

Le SysML au collège

Le SysML, c'est quoi ?

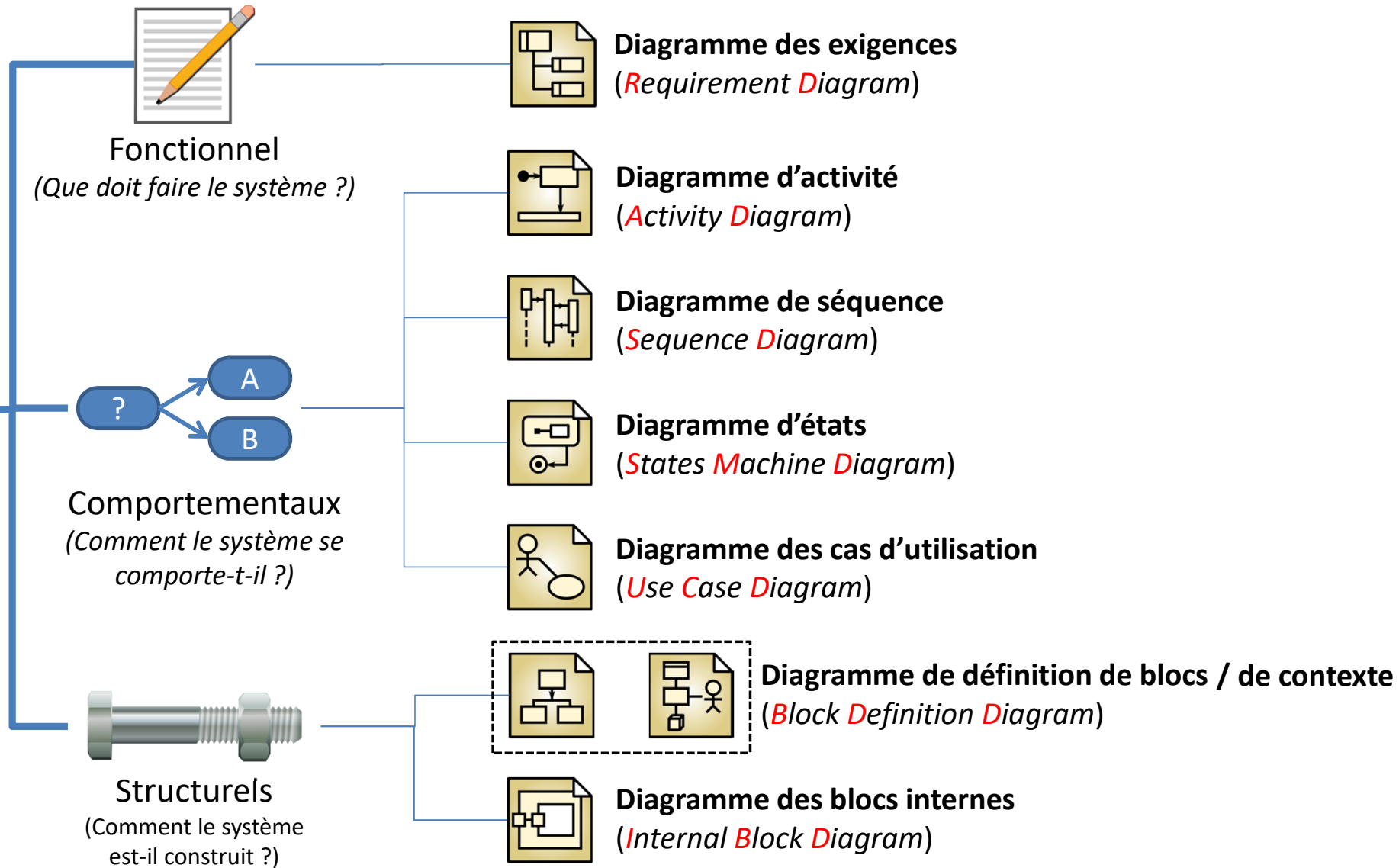
SysML = **S**ystems **M**odeling **L**anguage

Langage de **M**odélisation de **S**ystèmes



C'est un **outil** qui permet de **représenter** des **objets techniques**, sous forme de schémas appelés **diagrammes**.







