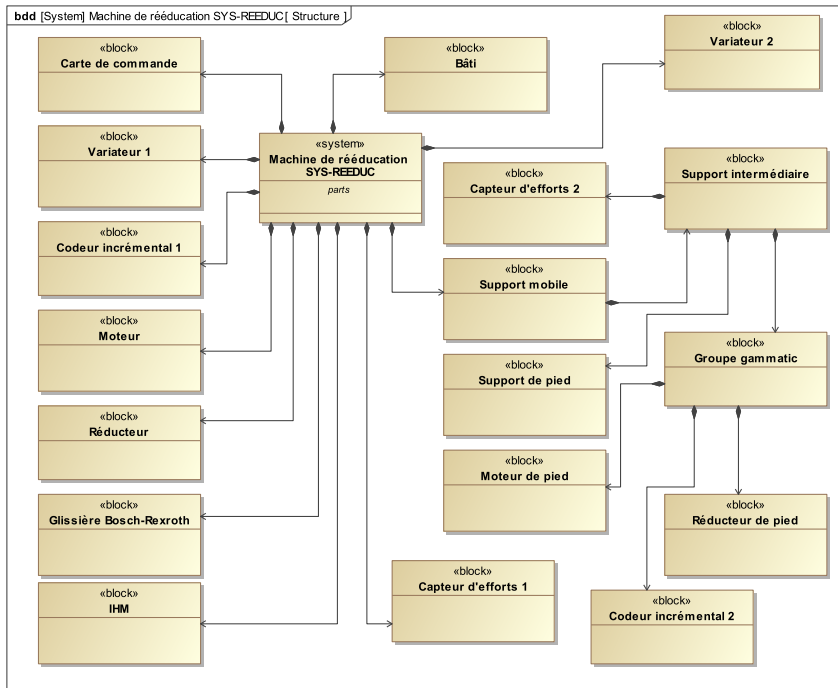
Le système doit répondre (entre autres) aux exigences précisées dans les tableaux suivants :

Exigences	Critères	Niveaux
Permettre au kinésithérapeute de rééduquer les membres inférieurs du patient	Angle de rotation de la cuisse Effort du patient Écart de position Rapidité	De 0° à 150° Jusqu'à 20 N. Nul $T_{5\%} < 0,2 \text{ s.}$
S'adapter à la morphologie des patients	Longueur de la cuisse et jambe Écartement du bassin Distance plat du pied – cheville	De 0,6 à 1,2 m. 370 à 600 mm.
Ne pas blesser le patient	Sécurité	Bloquer le fonctionnement en fonction de la taille du patient

5.2 Diagramme de définition de blocs (bdd)

OBJECTIF : Décrire le système via des blocs (blocks dans le langage SysML) et représenter des éléments matériels (cas le plus fréquent) mais également des entités abstraites (regroupement logique d'éléments) ou des logiciels.

Ce diagramme représente les caractéristiques principales de chaque bloc ainsi que les liens entre eux : il permet donc une modélisation de l'architecture du système. Graphiquement, un bloc est représenté par un rectangle avec le stéréotype « block » comprenant un titre et des compartiments étagés regroupant des propriétés particulières.



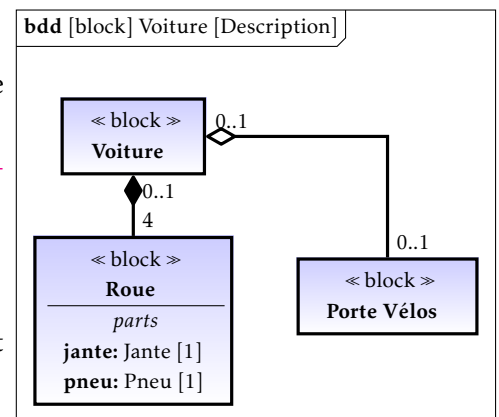
Il est ensuite possible de relier les blocs au moyen de liens dont la sémantique dépend de la nature particulière de la relation.

Sur ces liens, il est possible de préciser la multiplicité d'un bloc en plaçant une valeur au bout du lien.

Prenons maintenant l'exemple d'une voiture. Le diagramme ci-contre se lit :

- une voiture est composée de 4 roues obligatoirement (**composition**) et éventuellement d'un porte vélo (**agrégation**)
- une roue est associée à 0 ou 1 voiture.
- un porte vélo est associé à 0 ou 1 voiture.

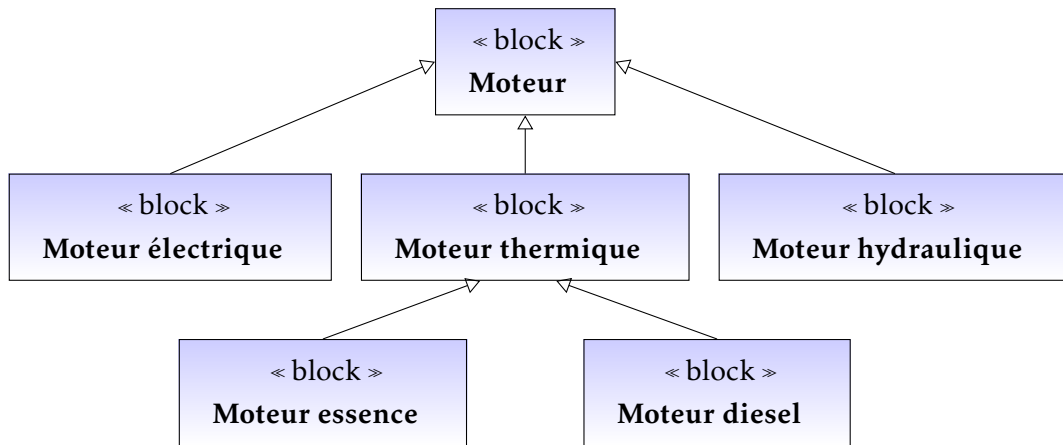
Nous noterons que le bloc « Roue » s'écrit au singulier, en effet, il s'agit de la classe « Roue ».



