

# Émergence des exigences par une approche MBSE avec SysML

Olivia PENAS & Régis PLATEAUX

# Sommaire

2

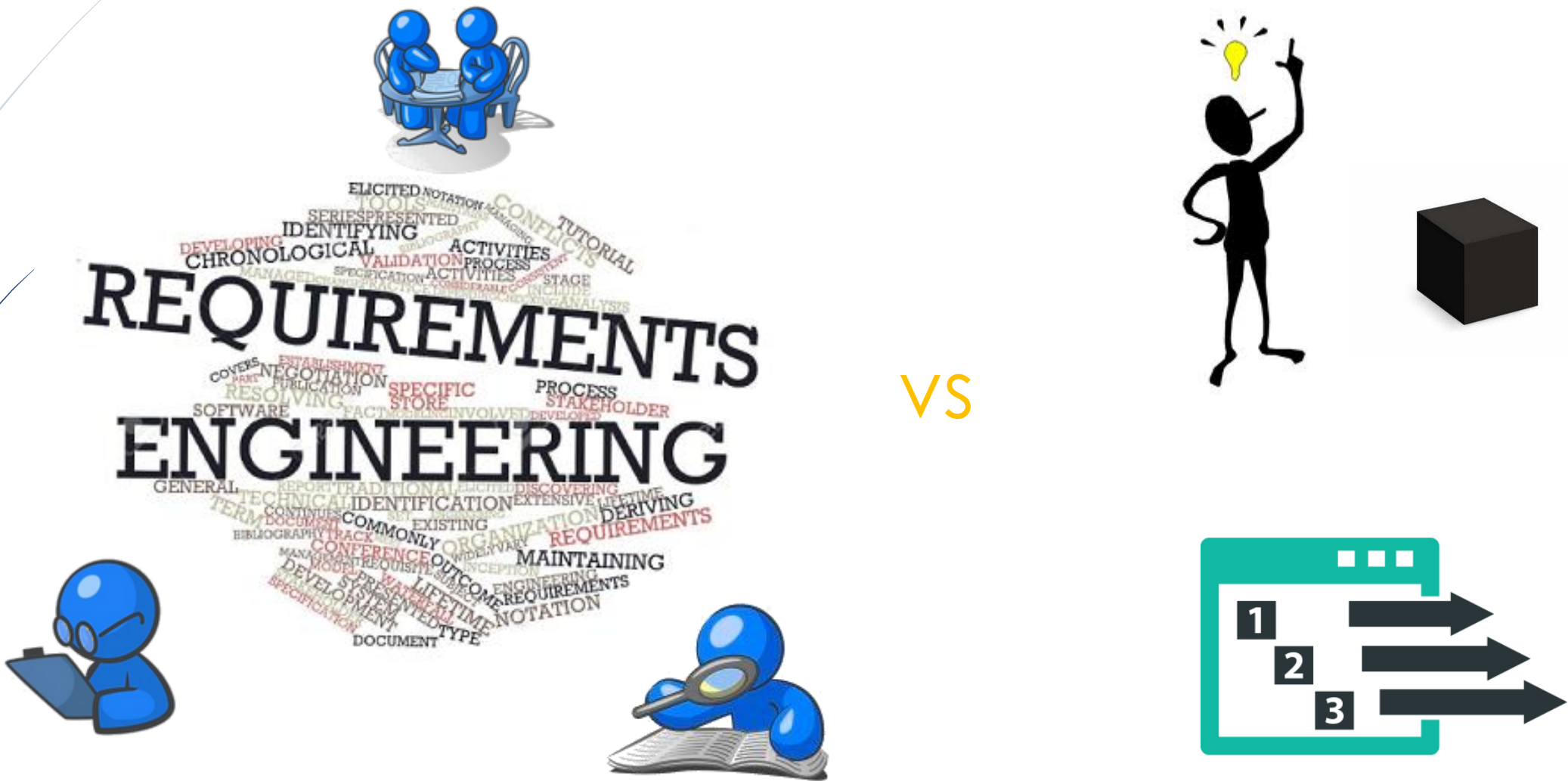
- Positionnement
- Contexte et origine
- Vue globale de la méthodologie
- Point de vue externe
- Emergence des exigences
- Formalisation de la démarche



3

# Positionnement

# Positionnement vs Ingénierie des exigences





5

# Contexte & origine

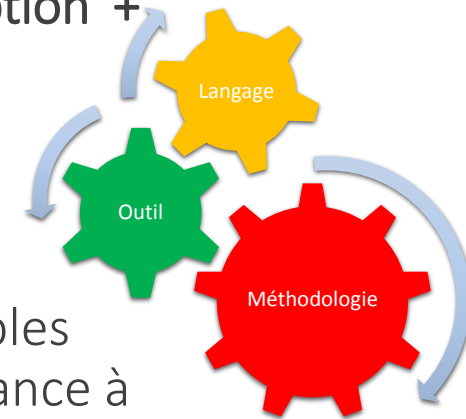
# Contexte

6

- Les entreprises doivent faire face à de **nouveaux challenges** (transition énergétique et normes associées ou ruptures technologiques (Internet des objets, systèmes autonomes) pour **concevoir les nouvelles architectures** de ces systèmes complexes/CPS (ex: avion tout électrique et le véhicule électrique hybride ou autonome ).
- L'intégration croissante des composants E/E dans la majorité des systèmes actuels requiert de **repenser les méthodes de conception + multi-domaines et multidisciplinaires**
  - ➔ Conception **mécatronique**

## Besoin « d'outiller » le travail de l'architecte système

- En parallèle, le **suivi de produits à longue durée de vie** ou susceptibles d'être déclinés avec de multiples **variantes**, ainsi que la forte tendance à la **réutilisation** (évolution de produits à partir de la version existante) nécessite la **traçabilité** entre les **solutions choisies**, les **exigences** auxquelles elles répondent et les **motivations** qui les ont faites apparaître.

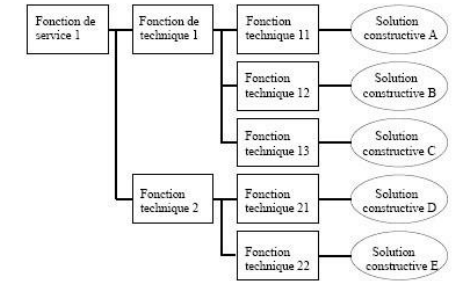
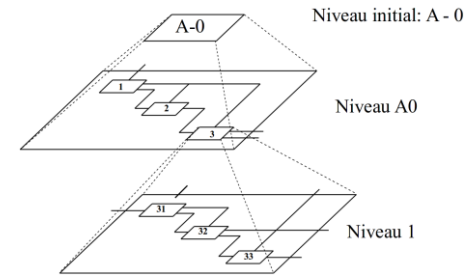
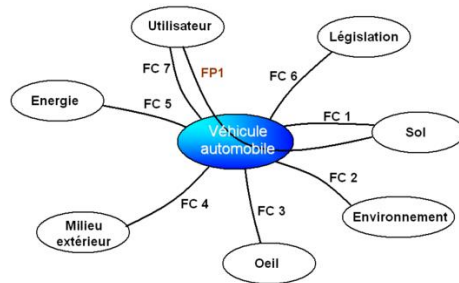


# Origine

7

- De nombreux outils d'analyse fonctionnelle en mécanique :

- APTE
- SADT
- FAST



- Des manques : absence de représentation comportementales, absence de formalisation globale liant les différents points de vue



Langage SysML

- Aspects multidisciplinaires → besoin d'une vision commune support à une collaboration efficace



Ingénierie Systèmes

- Besoin d'une approche basée sur les modèles pour la capitalisation et la réutilisation



Model-Based System Engineering (MBSE)

# Présentation de la méthodologie

F. Mhenni, J.Y. Choley, O. Penas, R. Plateaux, M. Hammadi

*A SysML-based methodology for mechatronic systems architectural design, Advanced Engineering Informatics 28 (3), 218-231*



# Aperçu de la méthodologie

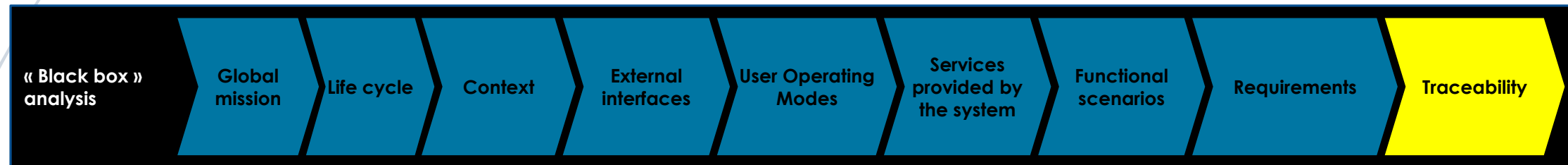
9

Un processus continu de modélisation descendante :

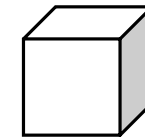
- ✓ de l'analyse " boîte noire " (point de vue externe) :



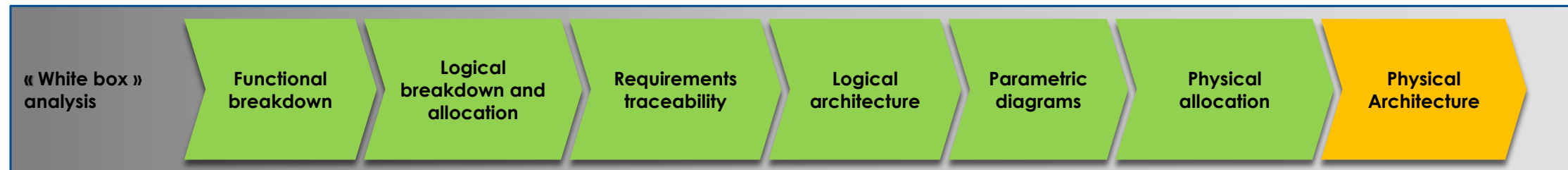
Que doit faire mon système ? Pour qui/quoi ?

