



# De la nécessité d'une bonne analyse

ou

*« petit traité de bonnes pratiques liées à l'analyse des besoins / exigences en SysML en vue de son exploitation pédagogique ».*

Y. Le Gallou

# Remarques préliminaires

Cet article, basé sur des faits réels, résume et approfondit la démarche qui fut mienne pour expliquer la nécessité d'une bonne définition des besoins (complétée partiellement par l'analyse des exigences) en vue d'une exploitation pédagogique, lors d'une formation SysML à des collègues de CPGE (nov 2013).

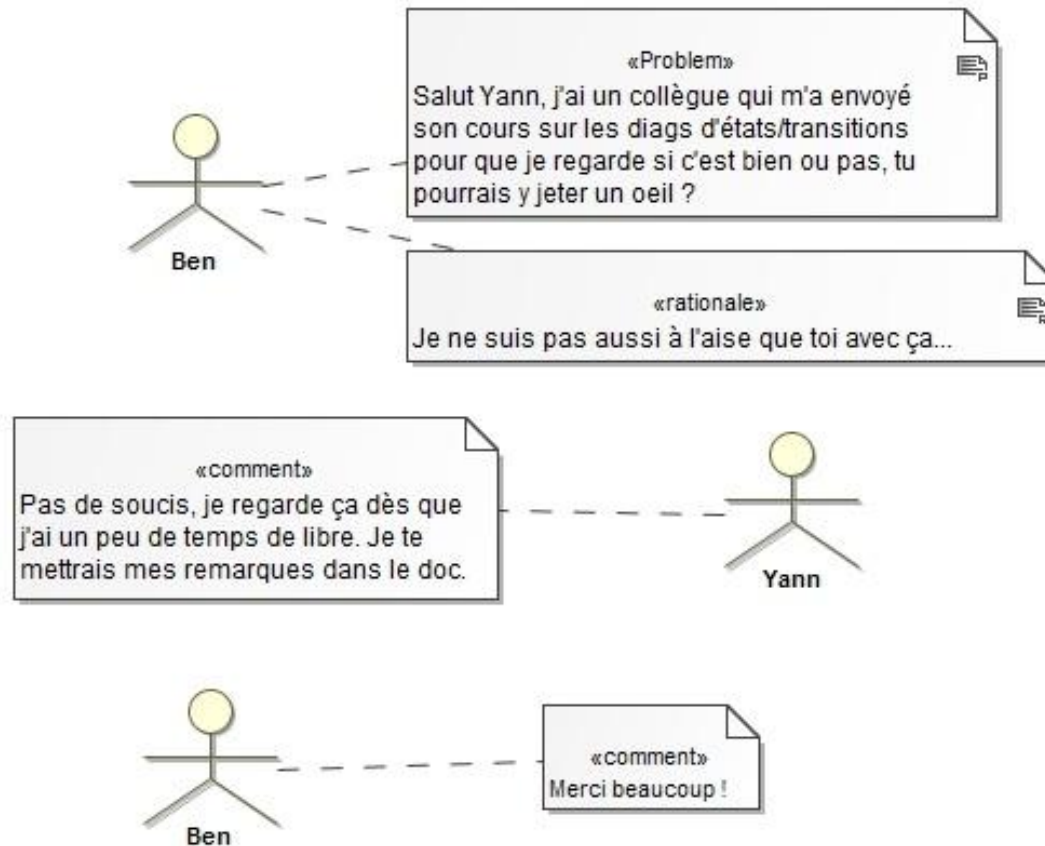
Durant cet exposé je corrige le travail initial d'un collègue : ces remarques ne sont pas là pour critiquer un travail (et en l'occurrence ce que propose le collègue est quand même loin d'être le pire de ce qu'on a pu voir jusqu'à présent), mais pour expliquer à quoi des erreurs fréquentes sont dues et comment les éviter.

Je tiens donc à remercier ce collègue pour sa coopération et son accord quant à l'utilisation que je fais de ses travaux dans cet exposé.



# Prologue

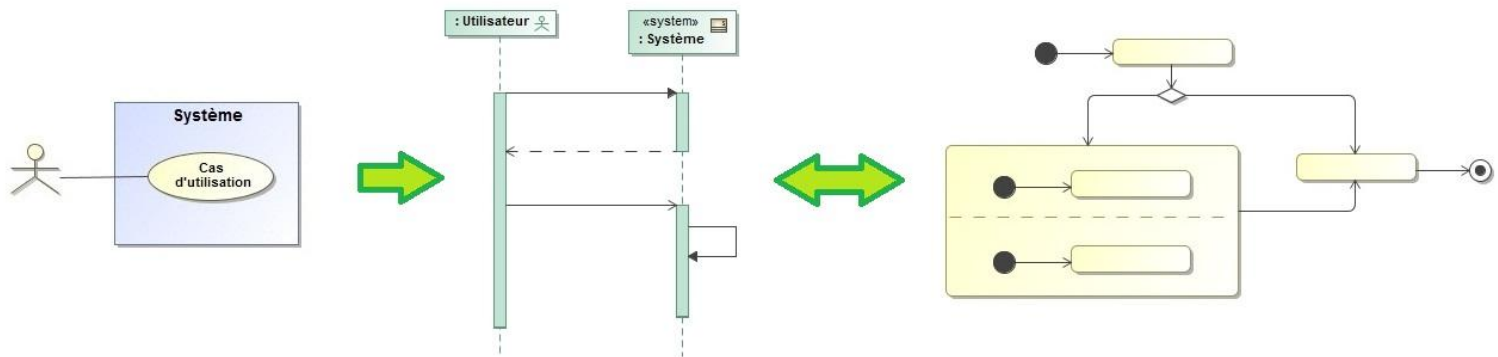
# Il était une fois un besoin.



Ma mission, si je l'accepte, est de le satisfaire.

# Préambule

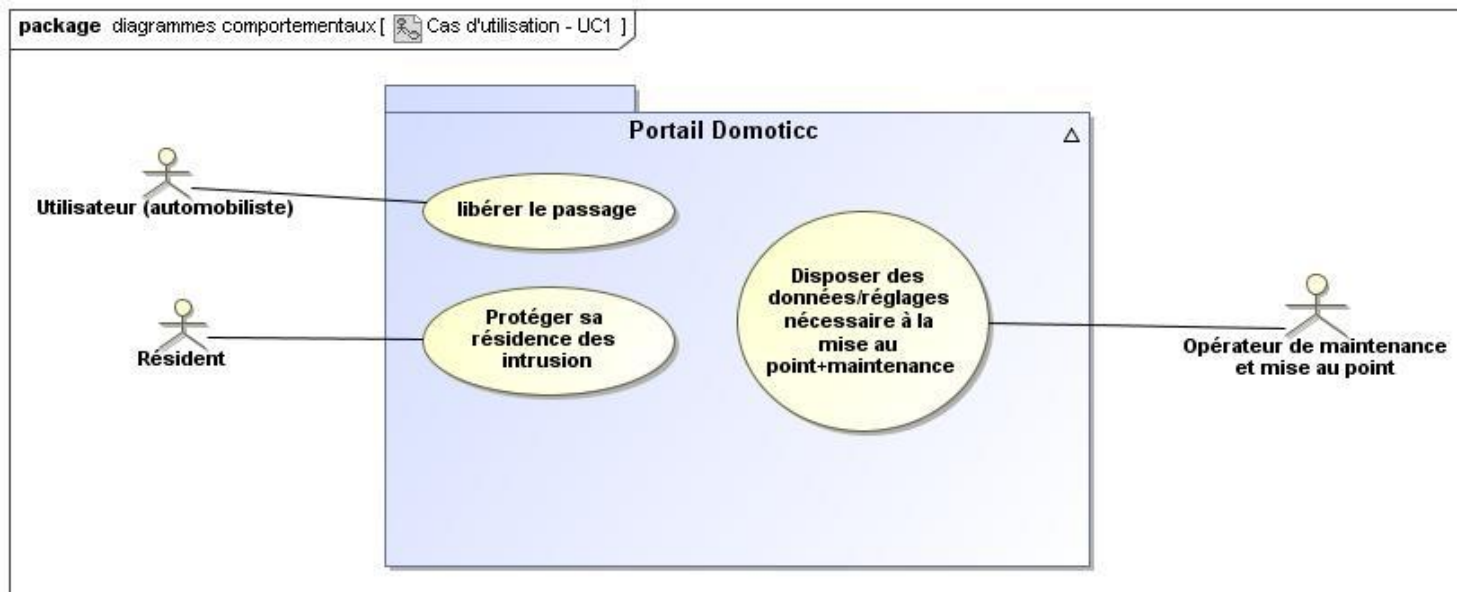
- Le collègue cherche à introduire le diagramme d'états/transitions au travers du diagramme de séquence, et donc d'un scénario d'utilisation du système.
- Il lui faut donc partir d'un cas d'utilisation :



Cela me semble cohérent, voyons cela...

# Cas d'utilisation

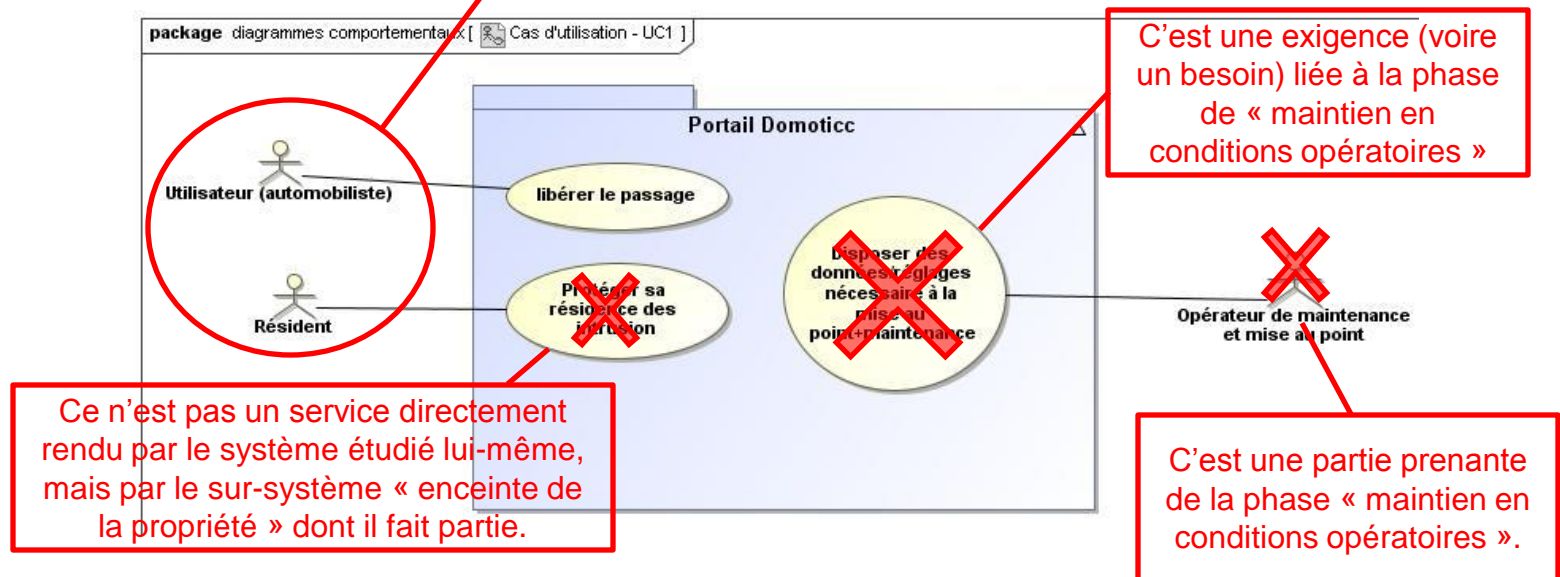
- Le support fil rouge est un portail automatique : le « **Domoticc** » (que je ne connais pas personnellement) :



- Premier constat : des erreurs classiques sont présentes, comme cela se retrouve fréquemment, le diagramme des cas d'utilisation étant l'un des plus mal compris encore actuellement. Néanmoins cela reste de bonne facture pour un premier jet. Analysons...

# Cas d'utilisation (annotations)

Cela pourrait être une seule et même personne... Utilisateur est un terme générique, et on comprend que la dénomination de l'acteur pose problème (automobiliste utilisateur du portail...)