Exigences



Cas d'utilisation



Séquence



Définition de blocs



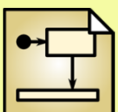
Contexte



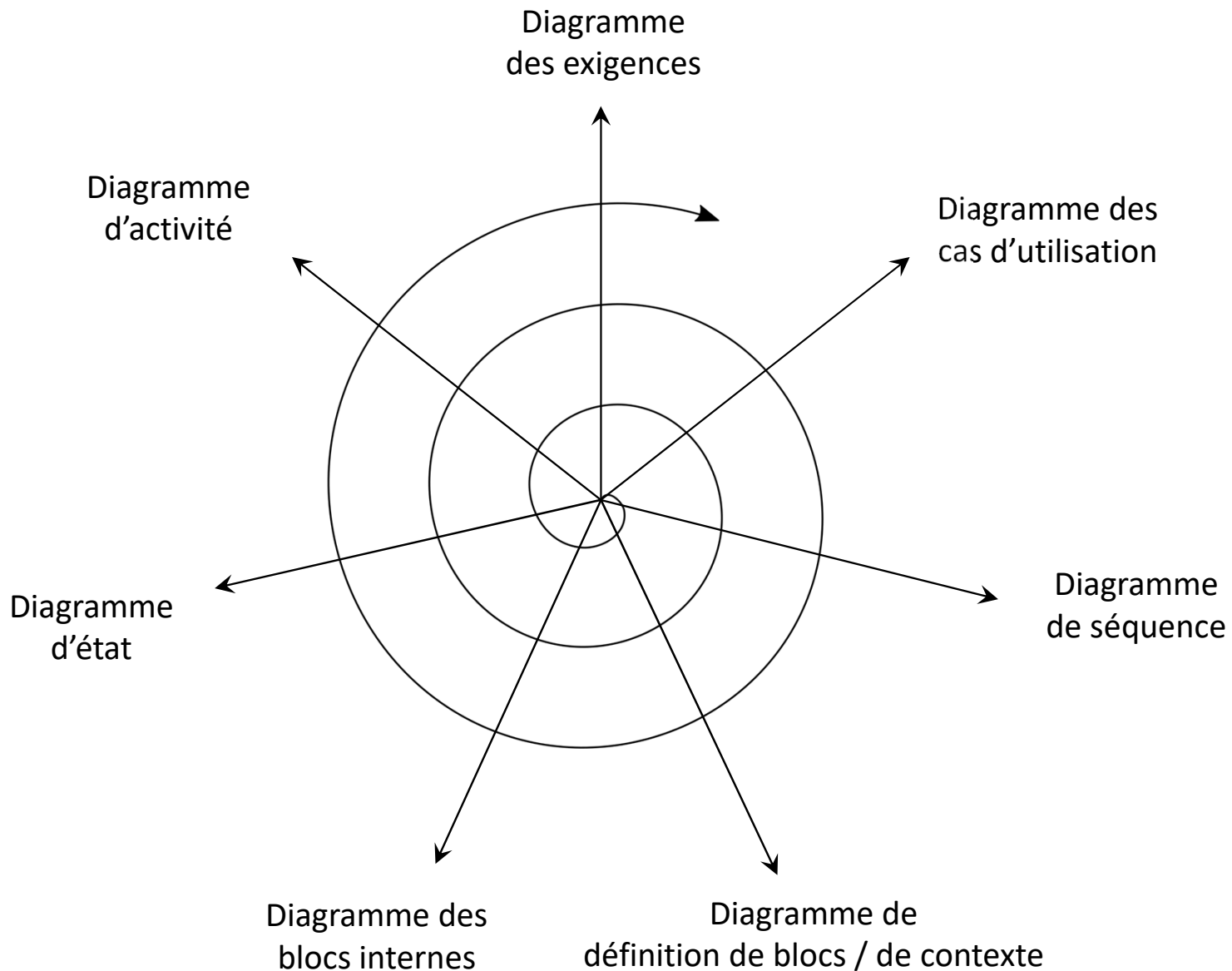
Blocs internes

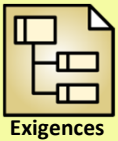