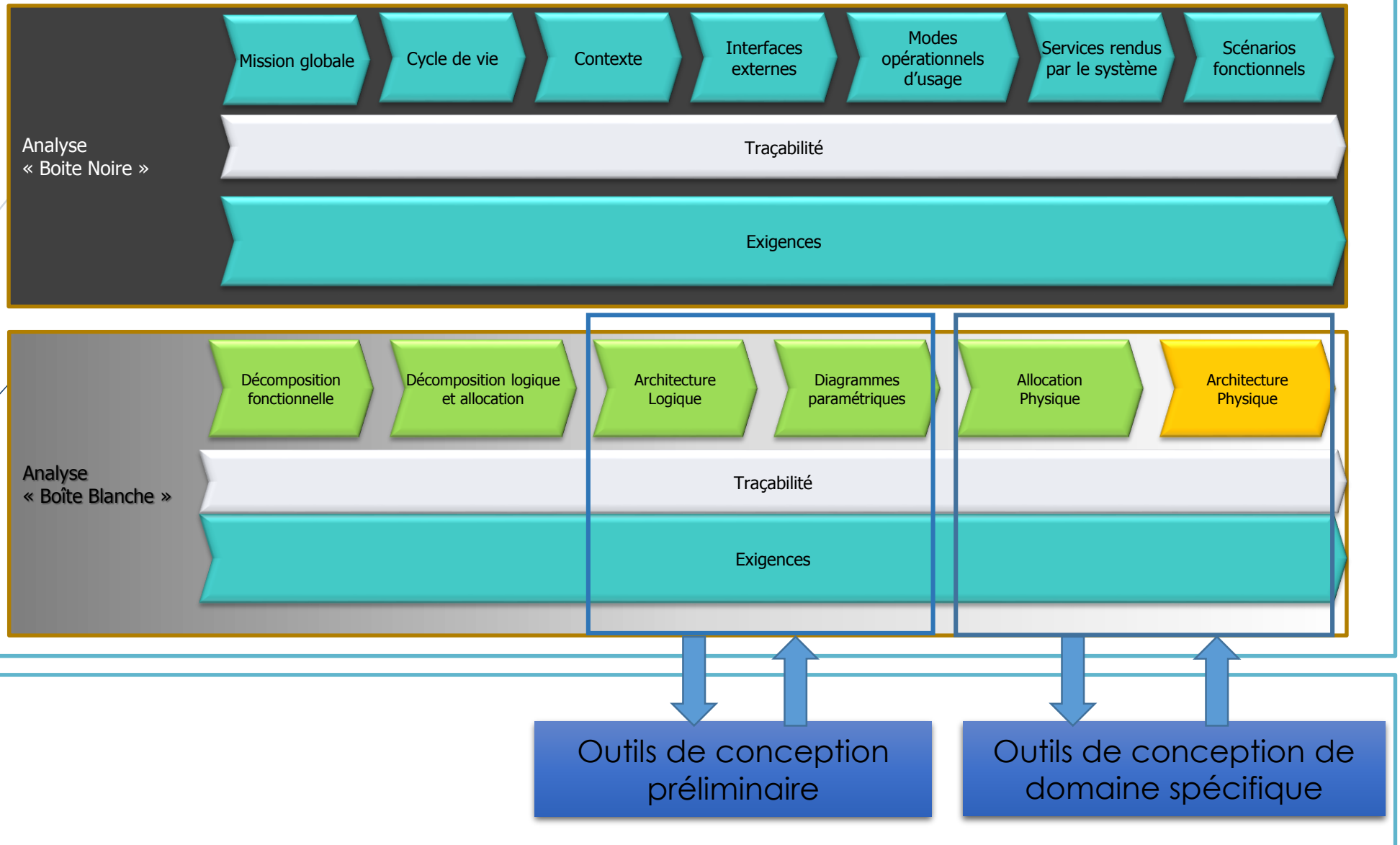


- ✓ à l'analyse " boîte blanche " (point de vue interne)



Comment doit être mon système pour répondre à ces exigences ?





# Emergence des exigences à partir du point de vue externe

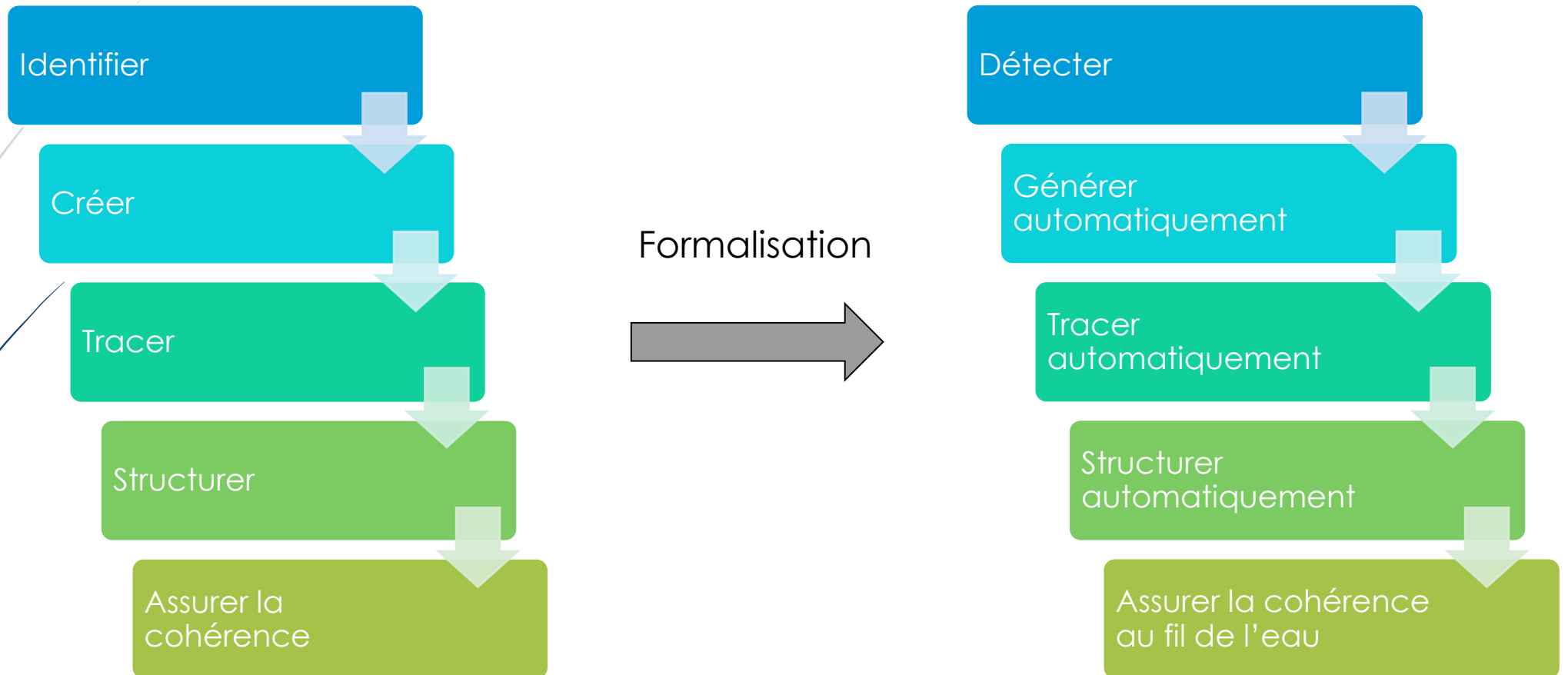
Farid Louni, Olivia Penas, Régis Plateaux, Antoine Spicher



#	ID	Name	Text
1	1	Doser un fluide	Permettre de doser un volume de fluide contenu dans le réservoir en minimisant fuites et pollution
2	1.7	Ne pas polluer ni être pollué	Étanchéité maximale
3	1.6	Acqu岸ir les grandeurs physiques	On souhaite pouvoir acquiser les évolutions temporelles de la pression et du déplacement du piston - et mesurer visuellement la pression et le débit de fluide dosé
4	1.5	Contrôler la pression	On doit pouvoir régler la pression du fluide refoulé
5	1.4	Contrôler le débit	Le débit doit être réglable entre 0 et 240 L/h
6	1.3	Mettre en mouvement le fluide	
7	1.2	Contenir le fluide véhiculé	Le fluide doit être véhiculé sans fuites
8	1.1	Stocker et isoler le fluide	Le fluide est stocké à vie et utilisé en circuit fermé dans le cadre du laboratoire de S2I
9	5	Pression maximale 10 bar	
10	3	Débit maximum 240 L/h	Suffisant pour une application pharmaceutique
11	2	Volume maximum	Le volume du réservoir est suffisant pour l'utilisation au laboratoire de S2I

Exigences

# Démarche pour la génération d'un ensemble complet et cohérent des exigences



# Illustration de la démarche par un exemple: Besoin client : conception du vélo électrique



Le système développé permettra le déplacement facile d'une personne en vélo d'un point A à un point B grâce à une assistance électrique.