5.3 Notions de classes, instances et parties

Une classe permet donc de décrire un ensemble d'objets (ou instances de classe) ayant des caractéristiques communes. Les objets quant à eux possèdent un état et un comportement.

Le but est souvent de factoriser des propriétés communes (valeurs, parties, etc.) à plusieurs blocs dans un bloc généralisé. Les blocs spécialisés **héritent** des propriétés du bloc généralisé et peuvent comporter des propriétés supplémentaires.



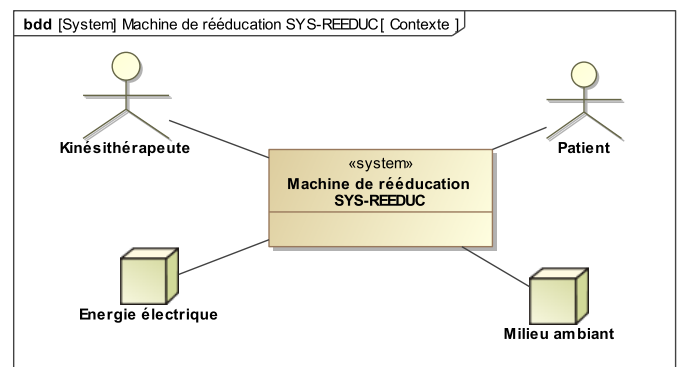
Ce format est l'instanciation de la partie. Il est cependant fréquent de ne pas le respecter pour ne garder que le nombloc.

5.4 Diagramme de contexte (pas d'identifiant)

OBJECTIF : Préciser, si possible de manière exhaustive, les acteurs et éléments environnants au système étudié. Il permet également de faire apparaître les différents acteurs ou éléments intervenant dans une exigence.

Le diagramme de contexte est une extension non normalisée du langage SysML qui permet de définir les frontières de l'étude et la phase du cycle de vie dans laquelle on situe l'étude (il s'agit généralement de la phase d'utilisation normale du système).

De par sa position d'extension, il n'y a absolument aucune recommandation spécifique sur la manière dont ce diagramme sera établi : classiquement, on utilise un diagramme de définition de blocs.



5.5 Diagramme des cas d'utilisation (uc)

OBJECTIF : Montrer les fonctionnalités offertes par un système en identifiant les services qu'il rend.

DÉFINITION : Acteur

|| Humain ou un autre système interagissant directement avec le système.

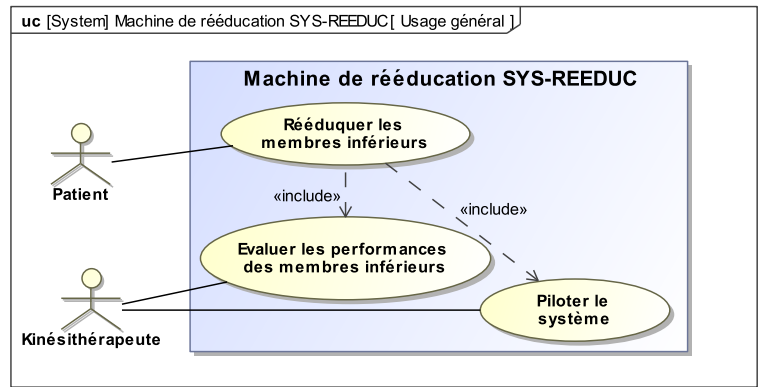
Un acteur participe à au moins un cas d'utilisation. On distingue deux types d'acteurs :

- l'acteur principal : il est associé à la fonctionnalité principale du système qui justifie son existence, qui répond au besoin.
- l'acteur secondaire : il est associé à des fonctionnalités complémentaires.

Dans la pratique, les **acteurs principaux** sont placés sur la **gauche** du diagramme et les **acteurs secondaires** sur la **droite**.

Les fonctionnalités d'un système correspondent à des cas d'utilisation, c'est-à-dire à des services rendus par le système.

Il n'apparaîtra donc pas ce qui ne peut être fait par des acteurs extérieurs : ainsi, par exemple, le lavage, la recharge, le recyclage, la réparation, etc. ne doivent pas apparaître si le système n'a pas été développé expressément pour cela.



DÉFINITION : Frontière d'un système

Représentée par un cadre. Le nom du système figure à l'intérieur du cadre, en haut. Les acteurs sont à l'extérieur et les cas d'utilisation à l'intérieur.

Pour affiner le diagramme de cas d'utilisation, SysML définit trois types de relations standardisées entre cas d'utilisation :

- une relation d'**inclusion**, formalisée par le mot-clé **include** \dashrightarrow : le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire.
- une relation d'**extension**, formalisée par le mot-clé **extend** \dashrightarrow : le cas d'utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifié indirectement dans celui qui procède à l'extension (appelé extension point).
- une relation de **généralisation/spécialisation** \dashrightarrow : les cas d'utilisation descendants héritent la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des interactions spécifiques supplémentaires.

5.6 Diagramme des exigences (req)