États



Activité





Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



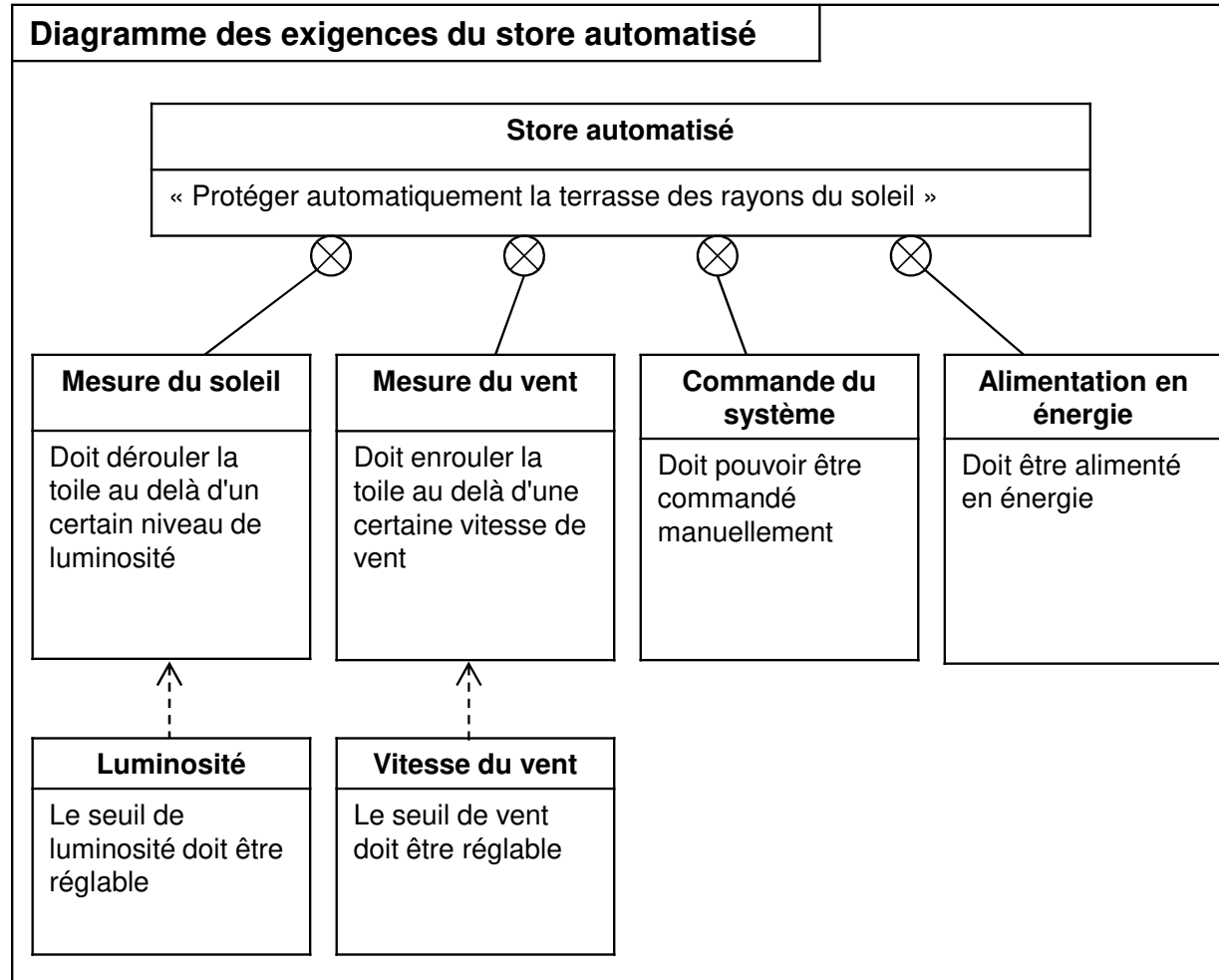
Activité



Diagramme des exigences (*Requirement Diagram*)