L'assistance énergétique aura une source d'alimentation sans émission polluante.

Elle assistera l'utilisateur durant son déplacement en apportant un couple supplémentaire.

L'assistance fournie, déterminée automatiquement, garantira le respect des normes de conception d'un vélo. Le niveau maximum d'assistance sera sélectionné par le cycliste.

Notamment, pour l'homologation d'un tel système, la législation spécifie les contraintes suivantes:

- ▶ Est considéré comme un cycle, tout véhicule à énergie humaine,
- ▶ à laquelle on ajouterait une source d'énergie complémentaire qui deviendrait inactive (aucun couple n'est fourni même si l'assistance est sur ON):
  - ▶ lorsque l'on actionne les freins
  - ▶ lorsque la vitesse dépasse la valeur de 25 km/h
  - ▶ dès que le cycliste arrête de pédaler.

Lors du déplacement, les fonctionnalités du vélo permettront au cycliste de s'éclairer, de changer de vitesse automatiquement (mécanique), de freiner et de sélectionner l'assistance et son niveau (quantité d'énergie apportée).

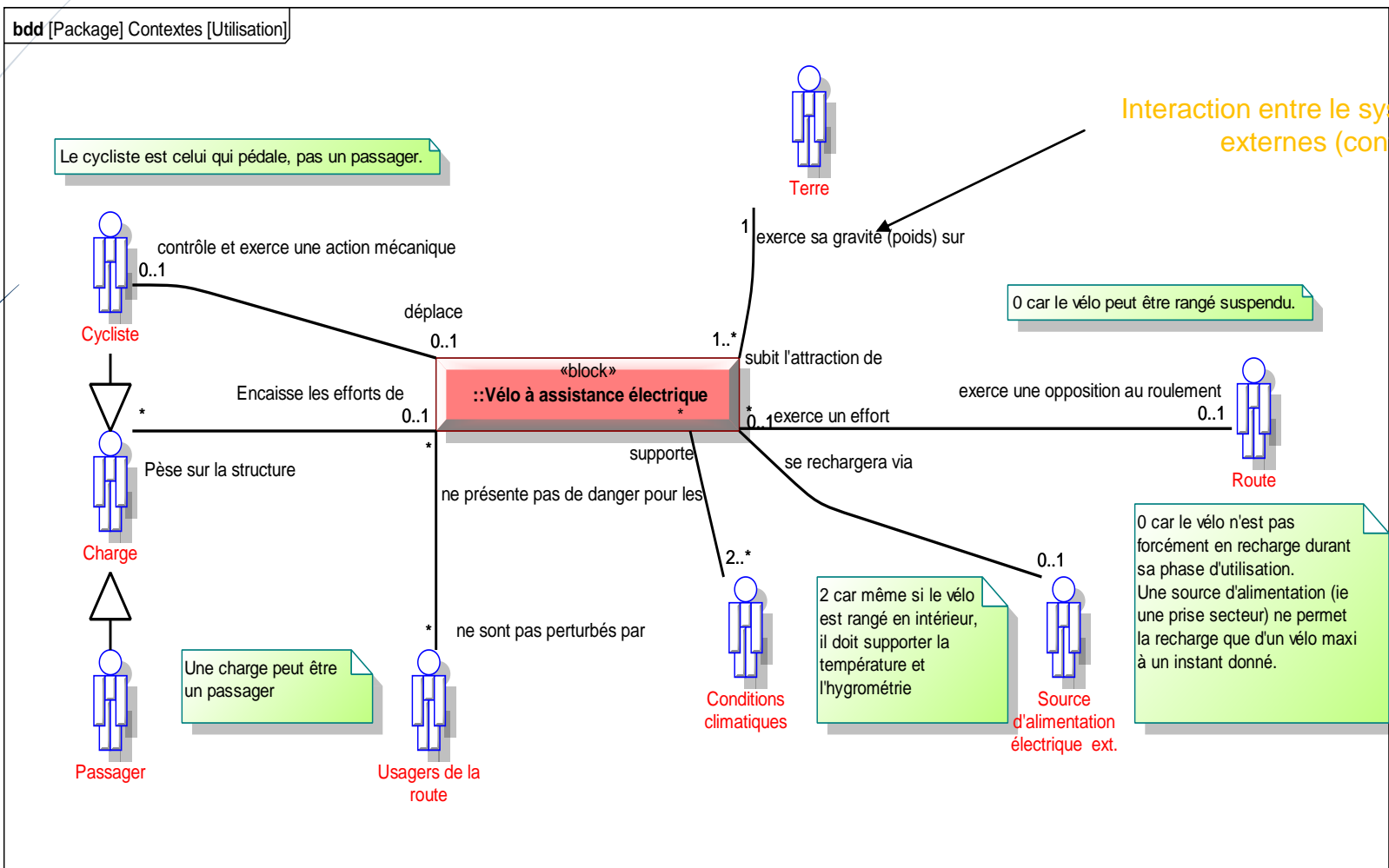
# 1. Identifier les conditions d'apparition des exigences dérivées

- Identifier quelles sont les conditions d'apparition des exigences dans les différents points de vue exprimés dans chaque diagramme SysML, par exemple :
  - Interactions système/environnement : associations, opérations

# Contexte de la phase d'utilisation

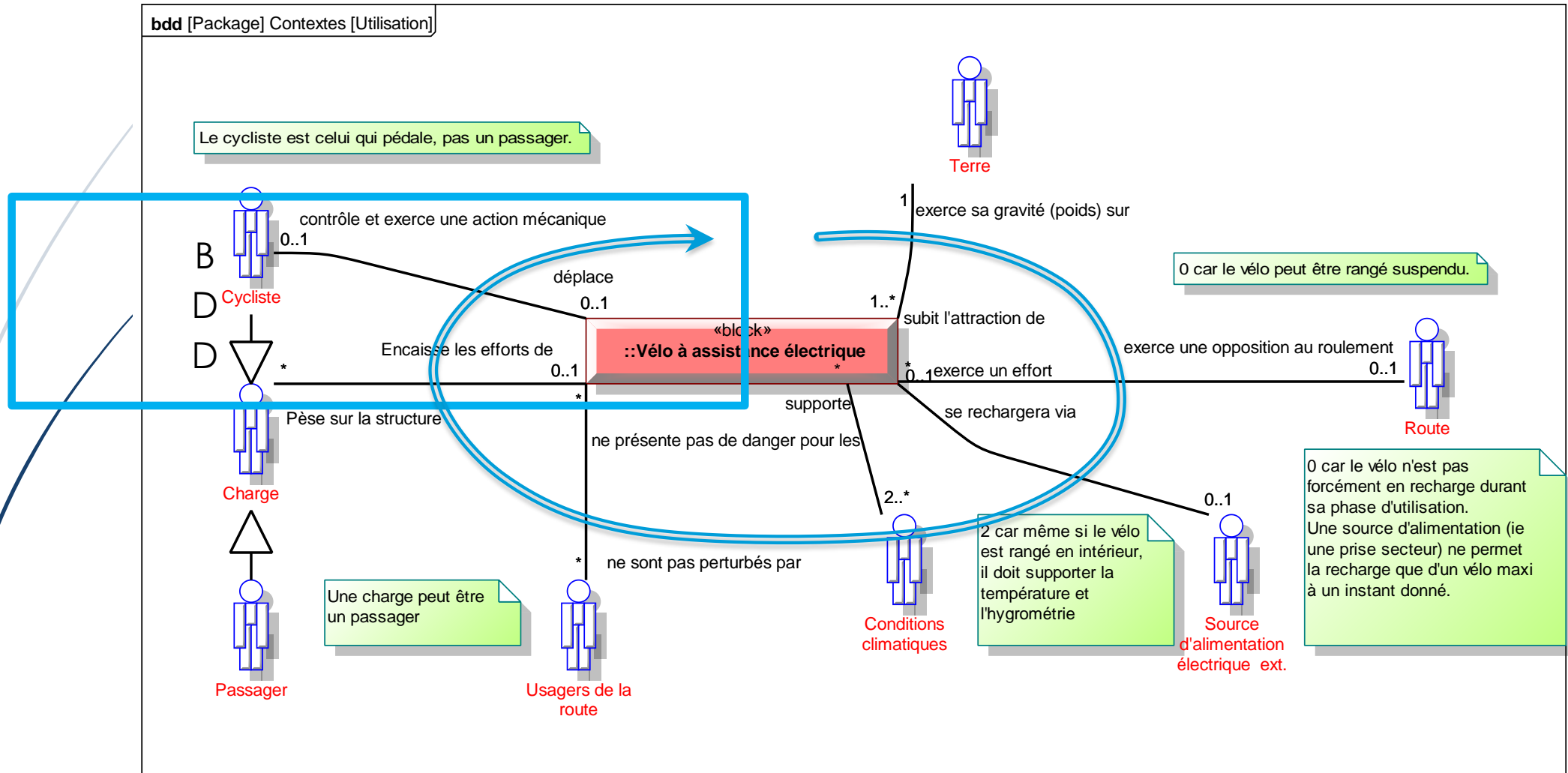
15

Définit les limites du système et les éléments de l'environnement (systèmes/acteurs)