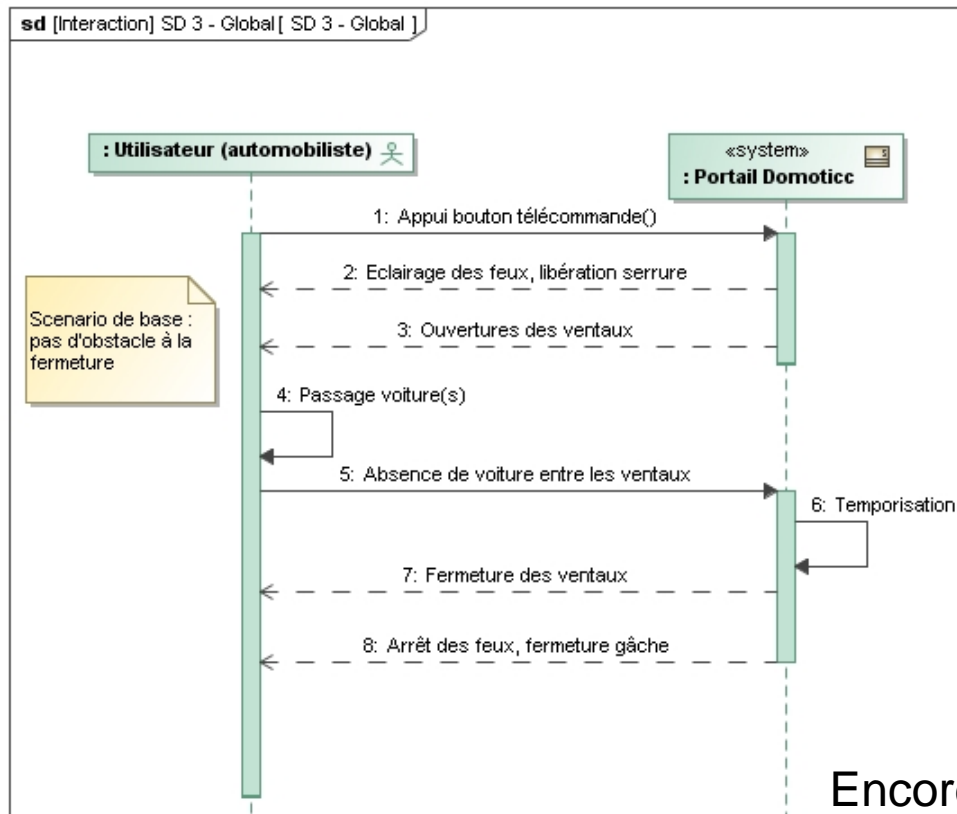
Manifestement ici, les erreurs sont dues à :

- une mauvaise appréciation des frontières du système ;
- une confusion des parties prenantes ;
- une non-distinction des différentes phases de vie du système ;
- une confusion entre un cas d'utilisation et une exigence (erreur classique).

A noter qu'ici le collègue n'est pas tombé dans le piège classique de la décomposition fonctionnelle, que nous voyons encore trop souvent (j'espère un jour pouvoir éviter ce genre de remarque...voir l'article de Pascal Roques à ce sujet [1])

# Séquence

- Le collègue introduit ensuite le diagramme de séquence par :  
*Observons le cas d'utilisation le plus intéressant du portail  
« libérer/fermer le passage »*

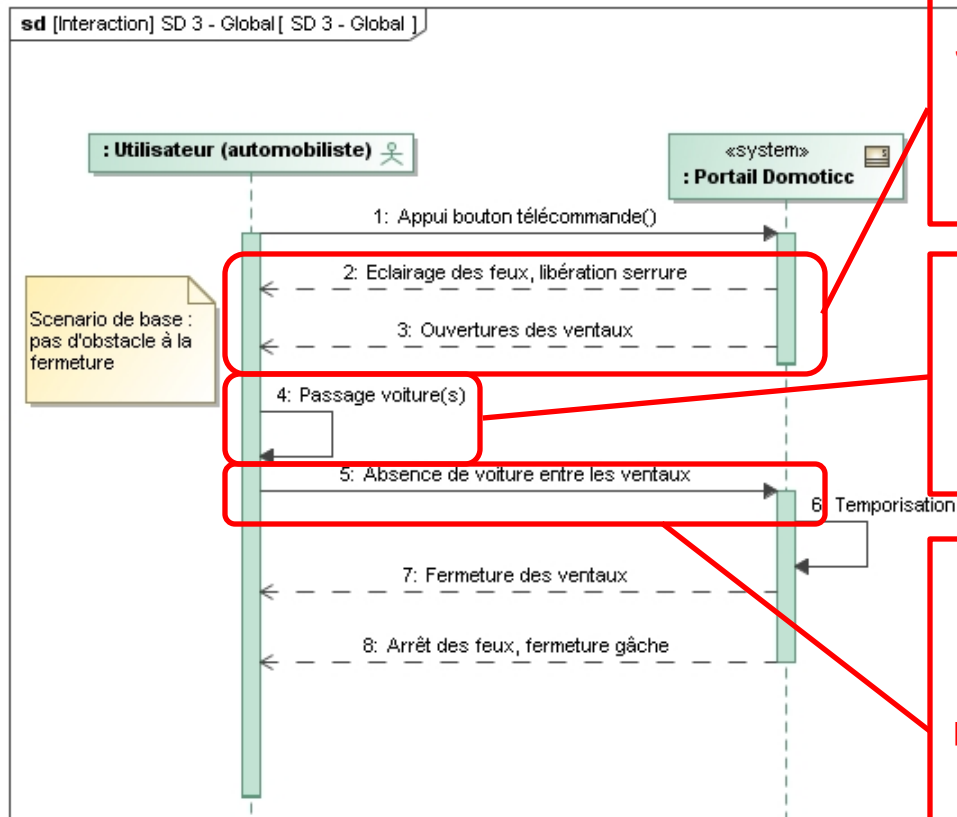


**Remarque** : en même temps il n'y a finalement que ce cas d'utilisation. Cela n'a rien de péjoratif, mais j'ai souvent constaté cette tendance qu'ont certaines personnes à vouloir qu'il y ait plein de choses dans un système pour pouvoir dire : « nous on va seulement regarder ce qui se passe là ! ». Poussé à l'extrême, on tombe très vite dans le phénomène de remplissage (comme on peut le constater trop fréquemment...).

Encore une fois, un premier jet assez abouti. Analysons...



# Séquence (annotations)



2 retours synchrones : il y en a un de trop. Je garderai l'éclairage des feux pour le retour (c'est bien une interaction), par contre l'ouverture est faite en autonomie par le système (on n'agit ni n'interagit pas avec l'utilisateur pour ouvrir).

La notion même de message réflexif sur l'utilisateur me laisse très perplexe : en quoi une activité réalisée en autonomie par l'utilisateur et qui ne donne un résultat observable que par lui-même est-il pertinent dans la description d'un système ? ...

...mais je saisis avec ce message l'intention de communiquer : on veut retranscrire la non présence d'obstacle en associant l'utilisateur à l'obstacle (la voiture en l'occurrence). Le problème ici est que l'utilisateur n'agit pas sur la détection d'obstacle, et il y a là encore manifestement une confusion entre les parties prenantes et les éléments du contexte.

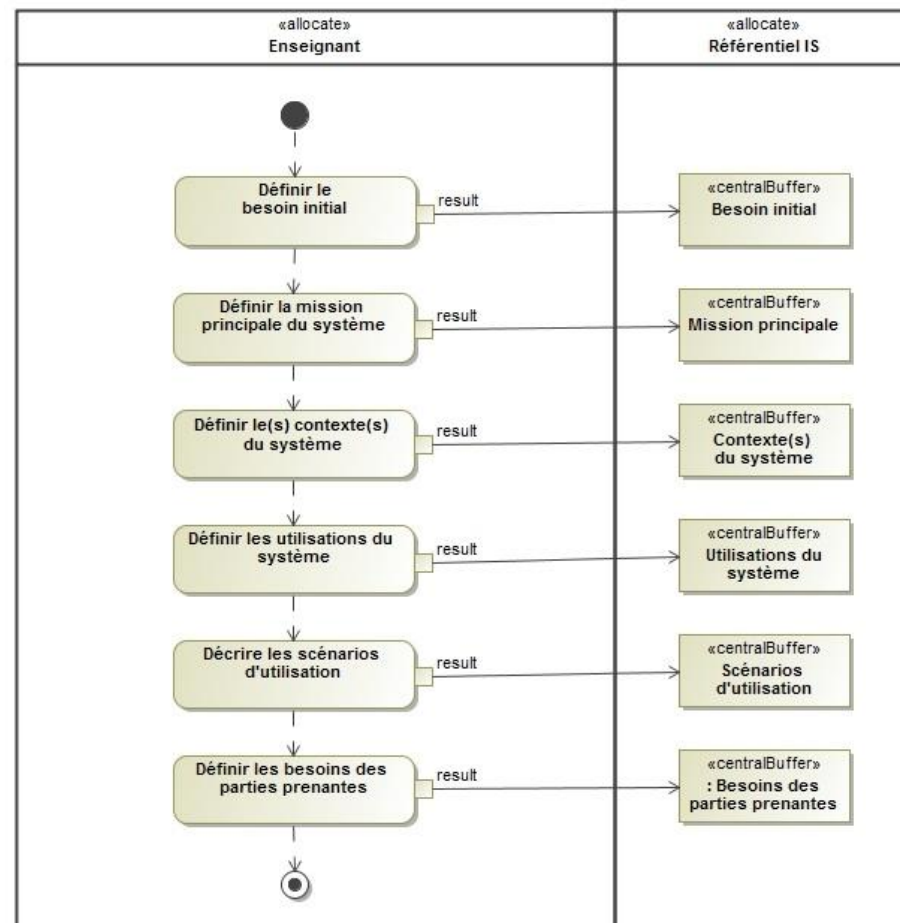
A ce niveau, et compte tenu des différentes erreurs précédentes, je me dis qu'il y a vraiment besoin de bien exprimer les besoins et de bien définir les contextes et utilisations, afin de bien décrire le système et de dissiper les éventuelles ambiguïtés.



# Définition des besoins

# Déroulement

Les étapes qui suivent reprennent les différentes activités telles que définies dans « IS et SysML dans l'EN » [2] :



# Expression du besoin initial / mission principale du système

- **Pourquoi mon système est-il justifié (Quelle est sa finalité) ?**

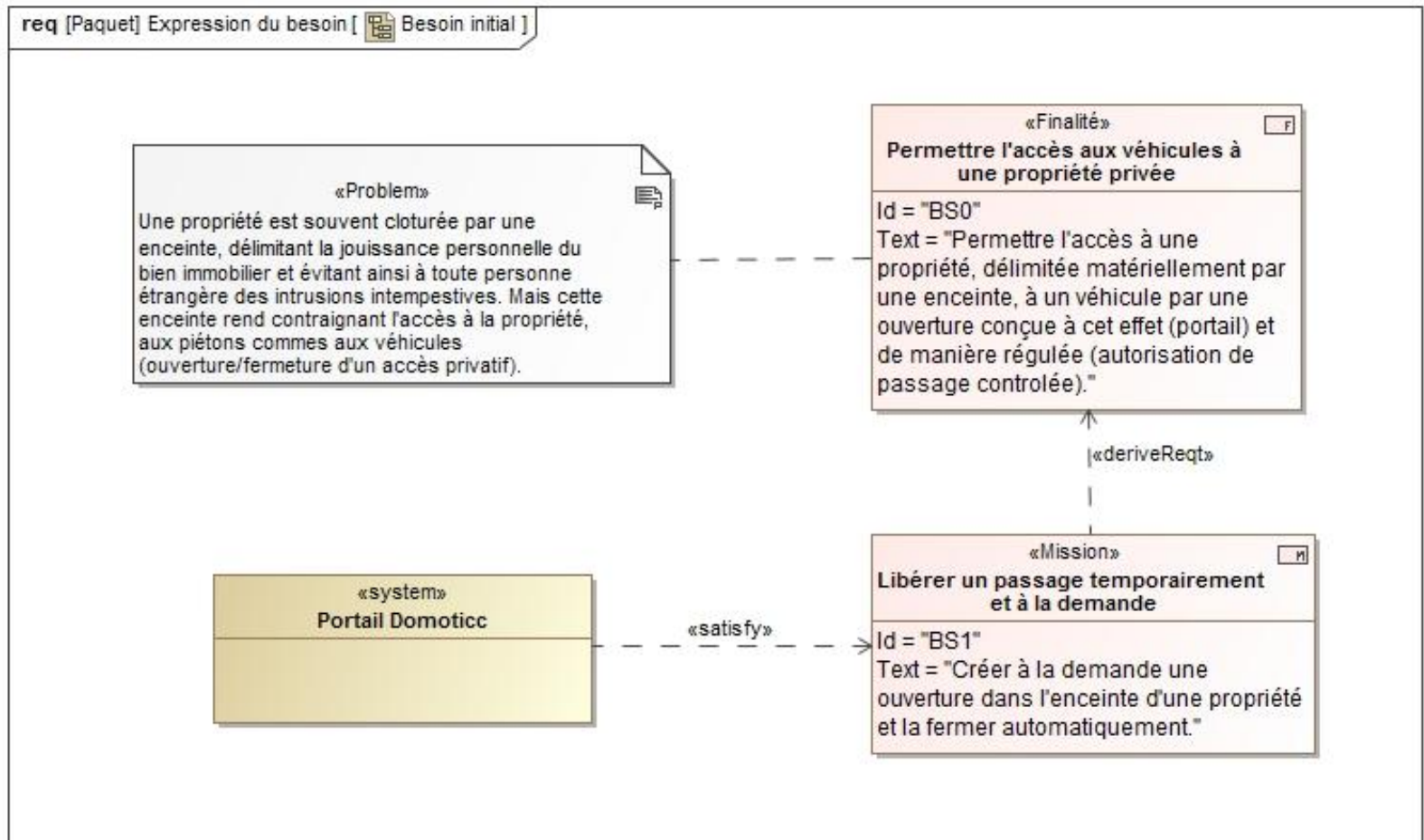
Parce qu'une propriété délimitée matériellement par une enceinte doit être accessible à ses habitants comme à ses invités, qu'ils soient piétons ou véhiculés.