C'est le **cahier des charges fonctionnel** du système
(Le système doit...). Ce sont les **exigences** du CdCF.



1- Dans le premier rectangle, on indique le **nom** et l'exigence du système (**mission principale**).

2- Ensuite, on décompose la mission principale en **exigences unitaires**. On précise au dessus l'élément concerné.

3- Pour chaque exigences unitaires, on peut ajouter des précisions, des valeurs, des données...

→ Ce diagramme présente les fonctions ou les contraintes que l'objet doit satisfaire. **Nous allons réutiliser ces éléments pour le diagramme suivant, le diagramme des cas d'utilisation.**



Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



Activité

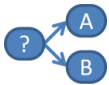
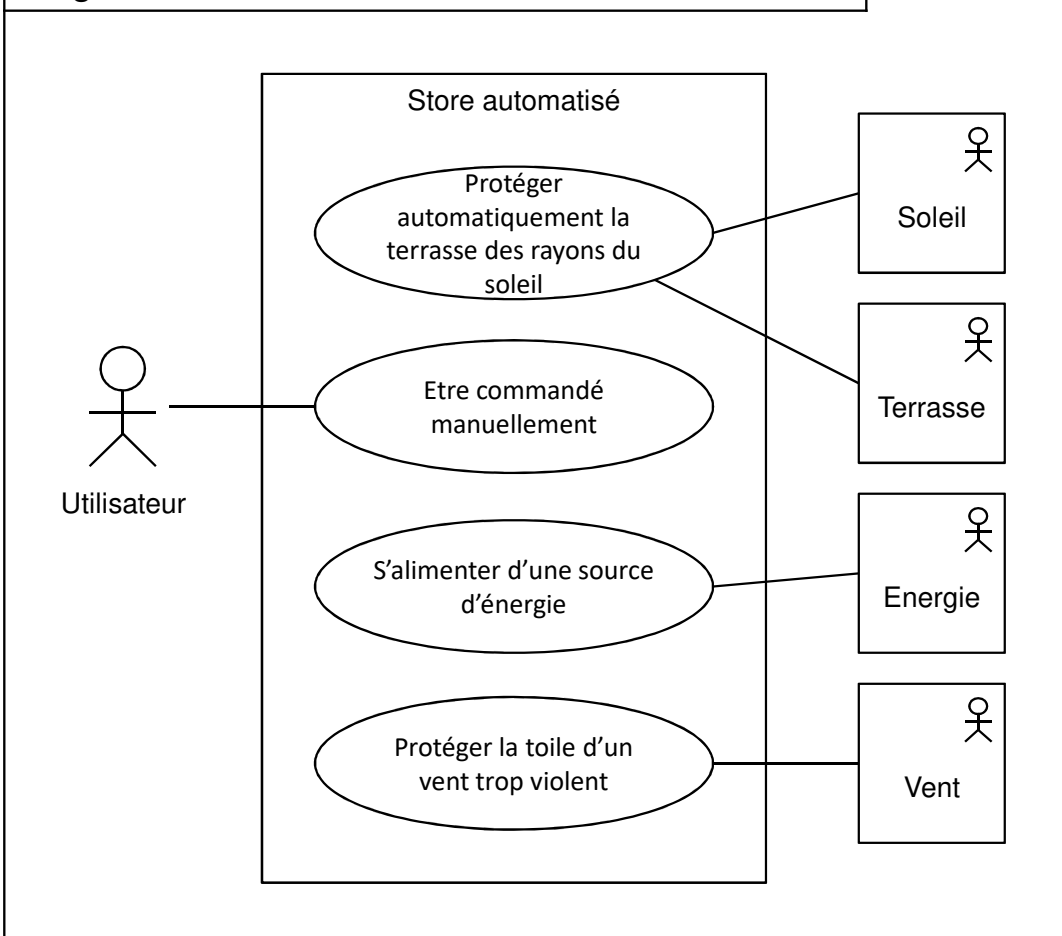