# A partir du diagramme de Contexte (BDD)

16





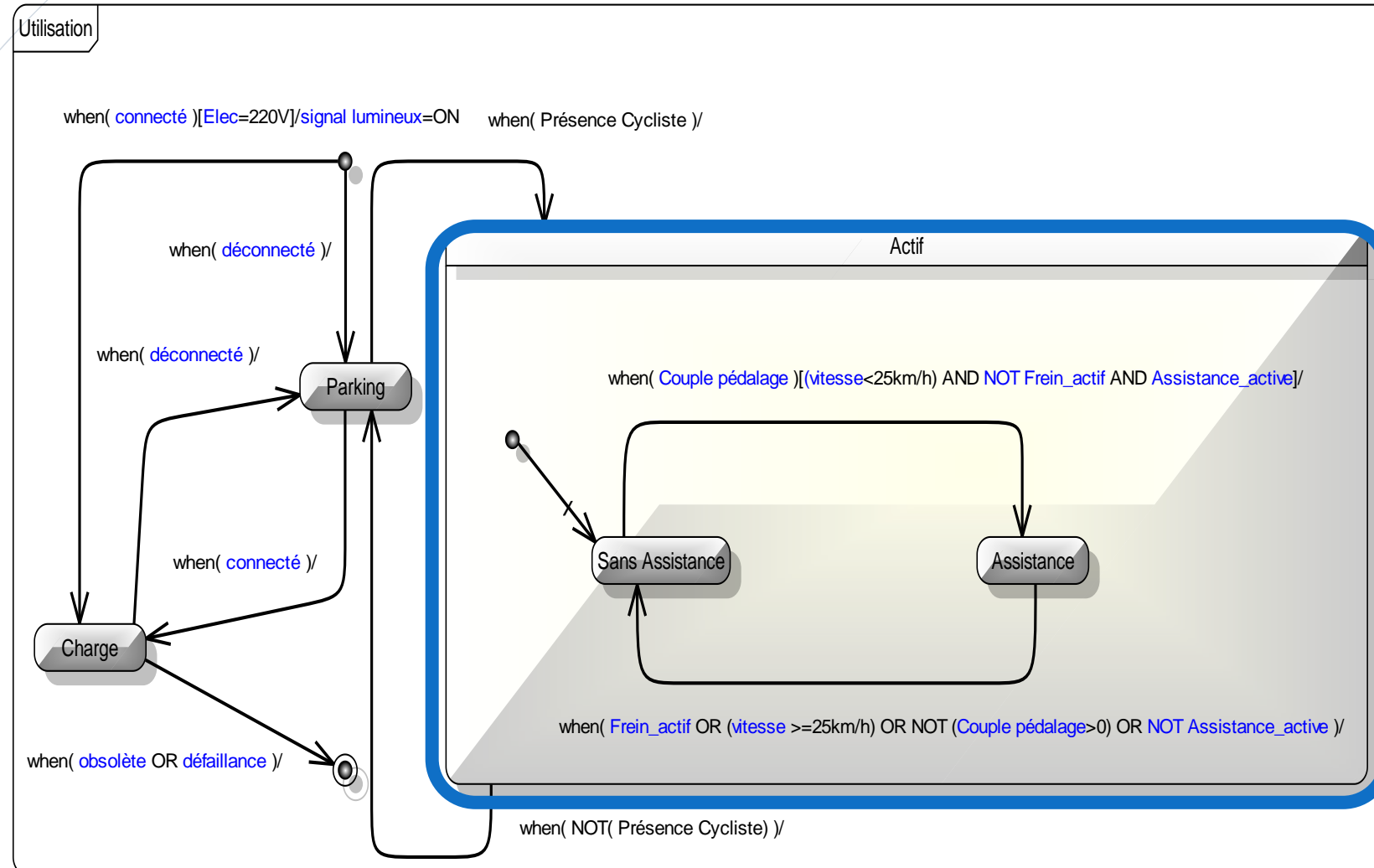
# 1. Identifier les conditions d'apparition des exigences dérivées

- Identifier quelles sont les conditions d'apparition des exigences dans les différents points de vue exprimés dans chaque diagramme SysML, par exemple :
  - Interactions système/environnement : associations, opérations
  - Déclencheurs liés aux changements d'état

# Modes de fonctionnement en phase utilisation

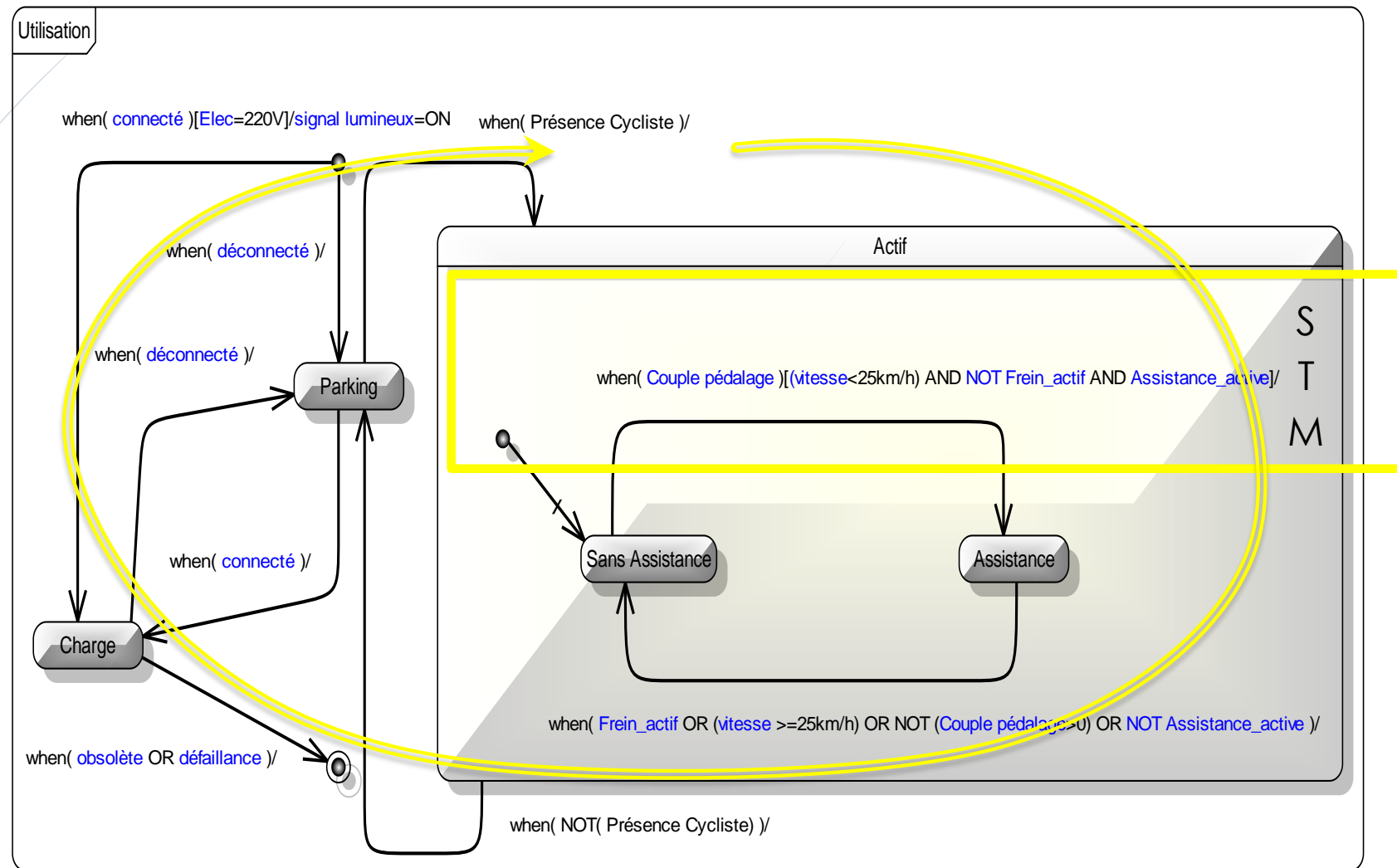
18

Définition des "modes de fonctionnement" du système perçus par l'utilisateur.