- **A quoi sert-il (Quelle est sa mission) ?**

Libérer un passage temporairement et à la demande, soit en d'autres termes créer à la demande une ouverture dans l'enceinte d'une propriété et la fermer automatiquement.

# Expression du besoin initial / mission principale du système

Soit en formalisme SysML (RD) :



# Aparté sur la définition même du système...

Certains pourraient me dire : « *Oui mais moi, **mon système c'est juste le système automatisé**, qui motorise le portail. Pas le portail dans sa globalité.* » (c'est du vécu, je le retranscris tel quel).

Soit. Dans ce cas, il est évident que la finalité du système ainsi que sa mission ne sont plus les mêmes. Cela pourrait être (par exemple) :

- **Finalité** : réduire la pénibilité et les contraintes liées à l'actionnement d'un portail manuel ;
- **Mission** : automatiser (motoriser) l'actionnement d'un portail.

Mais attention toutefois, car en me plongeant dans la norme NF (pour le besoin) on peut y lire ceci :

*« Au regard de la norme EN 13241-1 une porte automatique certifiée est un produit certifié dans son ensemble, elle est considérée comme une machine.*

*Il est donc interdit de poser une motorisation agréée CE sur un portail fabriqué à titre isolé. De même il est interdit de poser une motorisation de marque X agréée CE sur un portail ou une porte manuelle de marque Y agréée CE, sauf si ces deux produits ont été certifiés ensembles par un organisme notifié (Cascading). »*

Cette remarque n'a d'autre but que de vous faire comprendre, au besoin, l'importance d'une bonne définition de la finalité et de la mission d'un système (c'est en quelque sorte la « bête à cornes » légèrement décornée, le reste est dans le contexte et l'UCD\*).

\* : Voir la liste des acronymes employés en fin d'article

# Définition du contexte en phase d'exploitation

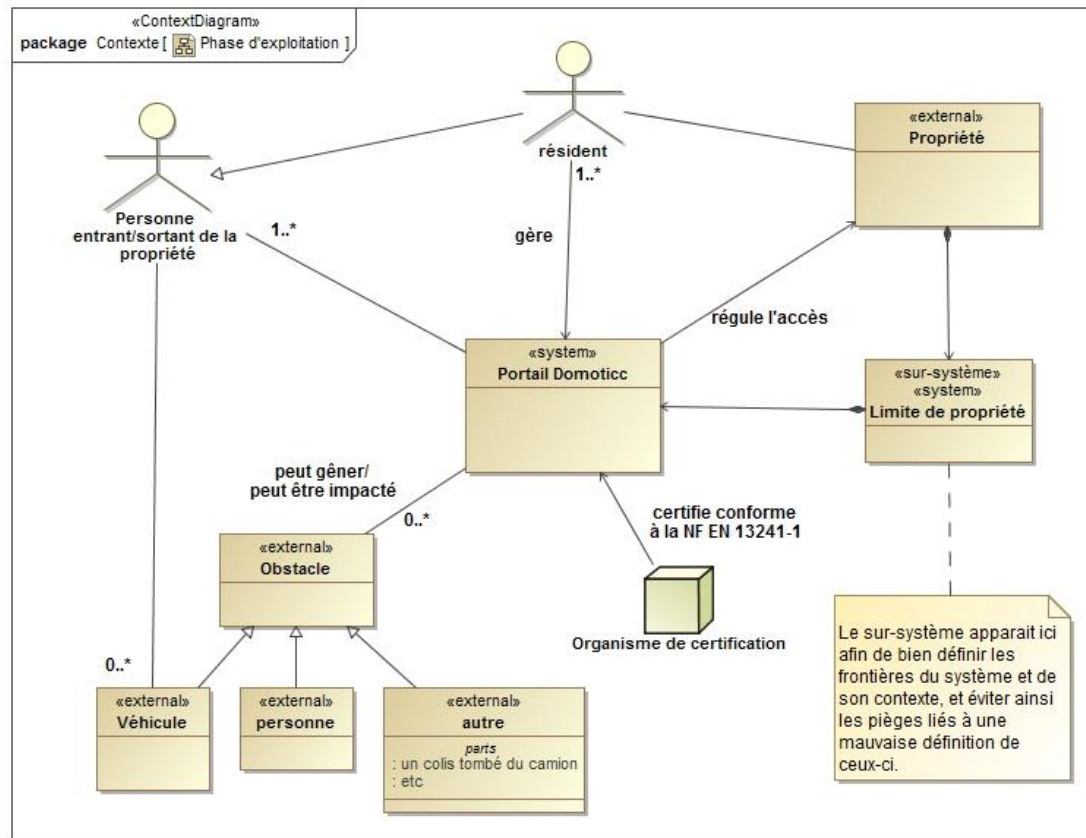
- **Identification des parties prenantes :**
  - la(les) personne(s) accédant à la propriété ;
  - Le(s) résident(s) : concerné(s) directement et gérant l'accès à la propriété ;
  - l'organisme de certification (fait apparaître la norme NF EN).
- **Identification des éléments du contexte :**
  - la propriété, et en particulier sa limite (sur-système) ;
  - les éventuels obstacles :
    - le véhicule lui-même (si personne véhiculée) ;
    - la personne elle-même (si piéton), ou une profitant du passage ;
    - tout objet se trouvant de manière inopinée dans la zone d'action (chargement tombé...).

**Remarque de l'auteur :** Peut-être estimez-vous qu'il y manque des éléments, mais chacun saura compléter un diagramme à son usage (comme le réseau électrique pour les besoins en énergie, les éléments climatiques pour les besoins de résistance aux intempéries, l'environnement pour le niveau sonore du système ; bien que la plupart de ces éléments se retrouvent dans la norme NF EN...).



# Définition du contexte en phase d'exploitation

Soit en formalisme SysML :

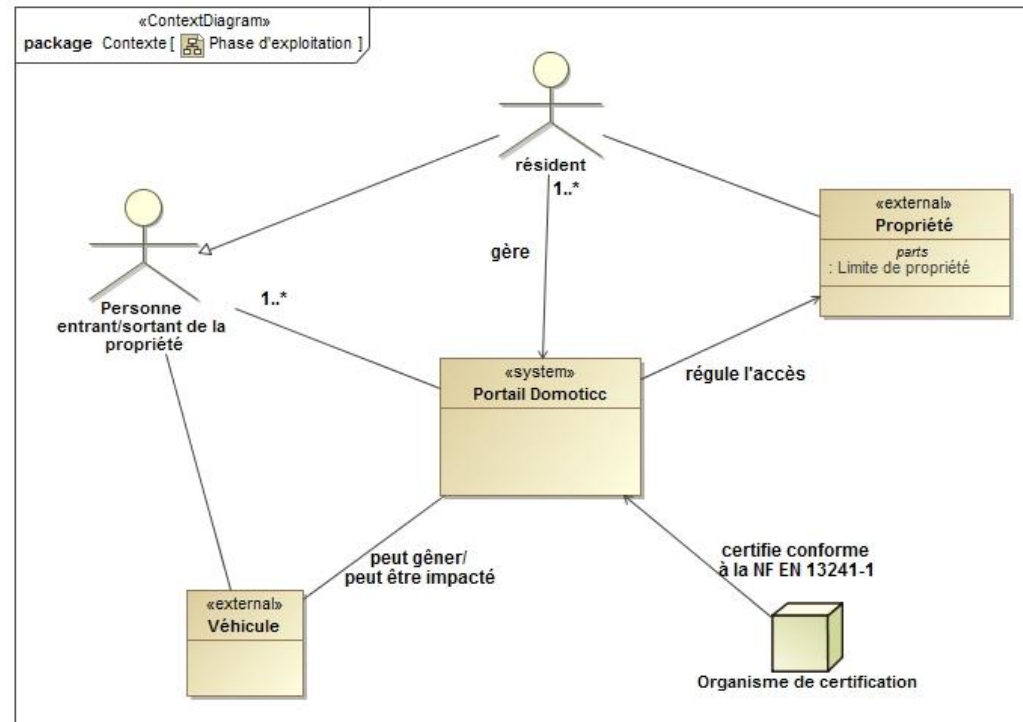


Mais peut-être le trouvez-vous trop généraliste...



# Définition du contexte en phase d'exploitation (exploitable)

...on peut épurer aux stricts besoins nécessaires :



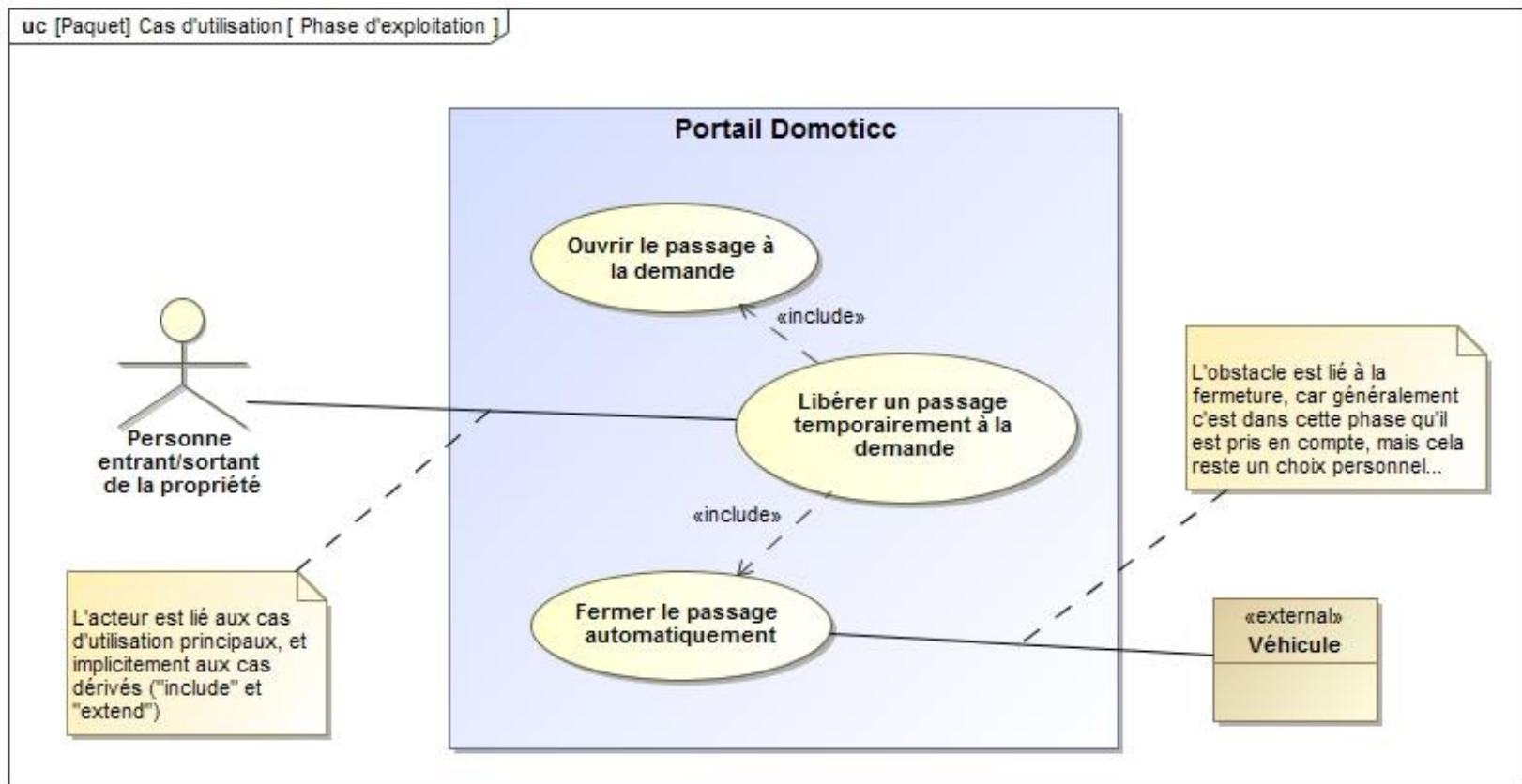
Chacun limitera sa description à son étude, en gardant à l'esprit que ne sont nécessaires que les éléments apportant une information utile (ici, l'obstacle est le seul véhicule).

# Utilisations du système

- La mission principale se retrouve généralement dans le cas d'utilisation principal :
  - « **Libérer un passage temporairement et à la demande** ».
- On peut de plus, à ce niveau déjà, distinguer les 2 sous-missions du système qui sont :
  - « **Ouvrir le passage à la demande** » ;
  - « **Fermer le passage automatiquement** », où là l'obstacle intervient de manière évidente (attention : le terme « automatiquement » désigne bien ici le mode opératoire, et non le fait que le système est automatisé).

# Utilisations du système

Soit en formalisme SysML:



# Utilisations du système (suite)

Mais **sont-ce bien là des cas d'utilisation** ? Et y en a-t-il d'autres ?

- « **Ouvrir le passage à la demande** » en est un, sans aucun doute, et le service est bel et bien rendu à la personne empruntant le passage ;
- « **Fermer le passage automatiquement** » est par contre **orphelin** : aucun acteur n'y est relié, car finalement la personne circulant ne participe pas à ce cas, ce qui pose problème. Mais ce cas est là pour retranscrire le caractère temporaire de l'ouverture créée, service qui est bien rendu au(x) résident(s) désireux de ne pas laisser le passage ouvert indéfiniment ;
- Et c'est tout.