Diagramme des cas d'utilisation (Use Case Diagram)



Il exprime les **services offerts par l'objet aux acteurs**.
→ Décrit ce que fait l'objet (et non ce que fait l'utilisateur) mais sans dire comment il le fait.

Diagramme des cas d'utilisation du store automatisé



1- Un grand rectangle indique la **frontière de l'objet**

2- À gauche, on place des **acteurs humains**.

À droite, les **acteurs non humains**.

(→ Ces éléments sont les mêmes que ceux du diagramme de contexte).

3- Dans des ovales, on décrit les actions réalisables par le système (les services rendus par le système aux acteurs, sous forme de verbe à l'infinitif plus compléments)

Et on va les relier aux acteurs concernés.

i Il arrive que des actions (ovales) ou des acteurs soient reliés entre eux si il agissent l'un sur l'autre.

On y retrouve généralement la mission principale et les exigences déjà identifiées dans le diagramme de contexte (mais on précise comment sont assurées les missions)



Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



Activité

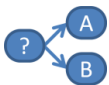
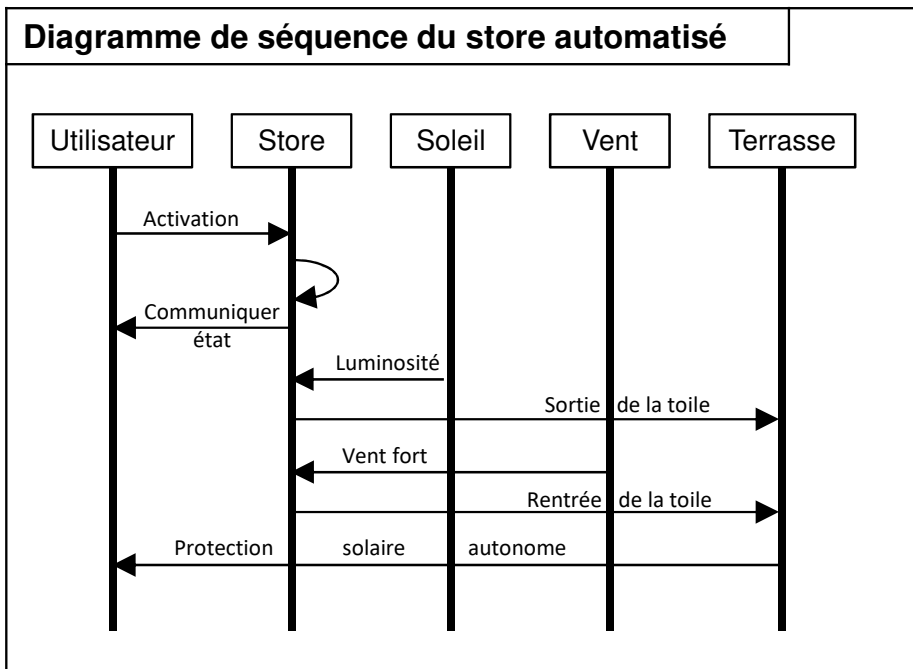


Diagramme de séquence (*Sequence Diagram*)



Il présente les **interactions entre les acteurs et l'objet** selon un **ordre chronologique**.



1- En haut, dans des cadres, on place **tous les acteurs d'un cas d'utilisation***. On commence par l'objet étudié. On replace à gauche de l'objet étudié (ici, le store) l'acteur principal et à droite les acteurs secondaires.