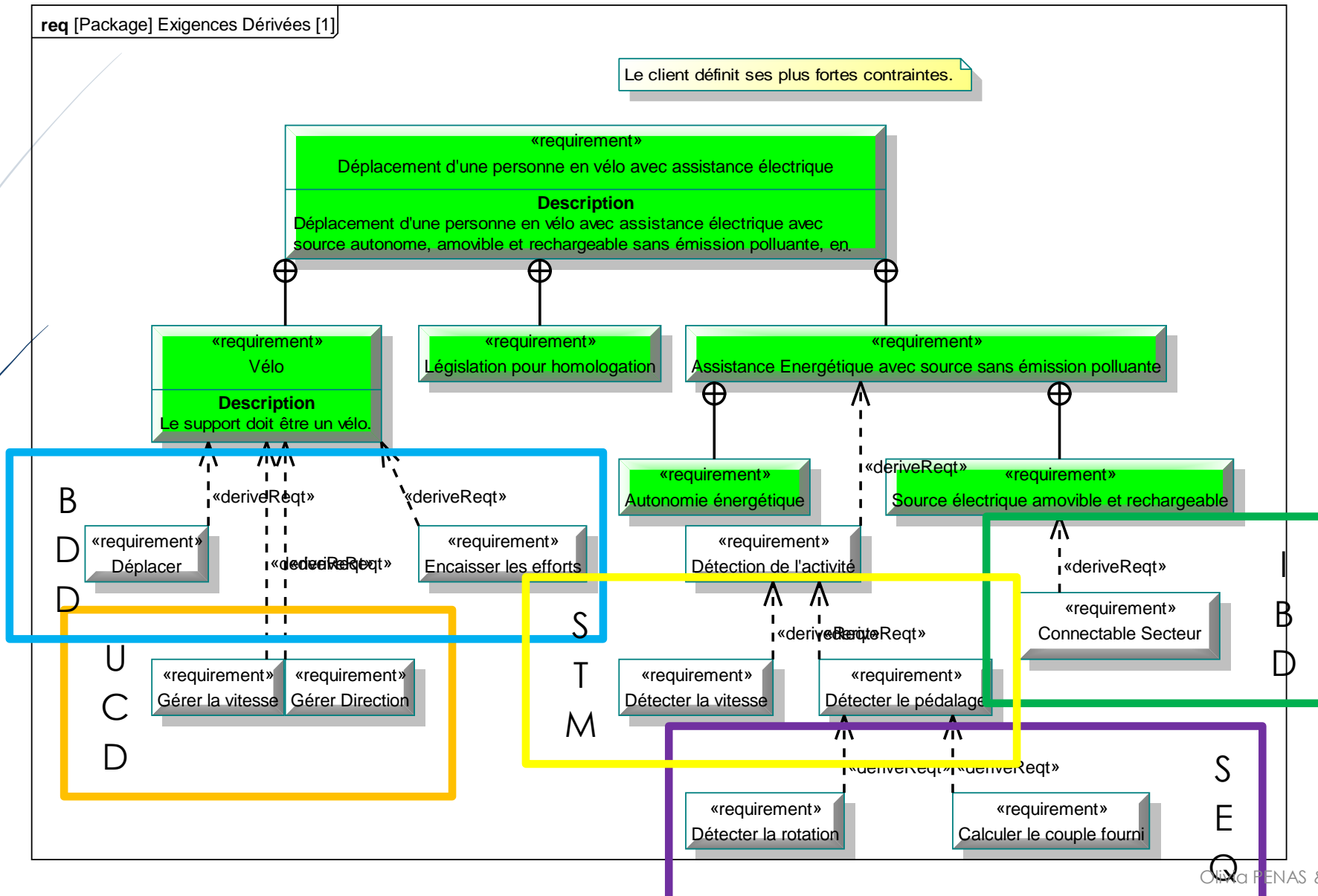
# A partir du diagramme STM

19



## 2. Créer les exigences dérivées (derived req)

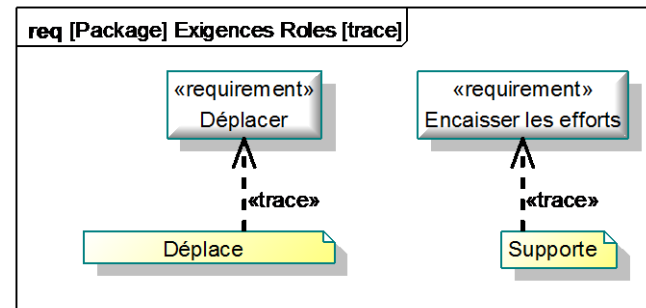
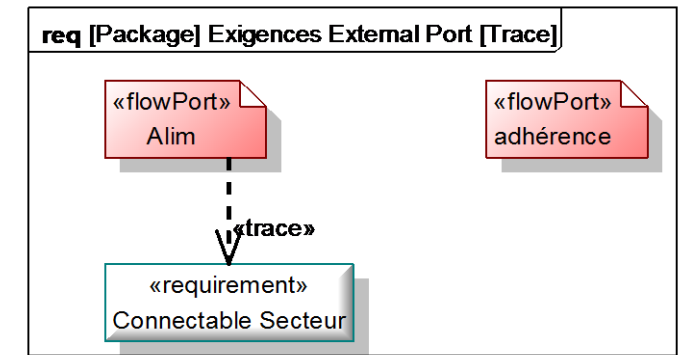
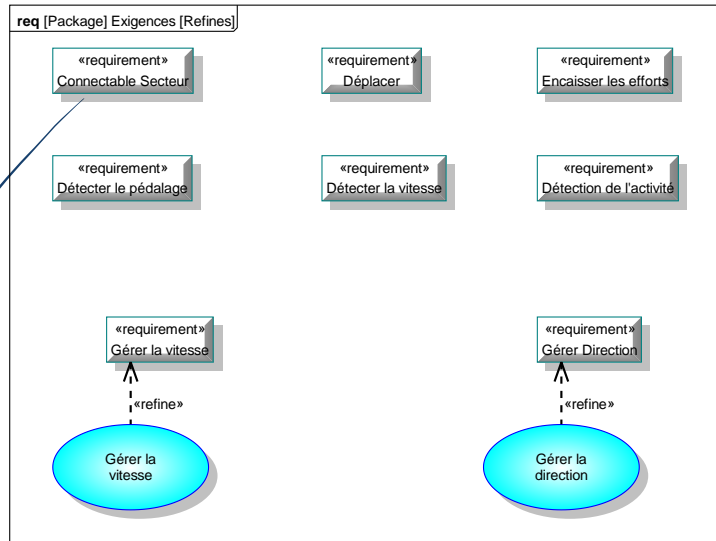
20



# 3. Les tracer

21

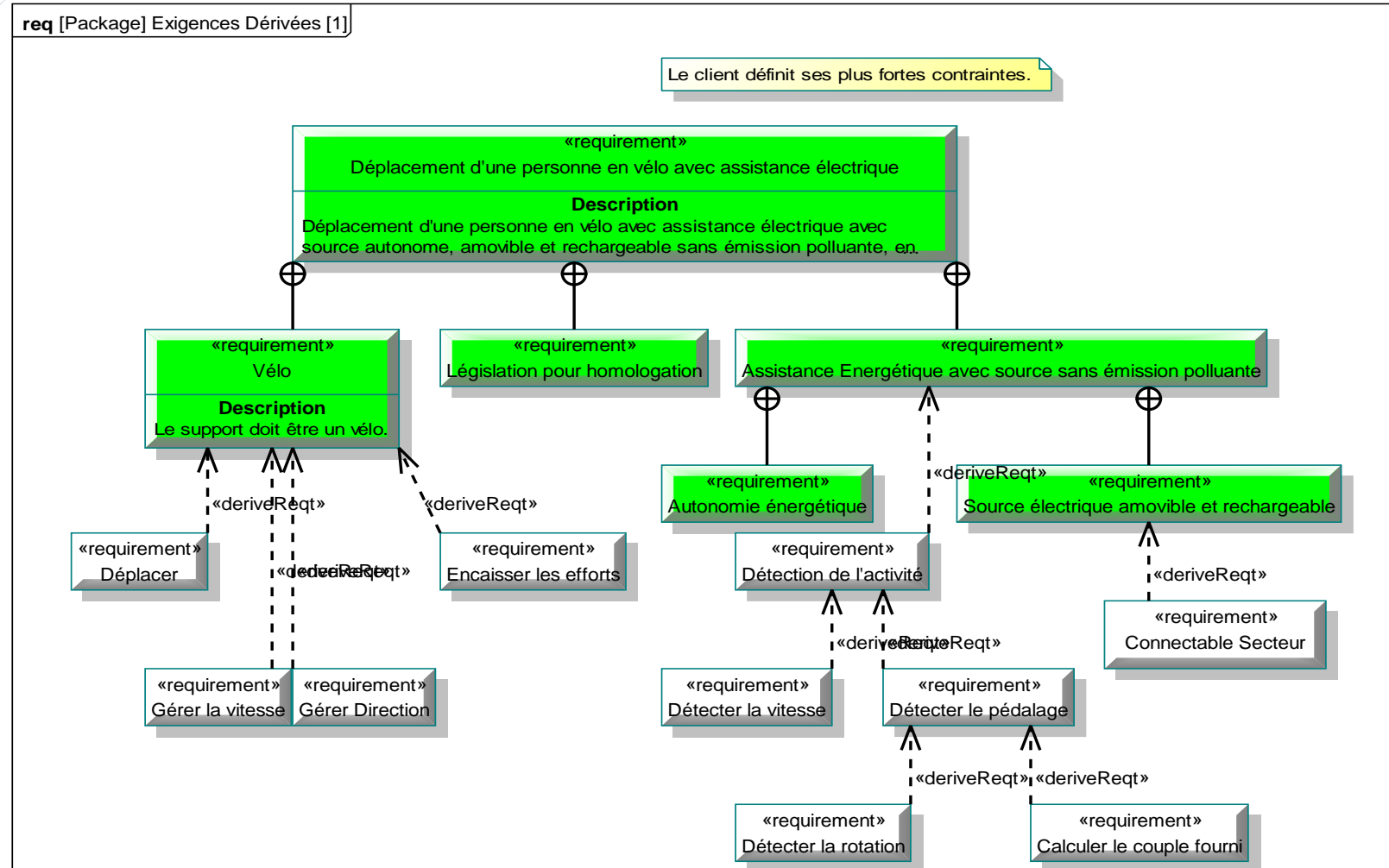
- “refine” entre Exigences et UC;
- “trace” entre Exigences et rôles dans le diag BDD de contexte;
- “trace” entre Exigences et ports externes (IBD).