Attention à ne pas rentrer dans les travers de la décomposition fonctionnelle (certains pourraient être tentés par « signaler l'ouverture » ou encore « détecter les obstacles », par expérience cette remarque est loin d'être anodine...).

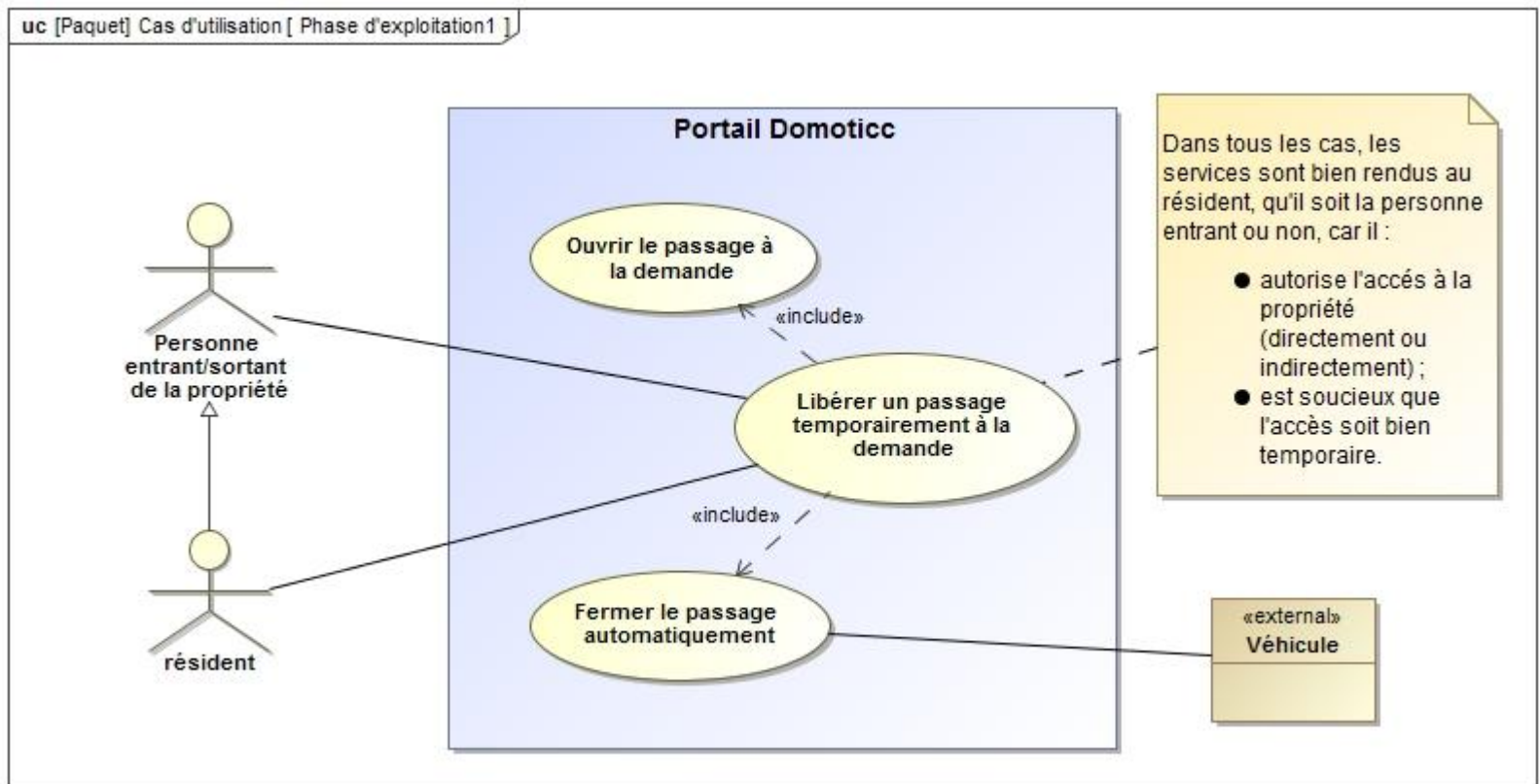
**Posez-vous toujours des questions du type :**

- « Est-ce bien un service rendu par mon système en autonomie ? » ;
- « A qui ce service est-il rendu ? » (tiens, un reste de ma bête à corne...) ;
- « Quel est l'élément (événement) déclencheur ? » ;
- « Quel autre(s) élément(s) (acteur) participe(nt) à ce service ? » ;
- « Quel(s) scénario(s) puis-je envisager ? ».

Je ne peux que vous recommander, encore une fois, la lecture de l'article de Pascal Roques [1].

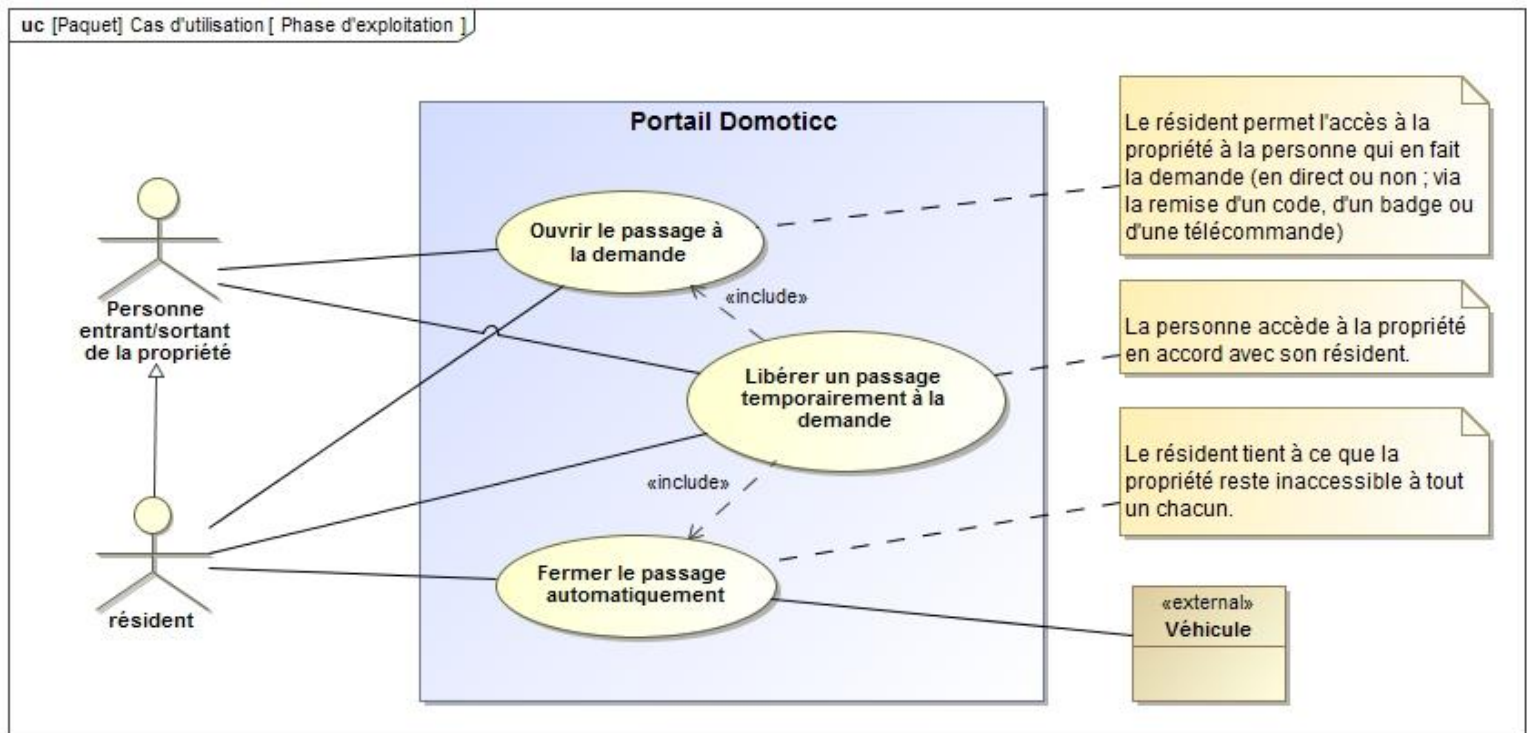
# Utilisations du système (V2)

Après modification (les choses n'étant pas linéaires) :



# Utilisations du système (exploitable)

Le même épuré et adapté aux besoins :



(j'ai mis volontairement tous les liens, cela peut parfois rendre les diagrammes plus lisibles...)



# Décrire les scénarios d'utilisations

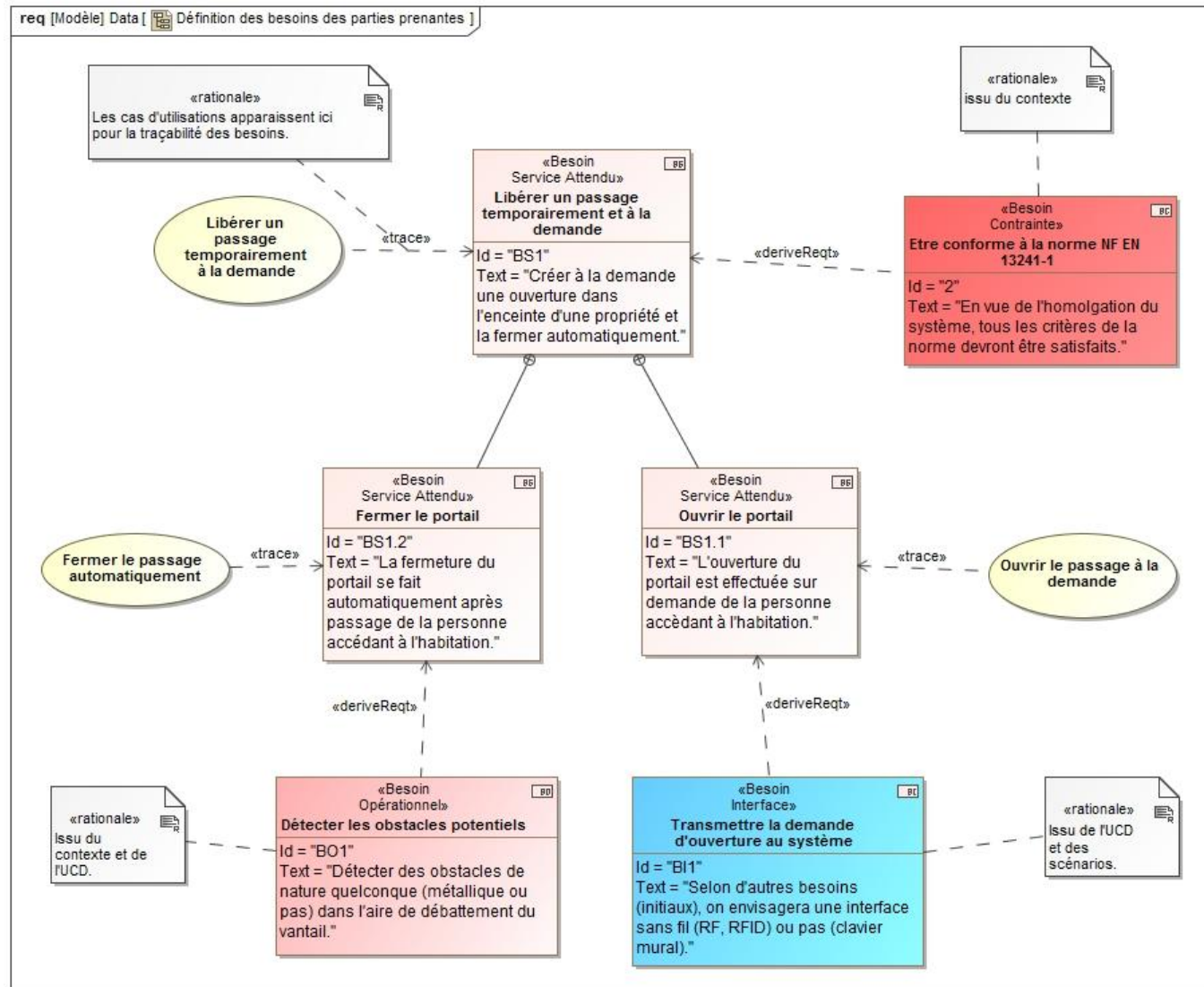
- A ce stade, une description textuelle peut amplement suffire, surtout que les diagrammes de séquence nécessaires seront eux issus d'une analyse plus poussée.
- Scénario nominal du cas d'utilisation « libérer un passage temporairement à la demande » :
  - Une personne véhiculée se présente devant le portail de la propriété ;
  - Elle demande une ouverture du portail ;
  - Le portail s'ouvre ;
  - La personne entre ;
  - Une fois la personne entrée, le portail se ferme, en gérant les obstacles éventuels.

# Définition des besoins des parties prenantes

- Sur la base des documents précédemment établis, on mène une analyse pour susciter les différents besoins non présents initialement.
- En toute rigueur il faudrait la mener de manière la plus complète possible. Pour ma part je l'ai mené partiellement à titre d'illustration du processus technique complet, en commentant légèrement celle-ci, car au final ce diagramme n'entre pas dans l'exploitation pédagogique attendue (rappel : seuls sont utiles les cas d'utilisation, une séquence pour finalement exploiter un diagramme d'états).