Le lien « Trace » sera utilisé pour exprimer le lien de causalité entre les éléments issus de l'analyse et la présence de l'exigence

# 4. Les structurer

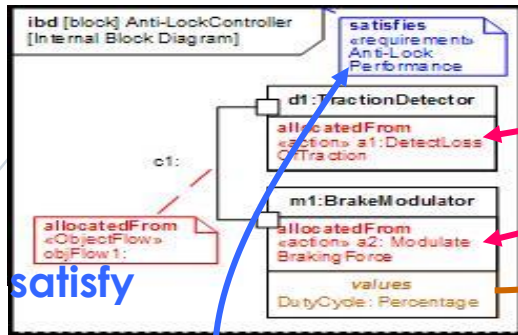
22



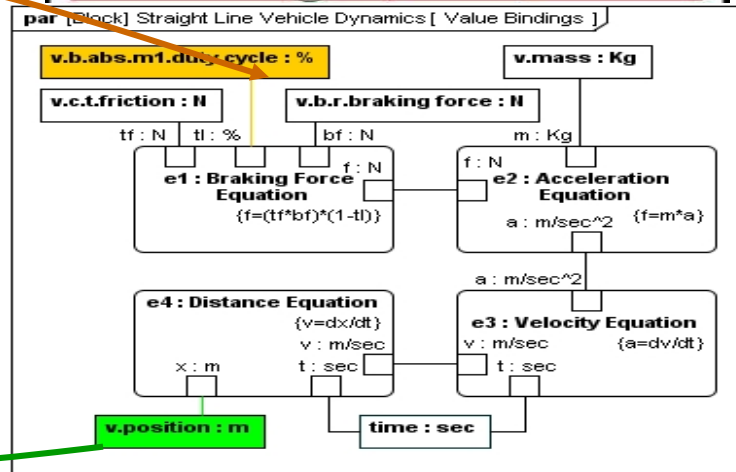
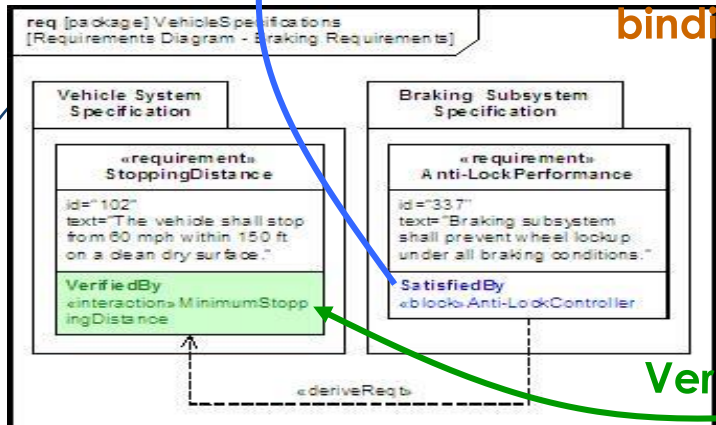
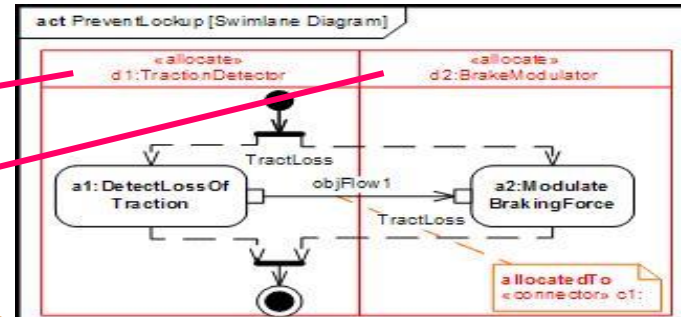
# 5. Assurer la cohérence

23

## 1. Structure



## 2. Behavior



## 3. Requirements

## 4. Parametric

Liens d'allocations, de traçabilité et de vérification de la cohérence principalement générés par l'architecte système.



# Formalisation

Structurer : Architecture des exigences via un langage pivot,  
Assurer la cohérence au fil de l'eau : Catégories



# Architecture des exigences dérivées

## Langage Pivot

Farid Louni, Olivia Penas, Régis Plateaux

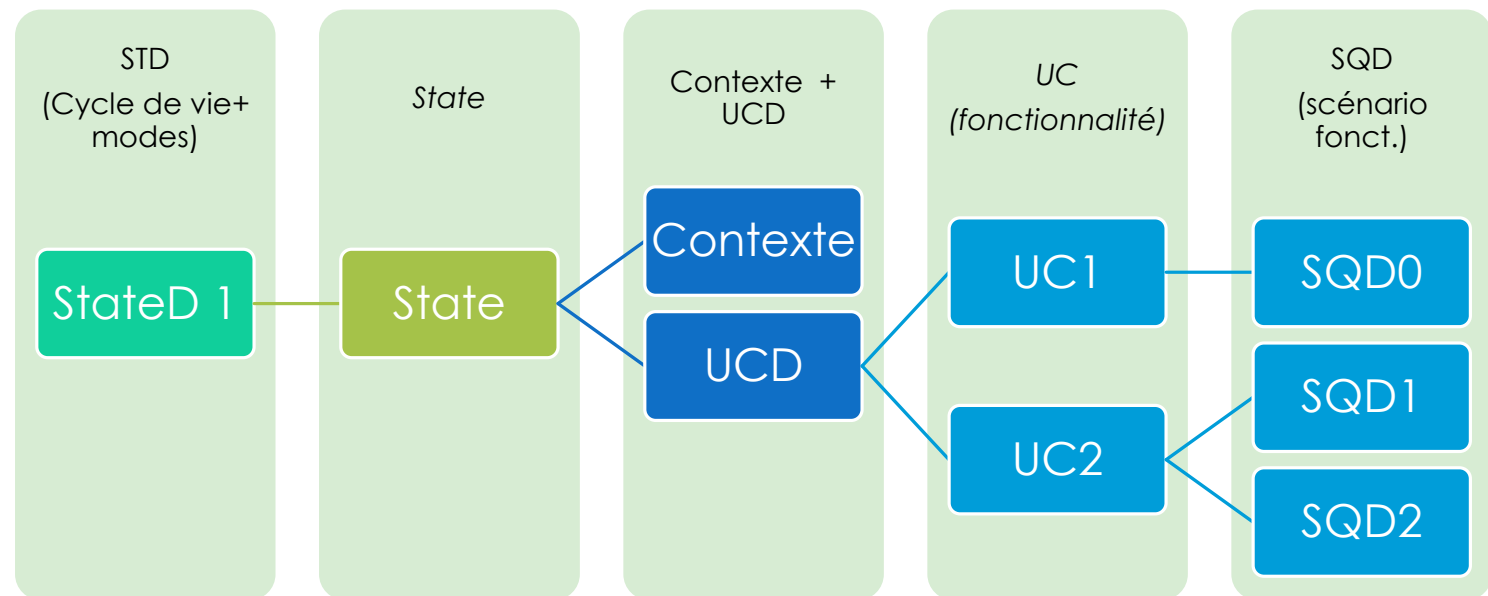
# Motivations: quelle architecture pour les exigences émergentes ?

- Aspect composite de SysML nécessitant des liens de traçabilité et d'allocation
- Le diagramme de séquence offrant le raffinement le plus fin a été choisi comme langage pivot : il permet de capturer les liens définis par la méthodologie de l'ensemble des diagrammes SysML dans une vue unifiée et ainsi d'aller au-delà des relations d'allocation et de traçabilité déjà présentes dans le langage de ce diagramme.
- Structuration des exigences via le processus d'analyse décrit par l'encapsulation des diagrammes de séquence
- Mais la cohérence des graphes reste assurée actuellement par l'ingénieur

# Structure sous-jacente à la méthodologie

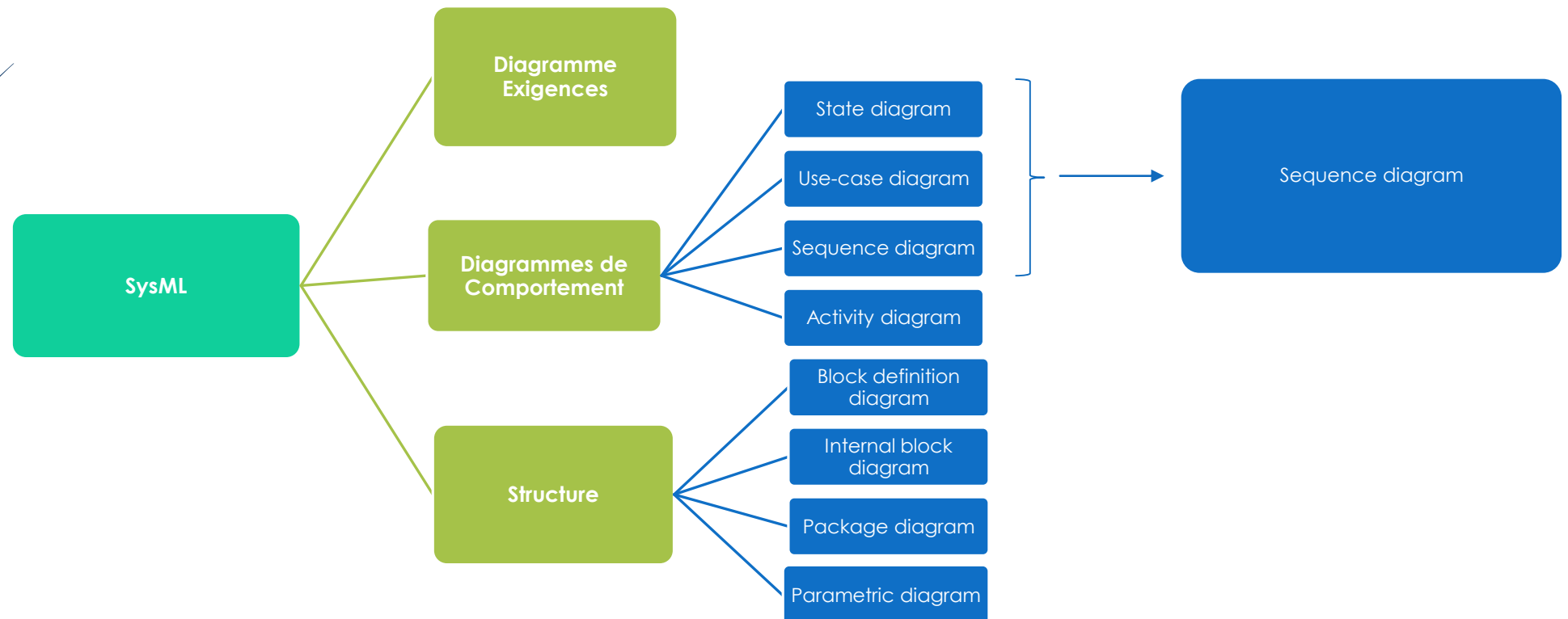
27

- Démarche descendante permettant un passage à une granularité de plus en plus fine jusqu'à arriver aux opérations du diagramme de séquence.
- Cohérence garantie par l'ingénieur
- Vérification pas-à-pas



# Unification des langages associés aux diagrammes SysML

- Unification sous-jacente des langages dans SysML : l'IS manipule les diagrammes SysML alors que le langage pivot garantit, par l'unification, la cohérence et porte le raffinement structurel des exigences dérivées.