# Diagramme (très) partiel de DBPP





# Analyse des exigences

# Déroulement

Le but ici n'est pas de reprendre toutes les étapes, telles que définies dans [2], mais d'arriver à l'obtention des états du système, à un certain niveau de raffinement.

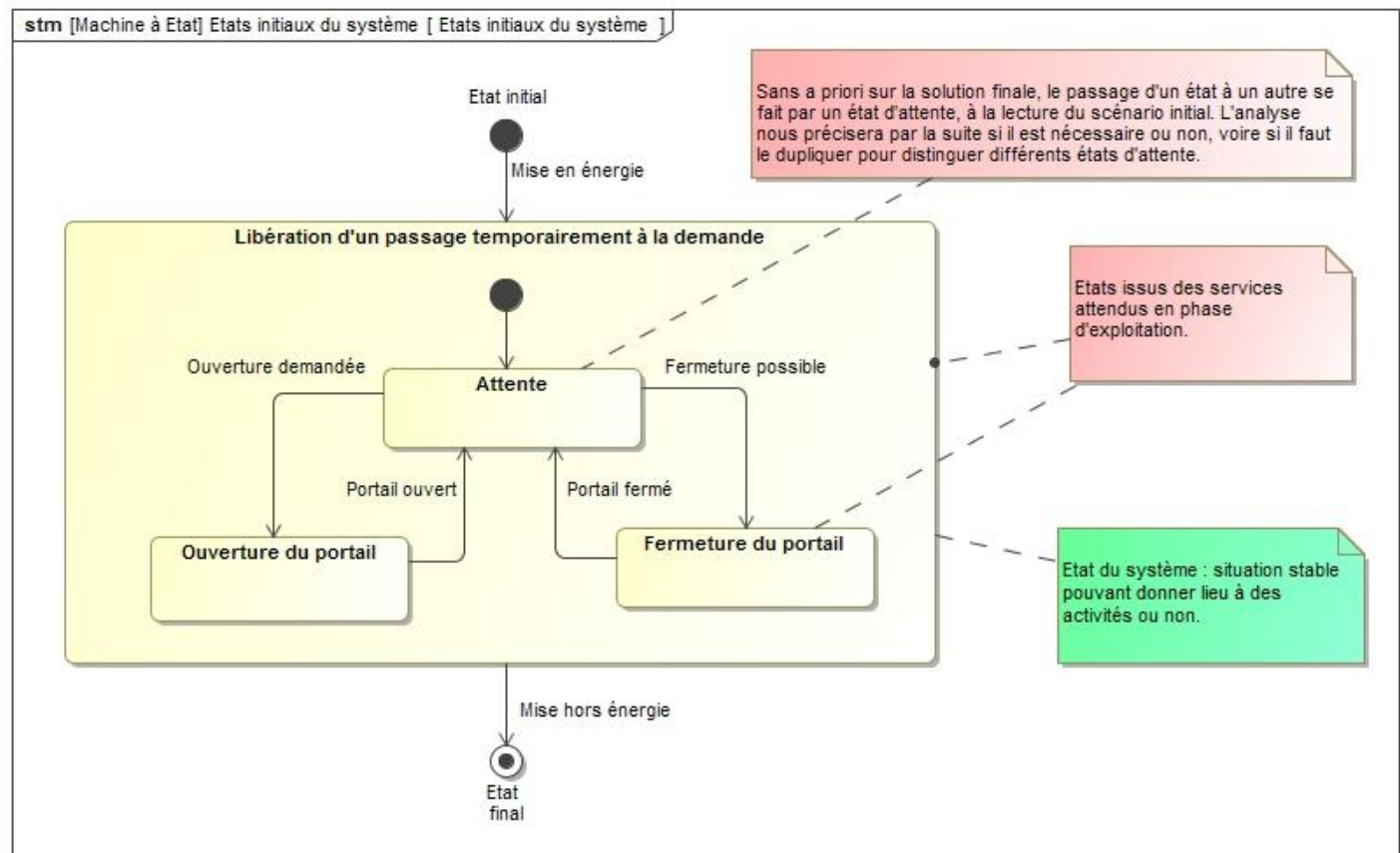
Un déroulement simplifié est alors le suivant :

- Obtention des états initiaux du système (directement issus de l'analyse précédente) : SMD\* initial ;
- Description des utilisations du système (sur la base des scénarios définis, complété des besoins nécessaires et des éléments disponibles) : SD\* ;
- Raffinement des états du système (sur la base de la description précédente) : SMD\* initial raffiné, voire AD\*.

\* : Voir la liste des acronymes employés en fin d'article

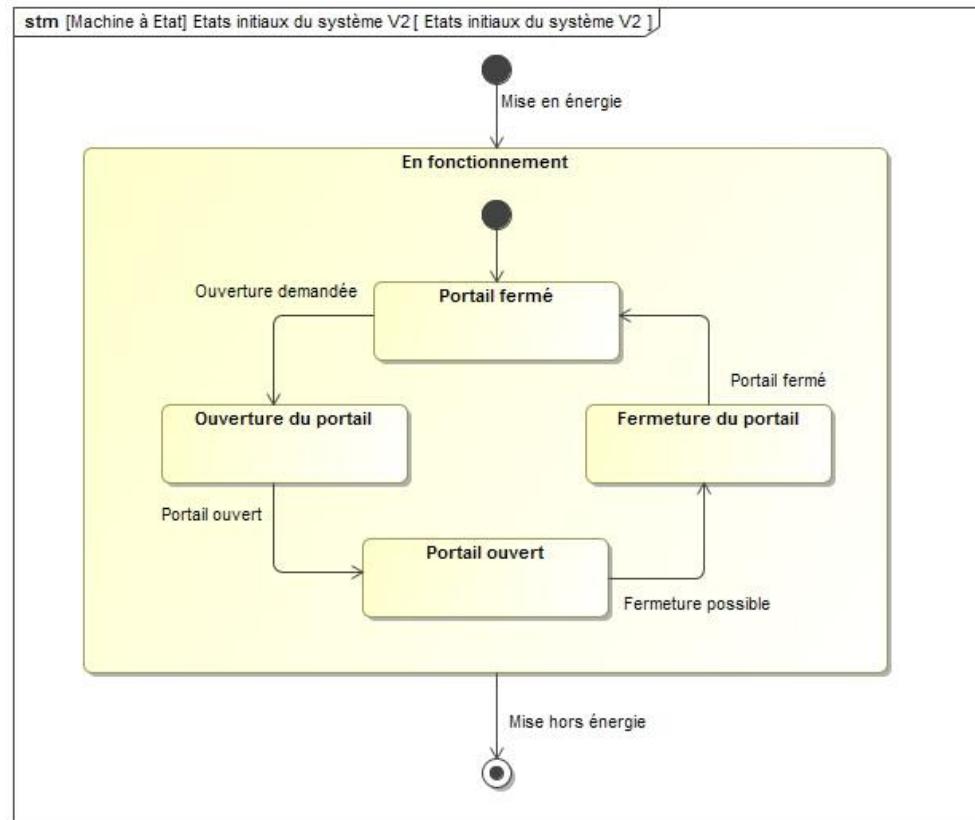
# Etats initiaux du système

Sur la base des cas d'utilisation (et donc des services attendus), on peut déduire ce diagramme initial :



# Etats initiaux du système (exploitable)

A la lecture de l'état d'attente, on distingue déjà 2 états bien distincts : « portail ouvert » et « portail fermé ». De plus, l'état composite est le seul lié à un fonctionnement, ce qui peut se simplifier :



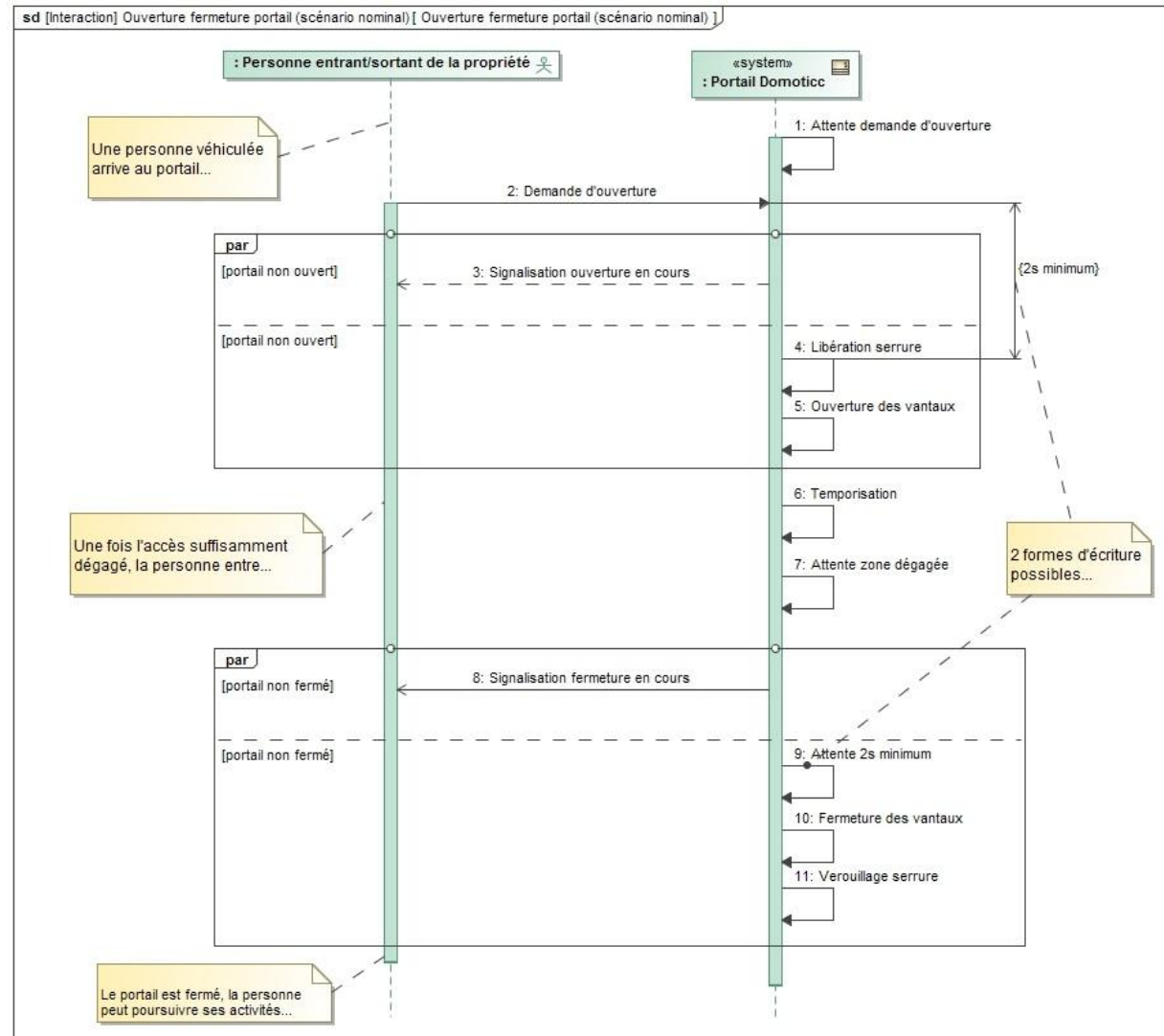
# Scénarios d'utilisations : forme textuelle

Scénario textuel du cas d'utilisation principal, sur la base du « passage d'un véhicule sans encombre avec attente de fermeture » :

- La personne cherchant à entrer demande l'ouverture du portail ;
- Sa demande est bien prise en compte par l'allumage d'un feu clignotant orange de signalisation jusqu'à l'ouverture totale ;
- S'en suit l'ouverture elle-même (attente de 2 s min, norme NF), avec une libération de serrure au préalable ;
- Lorsque le portail est ouvert la personne entre (remarque : elle pourrait ne pas attendre l'ouverture totale...) ;
- Une fois le portail ouvert, le système temporise (pour laisser au véhicule le temps de passer) ;
- Une fois la zone dégagée, le feu clignotant est allumé de nouveau jusqu'à fermeture totale, la fermeture se déclenche au bout de 2 s min, pour verrouiller la serrure et éteindre le feu (je vais vite...).

**Remarque de l'auteur** : faites-le d'abord par écrit, c'est un exercice de style mais ça laisse peut-être moins de place à l'interprétation. Choisissez de plus le scénario le plus « arrangeant », le but n'est pas d'en faire un algorithme (usage abusif des fragments combinés).

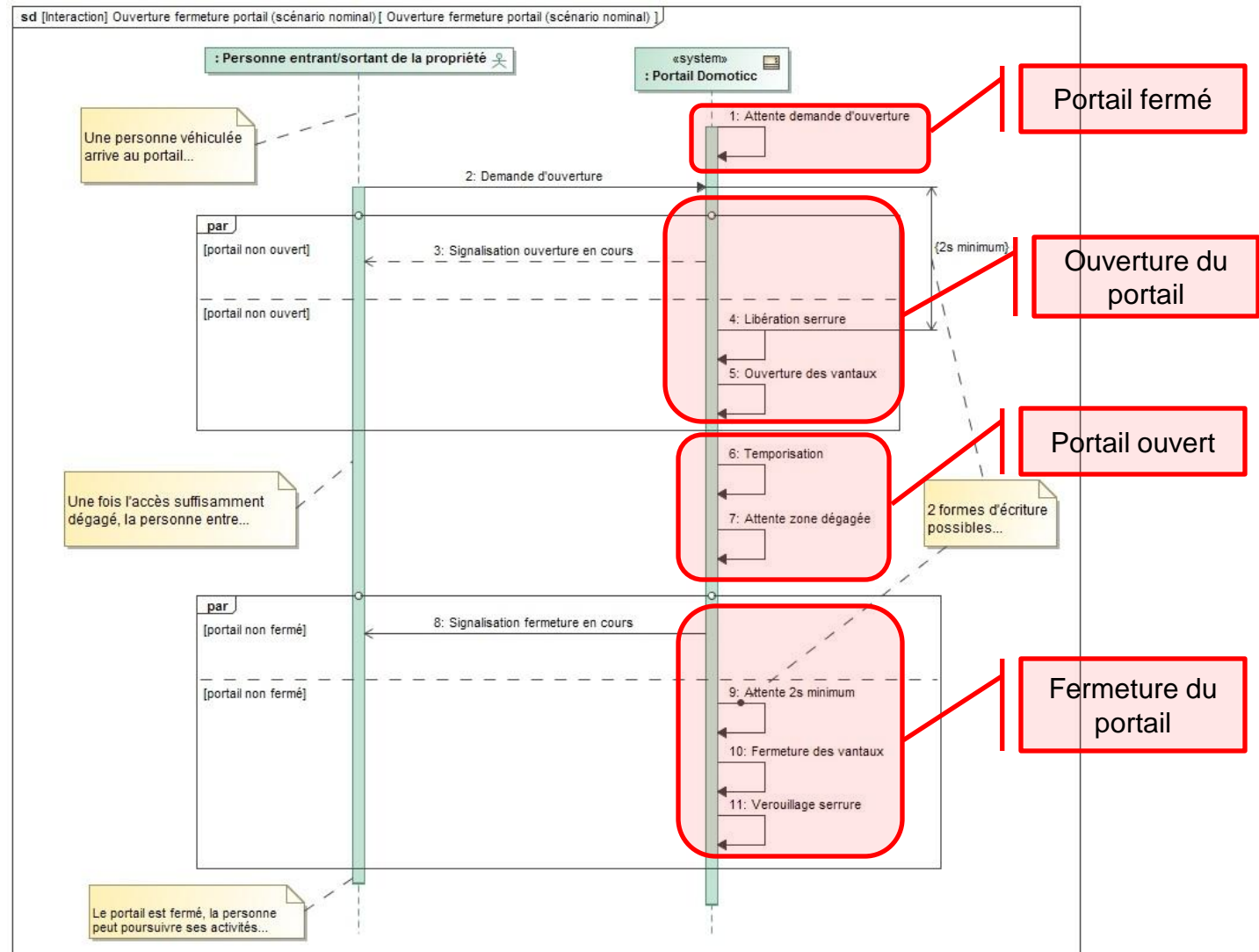
# Scénarios d'utilisations : SysML





# Scénarios : lien avec les états

On retrouve bien ici nos états initiaux :





# Justification des détails de la séquence (parallélisme, durées)

Afin de préciser les éléments de détails de la séquence j'ai mené ma petite analyse.

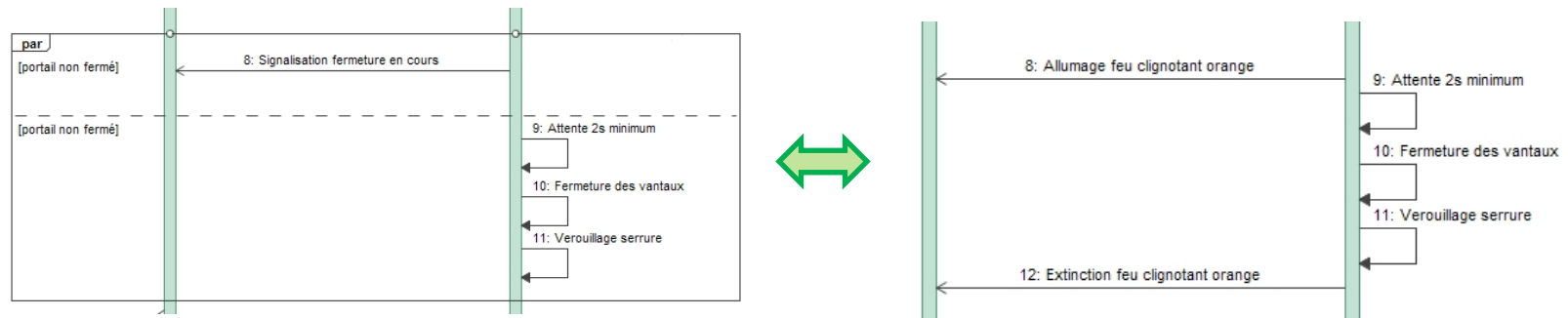
Si la norme NF EN 13241-1 a remplacé la norme française NF P 25-362, il n'en reste pas moins que la réglementation française précise les exigences de performances de cette norme :

- **Articles R 125-3-1 et R 125-4 du 27 juin 2006 du code de la construction :**  
« tout mouvement de la porte doit être signalé, tant à l'extérieur qu'à l'intérieur, par un feu orange clignotant visible de l'aire de débattement. La signalisation doit précéder le mouvement de la porte » → **Parallélisme, précedence**
- **Arrêté du 9 août 2006 article 3 :**  
« la signalisation du mouvement de la porte doit précéder d'au moins 2 secondes le mouvement de la porte. » → **Durée de 2 s minimum**

Mes choix sont ainsi justifiés, sans connaître le système ni la solution technique retenue (nous sommes dans l'espace du problème, et non de la solution). Il y a néanmoins un choix architectural qui a déjà été fait et qui n'aura pas échappé à votre vigilance : nous sommes sur une solution à battant. Je n'ai fait que reprendre les données du collègue et il faudrait en toute rigueur justifier ce choix...

# Scénarios d'utilisation (suite)

Maintenant, il est toujours possible, pour des raisons de simplification de la séquence, de représenter la signalisation parallèle par une action d'allumage initiale et d'extinction finale du feu :

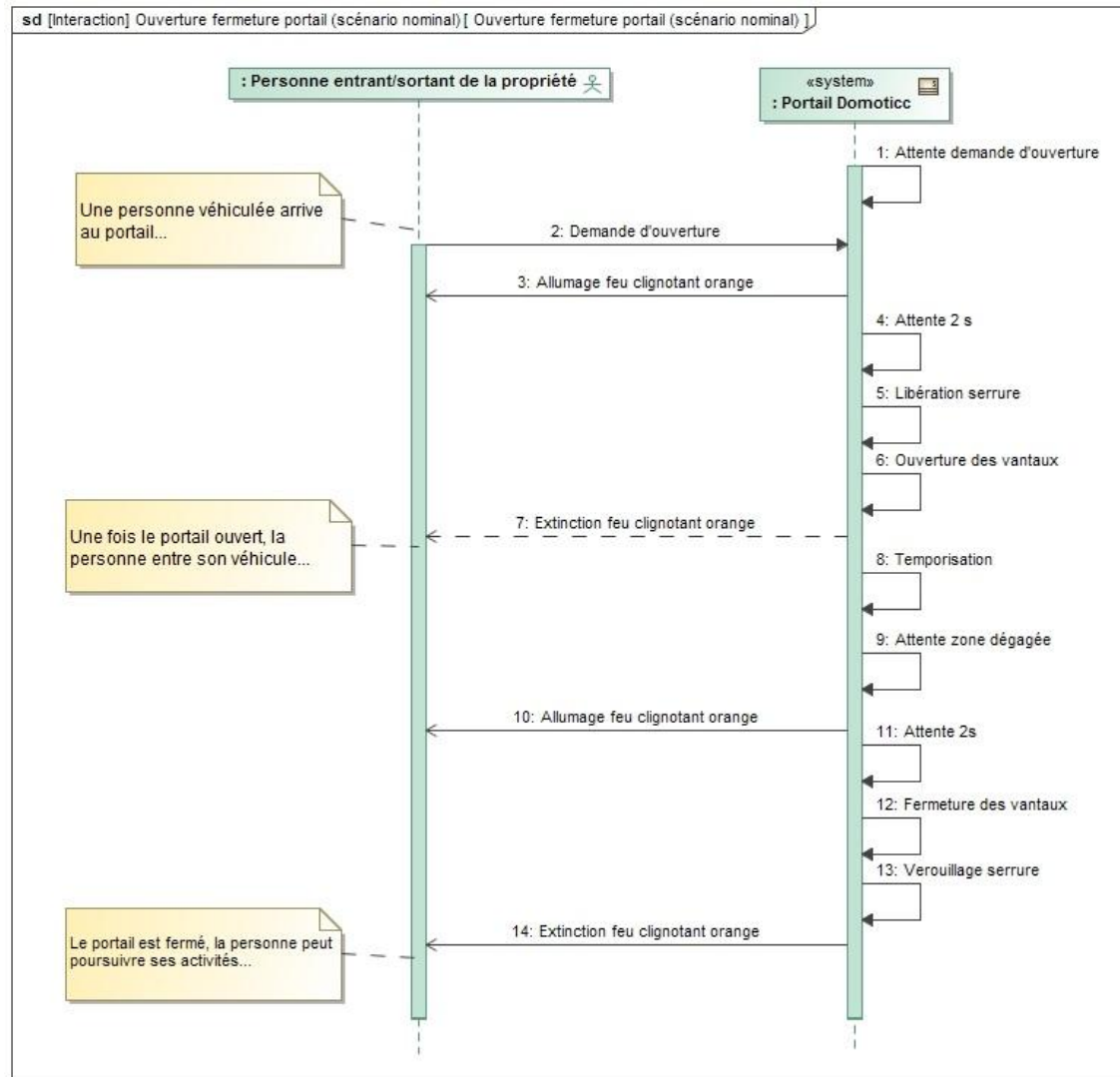


Cela implique ici des choix d'interface, sur la commande du feu de signalisation, mais il est fort à parier qu'elle soit de ce type...

## Remarques :

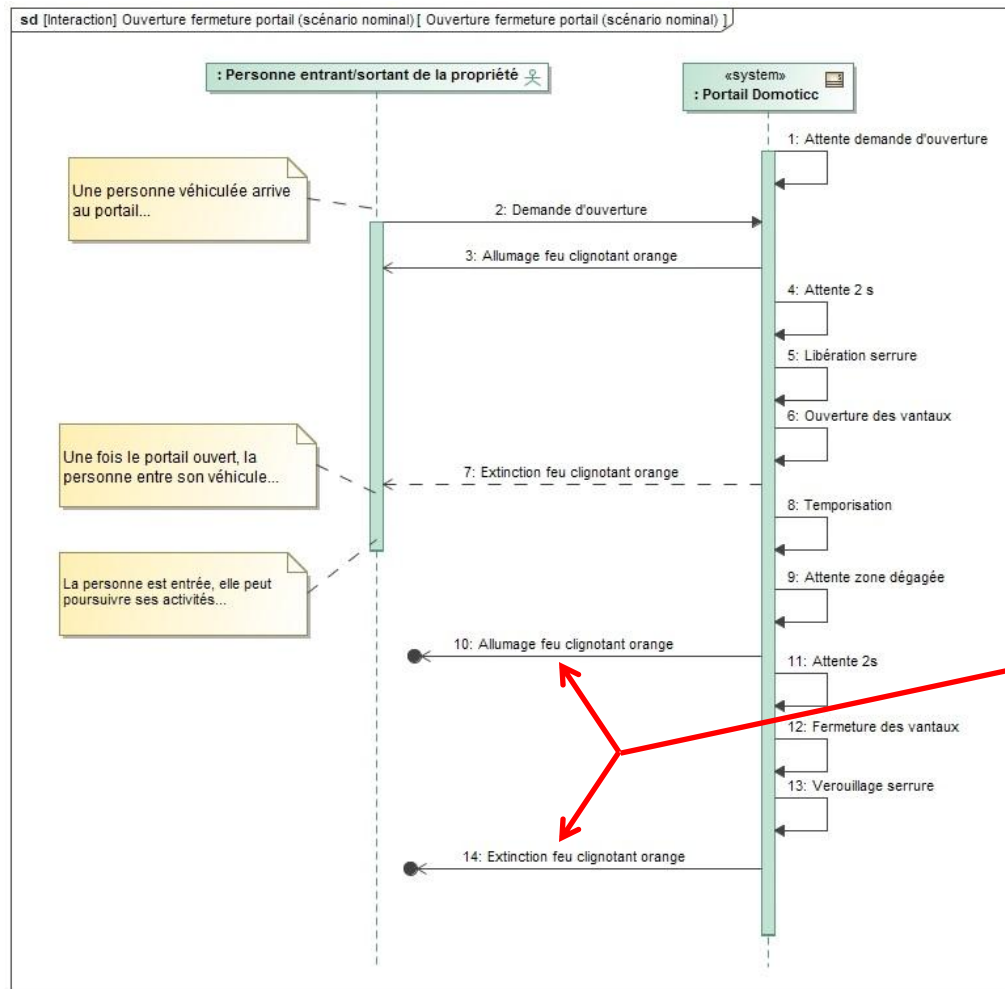
- Du coup, on synchronise plutôt sur l'extinction du feu à la fin de l'ouverture, moment qu'attend la personne pour entrer. La signalisation en fermeture n'est pas synchronisée, la personne pourrait ne pas être présente qu'elle s'effectuerait quand même ;
- Un message est porteur de plusieurs informations : un besoin d'interface entre le système et les éléments environnants et/ou un besoin opératoire et/ou un besoin fonctionnel. Si ce dernier n'apparaît pas explicitement comme une activité interne (du type message réflexif), il est bel et bien là (la ligne de vie active suffit à le signaler).