# Architecture des exigences capturée par le langage pivot

29

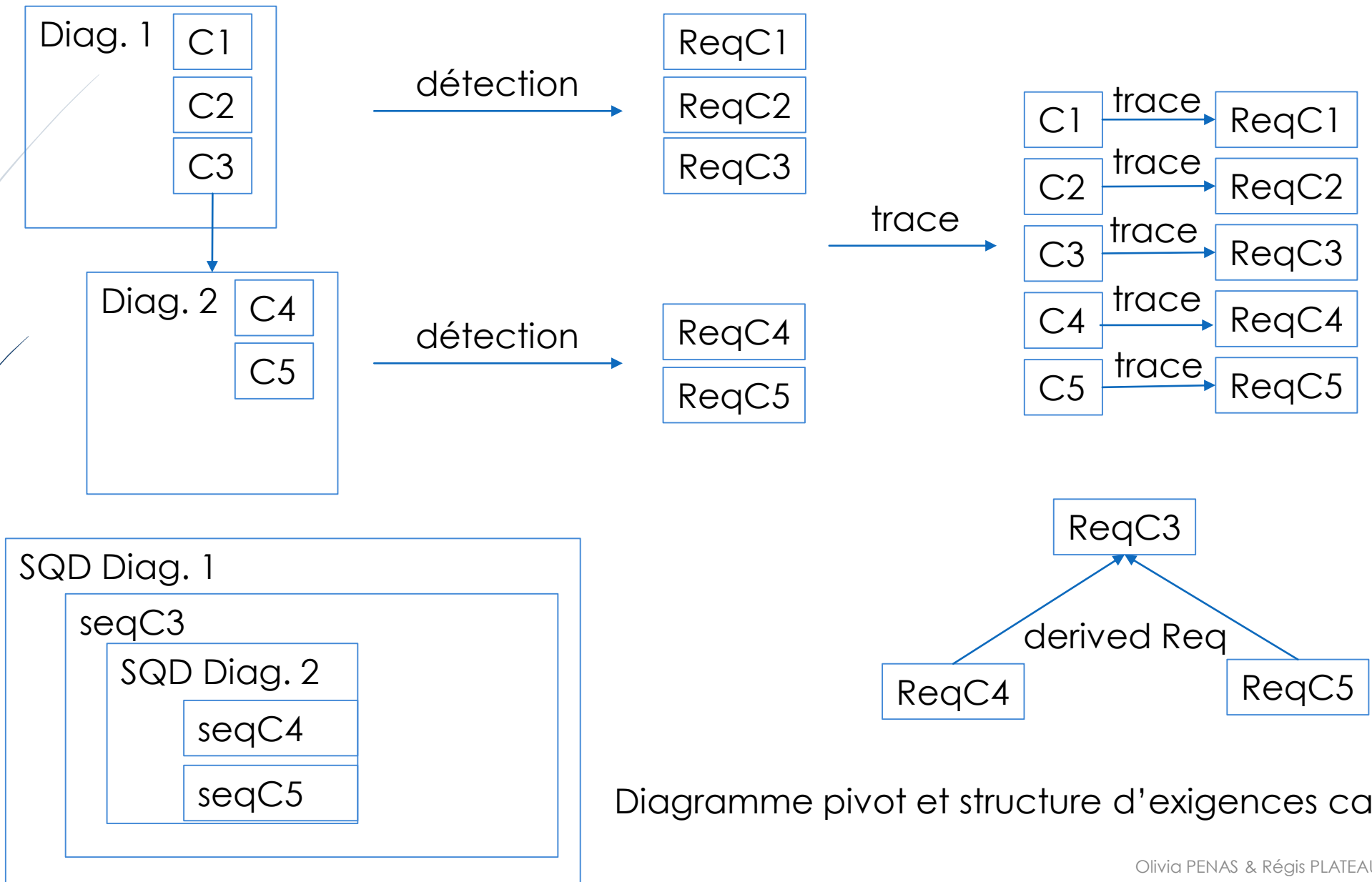


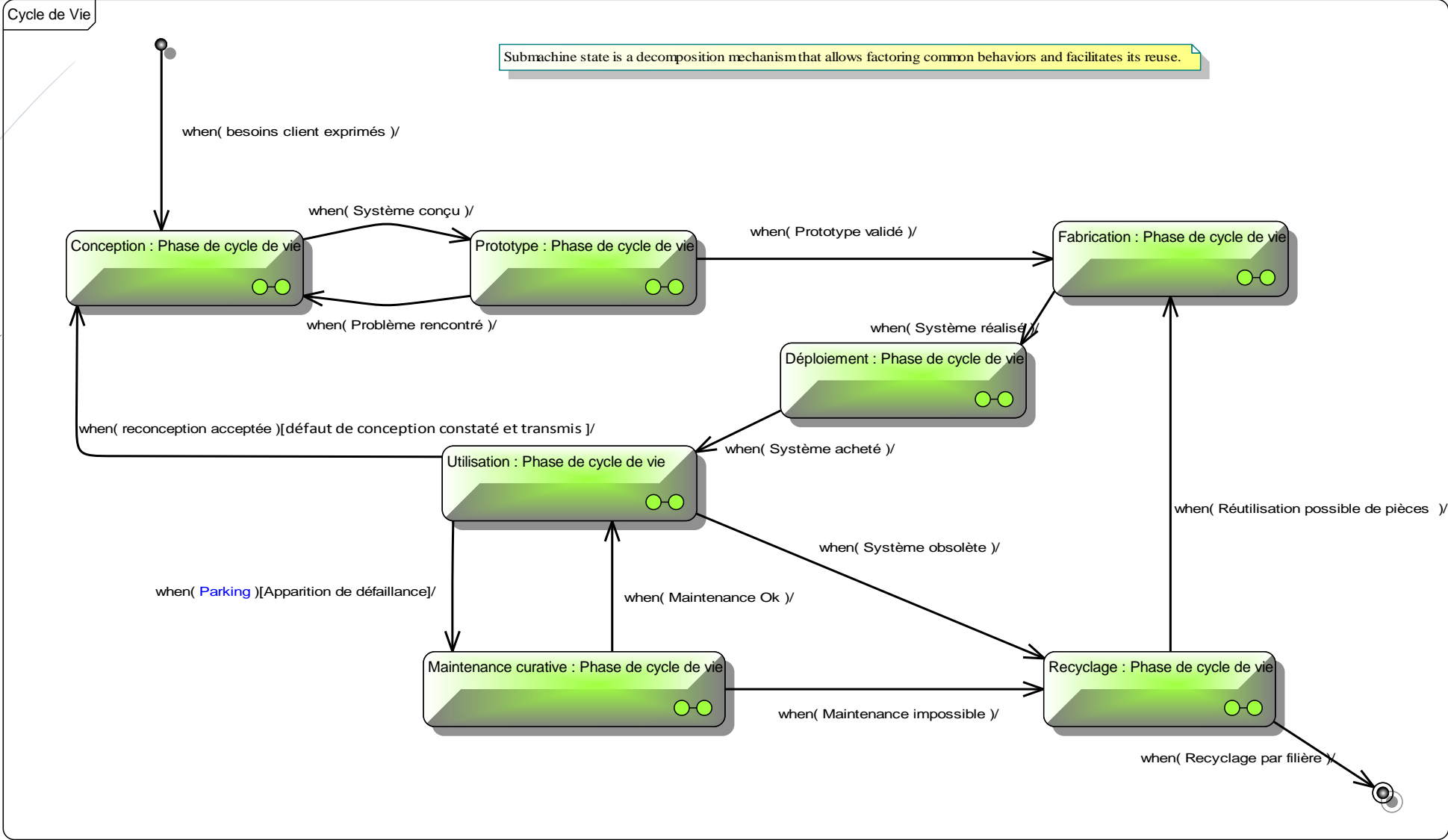
Diagramme pivot et structure d'exigences capturées