# Scénarios d'utilisation (exploitable)



# Scénarios d'utilisation bis (exploitable)

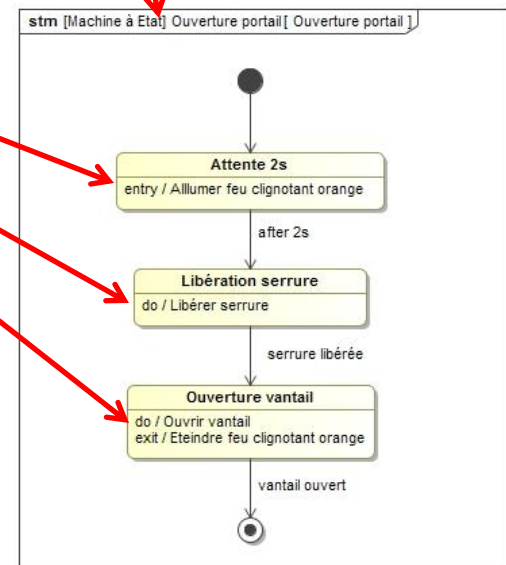
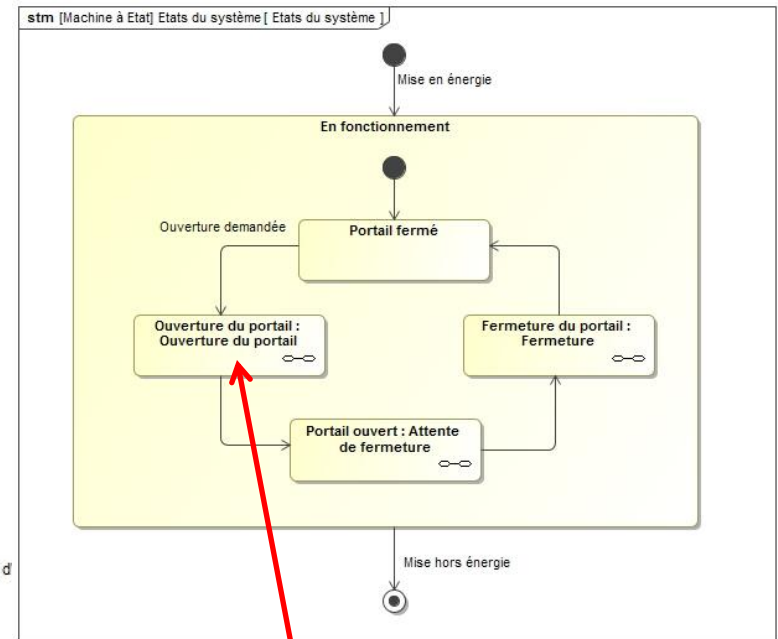
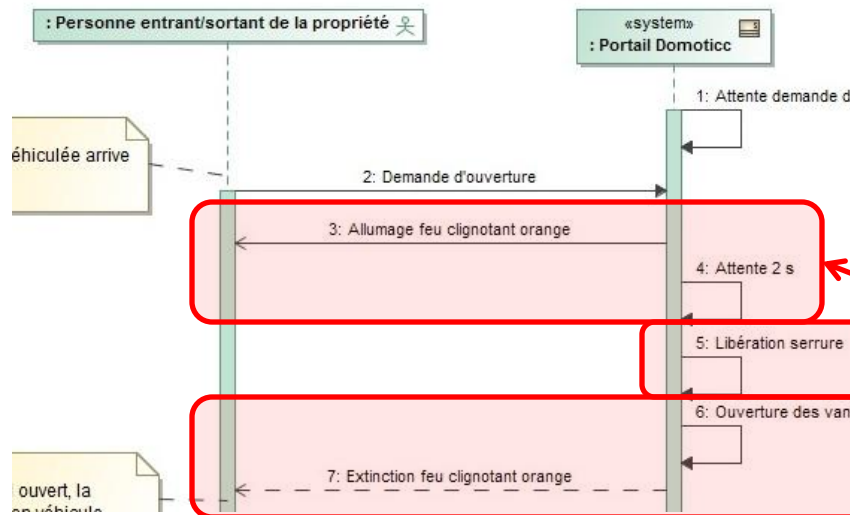
Et si la personne ne se préoccupe pas de la fermeture :



J'utilise ici le **message perdu** [3] [4] : message asynchrone à un participant inconnu ou non spécifié. Je trouve personnellement cela très pratique, pour ce genre de situation spécifique. A voir à l'usage...

# Raffinement des états du système

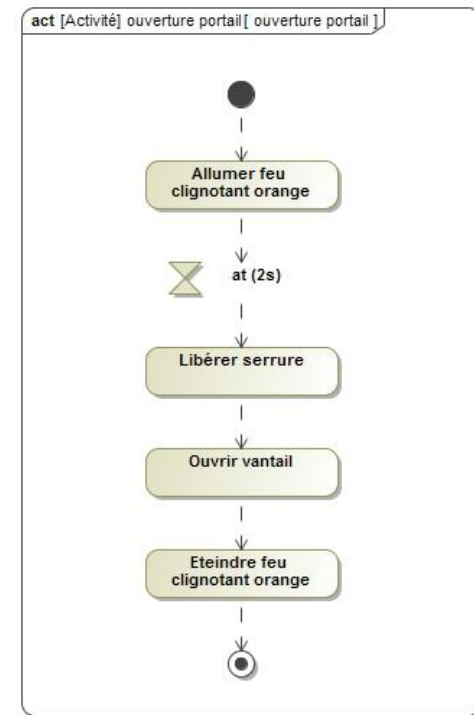
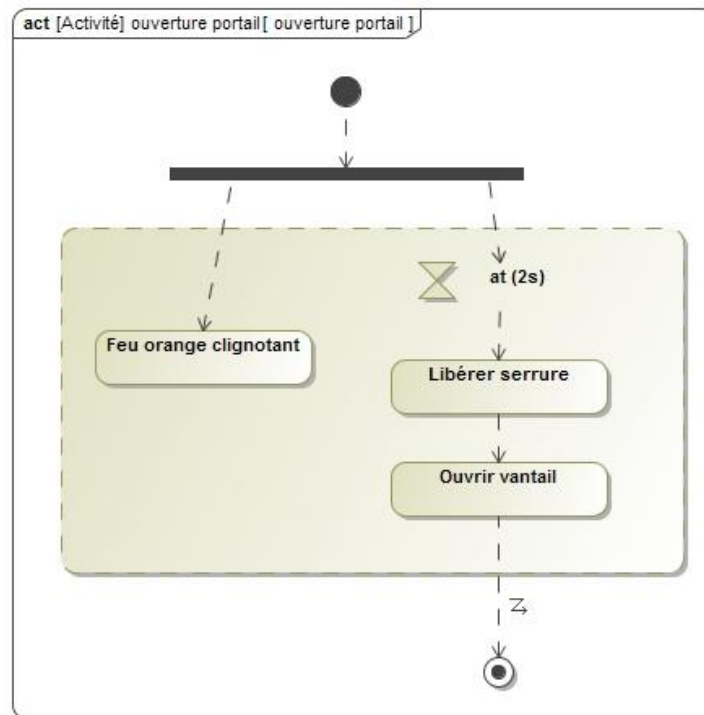
Finalement, sur la base du diagramme de séquence précédent, on peut maintenant raffiner notre modèle (remarque : nous ne sommes toujours pas dans la solution...) :



# Raffinement des états du système Complément

L'analyse précédente a été menée avec uniquement des SMD, comme préconisée dans [2]. Néanmoins, il est possible, voire plus naturel, d'effectuer un regroupement d'activités d'un diagramme de séquence sous forme de diagramme d'activités (AD).

En voici 2 possibilités, version parallèle et non parallèle, pour l'ouverture du portail :



# Conclusion

Dans une démarche de rétro-ingénierie système, on aura établi les BDD et IBD du système, et donc on pourra compléter le diagramme d'états en conséquence (sur les opérations/activités à réaliser, en fonction des transitions à valider, conformément aux ports et flux de l'IBD).

Toujours dans cette même démarche, cet exemple montre qu'il est possible d'obtenir un minimum d'éléments utiles dans le formalisme SysML pour peu qu'on se pose les bonnes questions, dans une démarche cohérente. Je ne connais pas le portail Domoticc et, partant de rien, je suis arrivé à un ensemble de définitions et spécifications non négligeables.

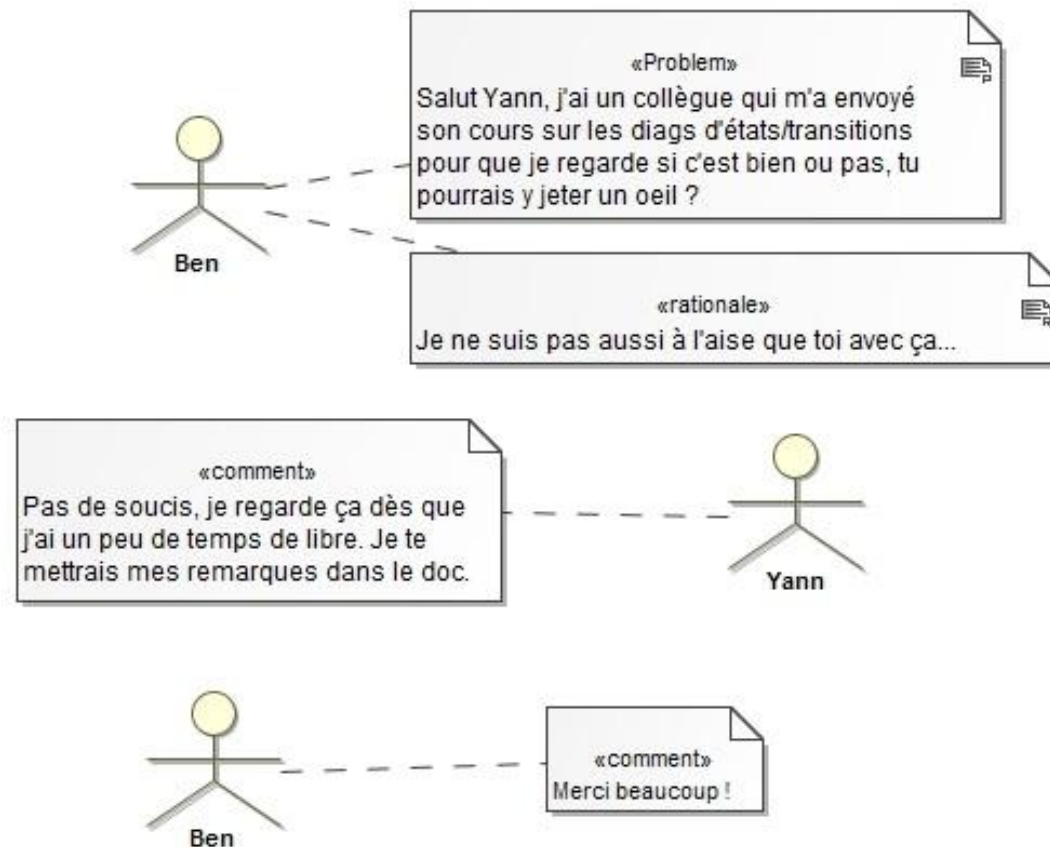
Je ne néglige pas le travail et l'expertise que cela suppose, je montre juste ici que c'est possible de donner du sens aux choses...