# Assurer la cohérence malgré l'hétérogénéité

## Ontologies et catégories

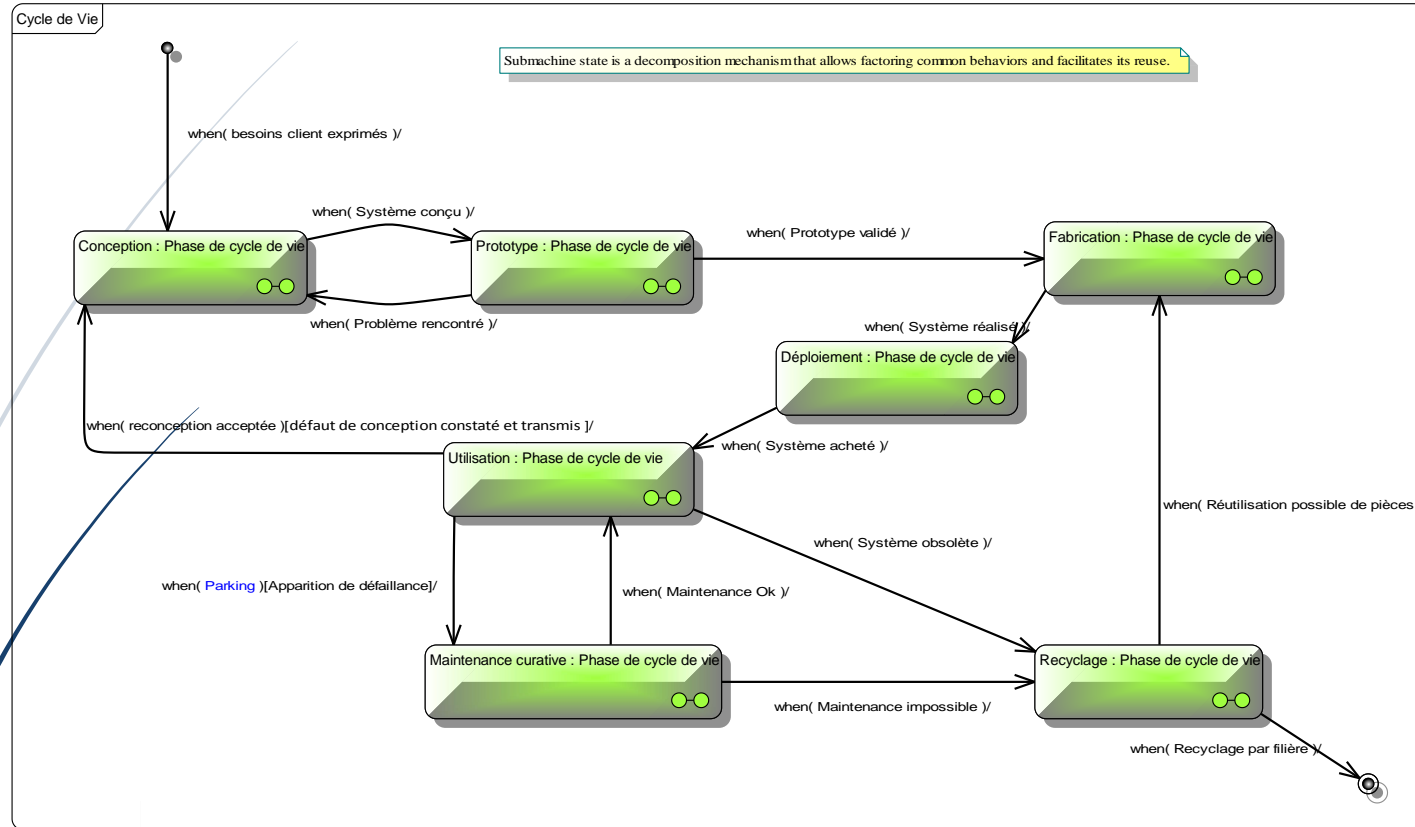
Olivia Penas, Régis Plateaux, Antoine Spicher

# Cycle de vie du produit

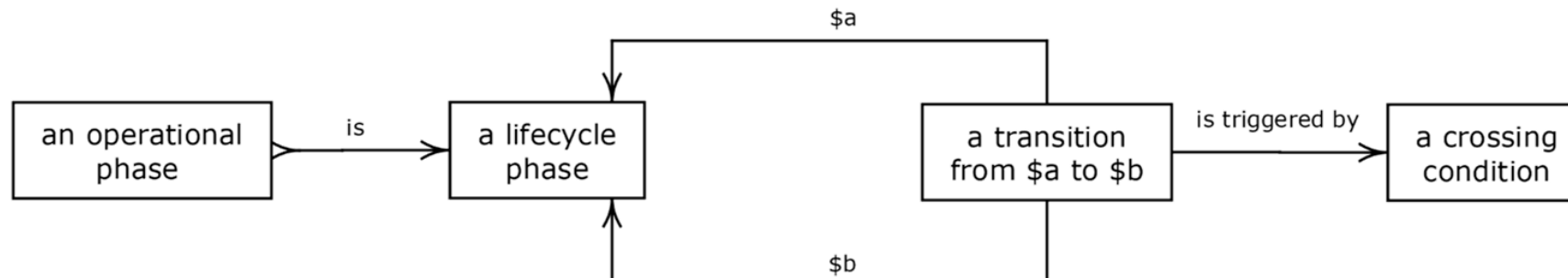


# Phases du cycle de vie

32



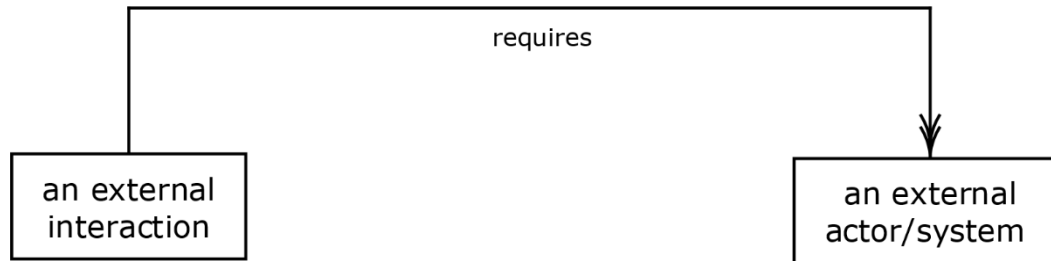
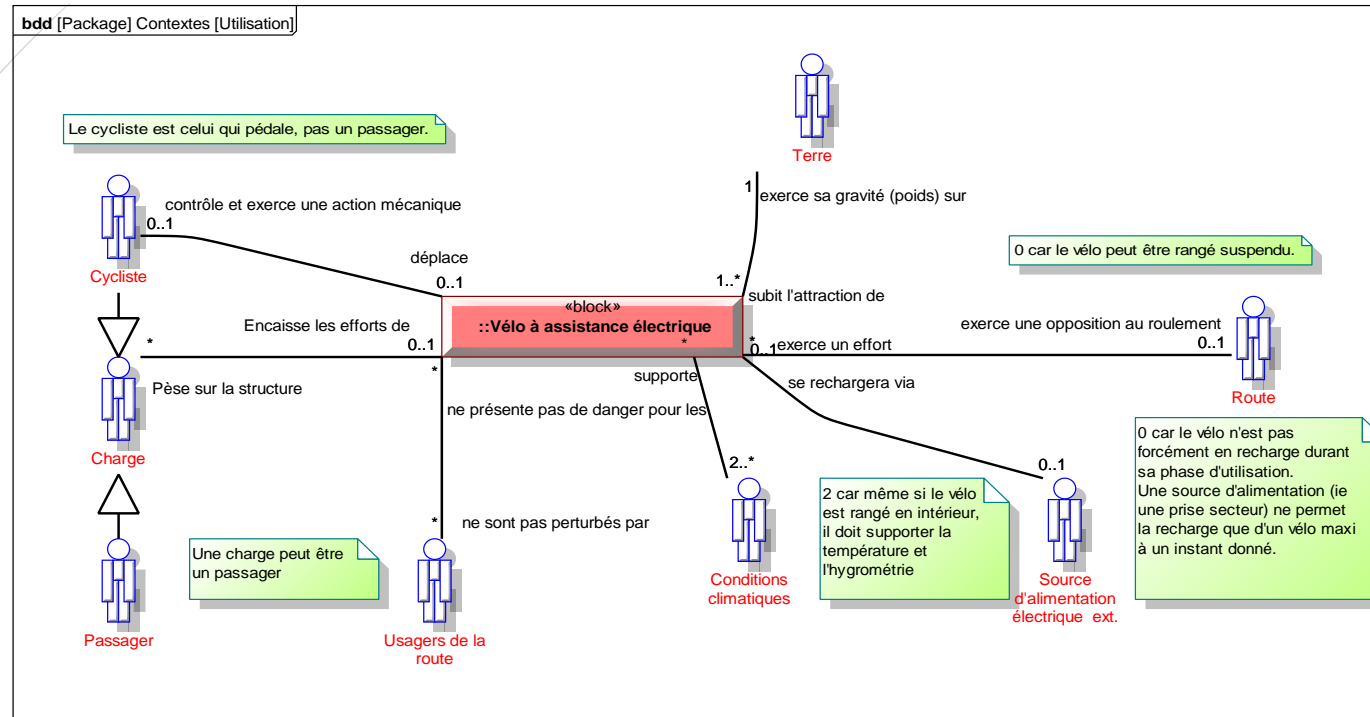
monomorphism





# Vue du contexte

33



epimorphism



# Conclusions

34

- ➔ Émergence des exigences dans une approche MBSE s'appuyant sur SysML
- ➔ Démarche pour structurer ces exigences via un langage pivot
- ➔ Formalisation pour supporter l'automatisation de l'ingénierie de ces exigences par les ontologies et les catégories.