# Epilogue

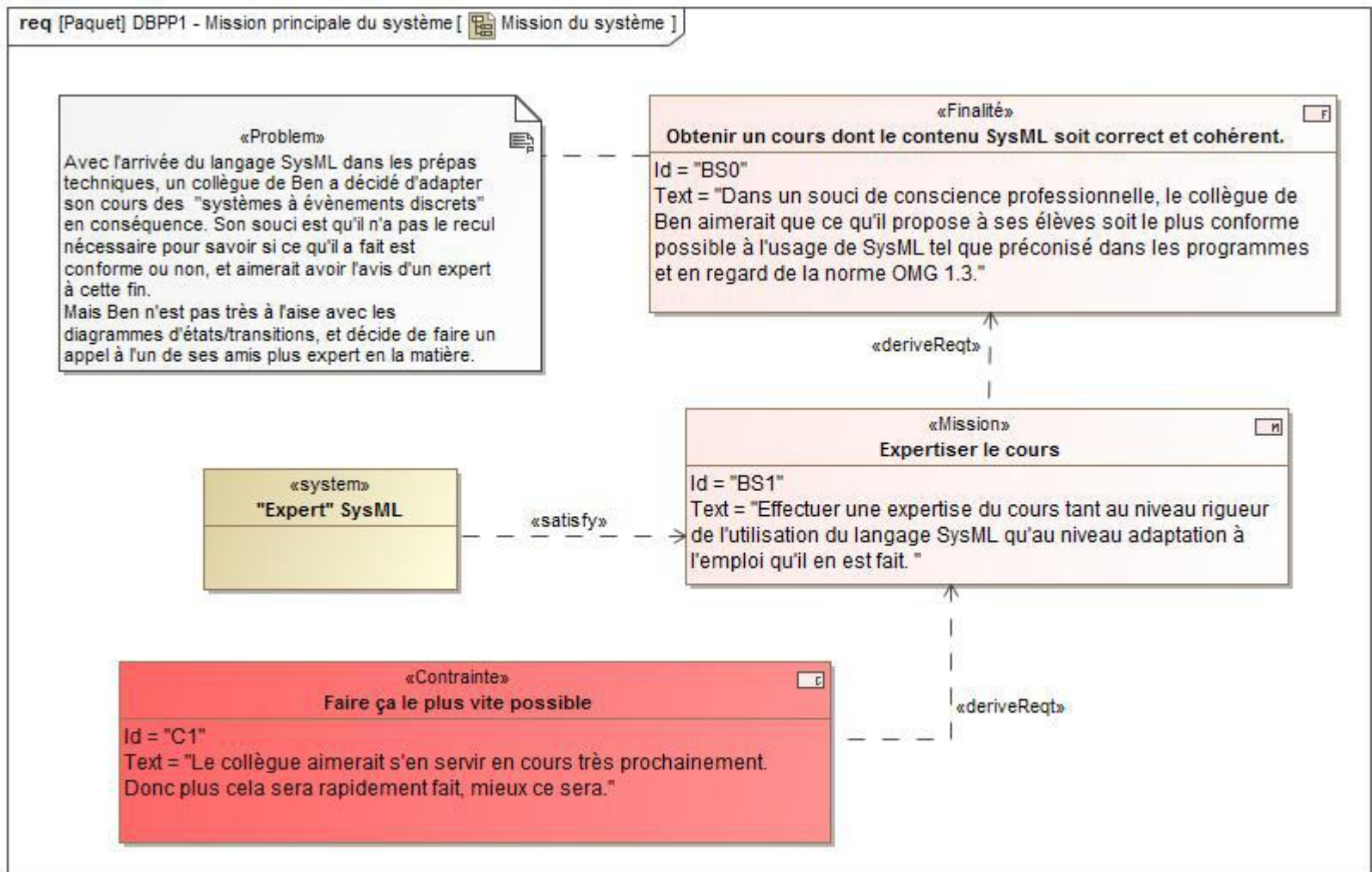


# Mais au fait , tout ça vient d'un besoin initial...

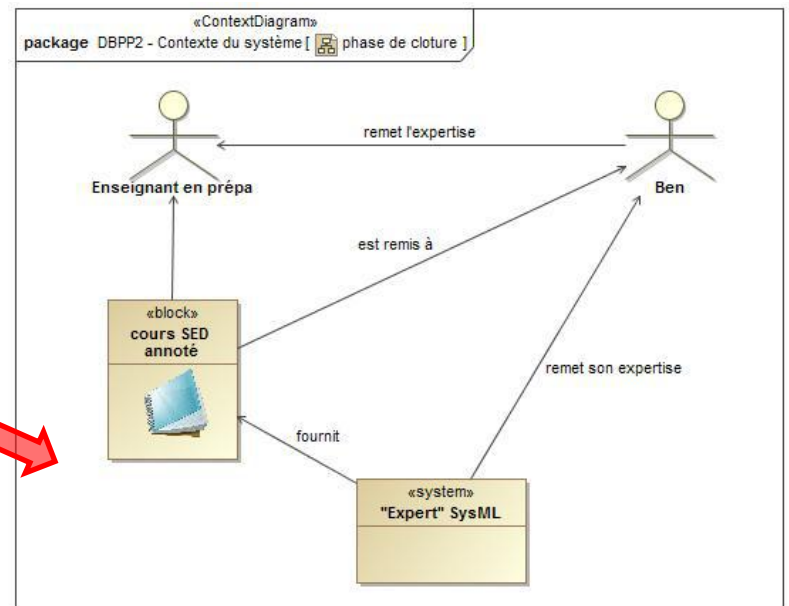
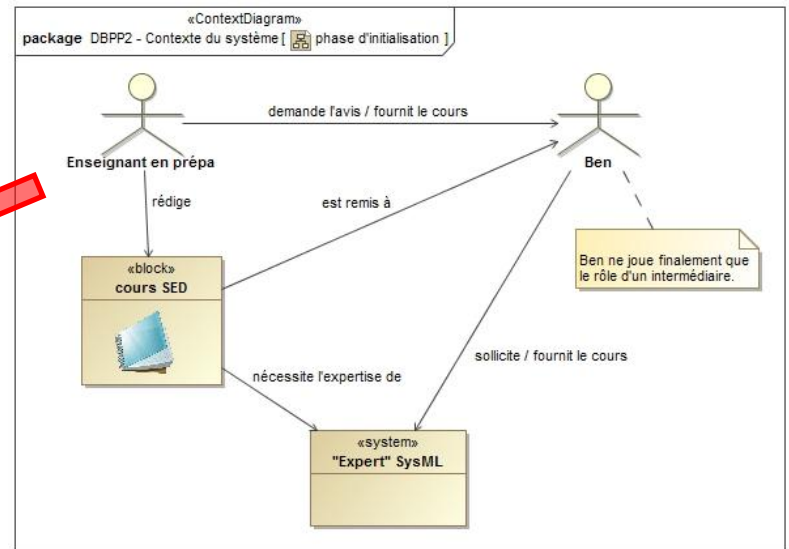
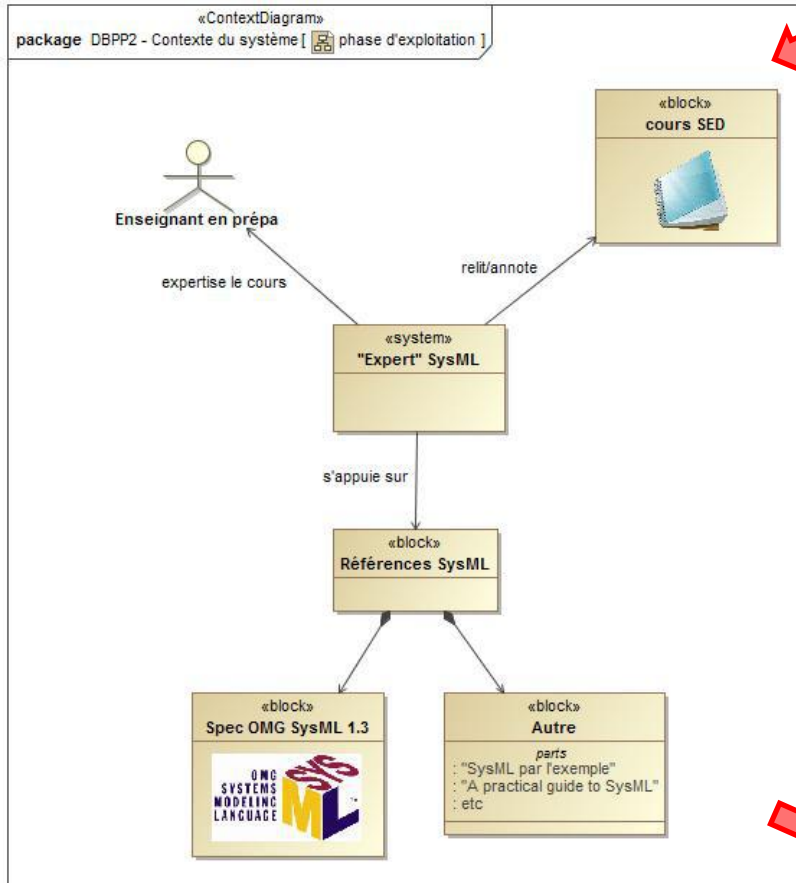


Et si j'étais moi-même un système ?

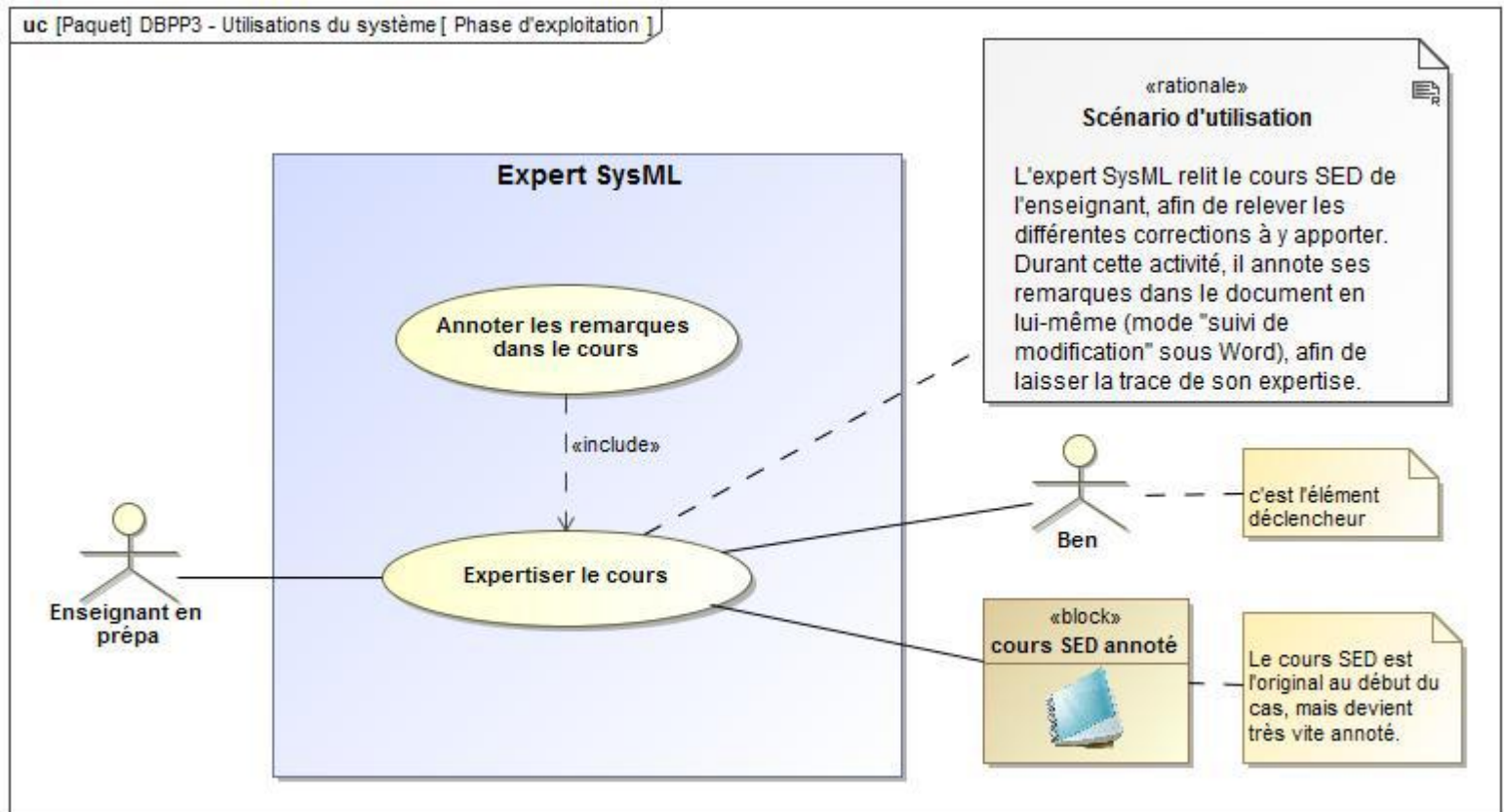
# J'ai une mission justifiée par une finalité...



# ...évoluant dans différents contextes...



# ...et bien sûr rendant des services !



# Conclusion

J'ai cherché ici à illustrer la même démarche sur un autre type de système, qui s'apparente plus à un « système pour faire », système à dominante organisationnelle (qui s'apparente au projet), qu'au « système à faire », système à dominante technologique (voir [5]).

Il est fort à parier que selon les cas, la description SysML d'un système se fera préférentiellement selon cet angle...

# Acronymes employés pour les diagrammes SysML

- **RD** : « Requirement Diagram »  
→ diagramme d'exigences
- **UCD** : « Use Case Diagram »  
→ diagramme de cas d'utilisation
- **SD** : « Sequence Diagram »  
→ diagramme de séquence
- **SMD** : « State Machine Diagram »  
→ diagramme d'états/transitions
- **AD** : « Activity Diagram »  
→ diagramme d'activités
- **BDD** : « Block Definition Diagram »  
→ diagramme de définition de blocs
- **IBD** : « Internal Block Diagram »  
→ diagramme de blocs internes



# Bibliographie

- [1] : « Cas d'utilisations : bonne et mauvaise utilisation ! », Pascal Roques, Valtech Training, 2006 ;
- [2] : « Ingénierie Système et SysML dans l'Éducation Nationale », production du groupe de travail IS & SysML dans l'EN, une collaboration EN, AFIS et AIP-Priméca, à paraître en 2014 ;
- [3] « A practical guide to SysML », S. Friedenthal/A. Moore/R. Steiner, éditions MK/OMG Press, 2008, p. 222 ;
- [4] : « OMG SysML version 1.3 », juin 2012, p.122.
- [5] : « Découvrir et comprendre l'Ingénierie Système », Groupe de travail IS (AFIS), 2009