2- On trace des flèches qui vont retracer, dans l'ordre chronologique, chaque échange entre les acteurs et l'objet. On lit ce diagramme de haut en bas.

* : l'acteur énergie n'est pas représenté. En effet, il y a en principe, un diagramme de séquence par cas d'utilisation mais on essaye d'en faire un seul qui contient toutes les interactions possibles. Ici, l'auteur a choisi de ne pas représenter l'alimentation en énergie.

→ Ce diagramme permet de représenter, étape par étape, les échanges (information, énergie, matière) entre chaque acteurs et l'objet lui-même dans un ordre chronologique, représenté par les lignes de vie verticales.



Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



Activité

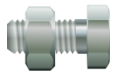


Diagramme de définition de blocs (*Block Definition Diagram*)



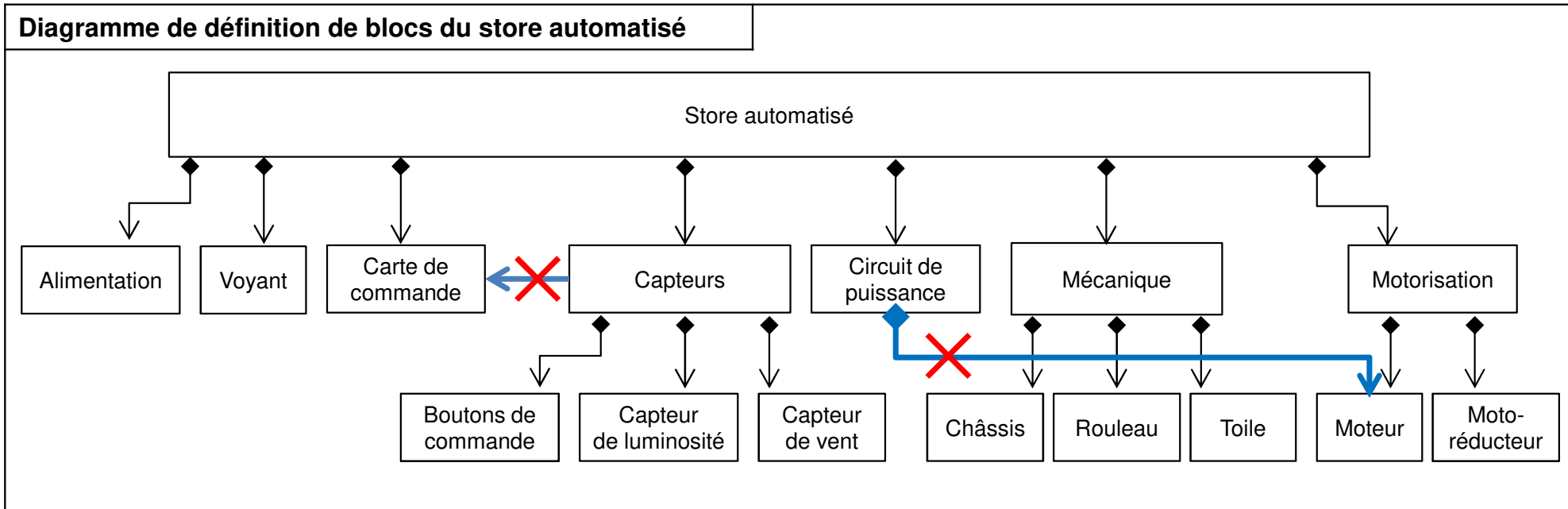
Il permet simplement de **lister tous les blocs** et les sous-blocs **qui constituent l'objet**.

1- Au centre, en haut, on place l'objet

2- En dessous, on place chaque bloc et éventuellement des « sous-blocs » que l'on trouve dans l'objet

i Dans ce diagramme, il n'est pas nécessaire d'expliquer comment les blocs interagissent entre eux. Par exemple, pas besoin d'indiquer que les capteurs sont reliés à la carte de commande.

→ Ce diagramme permet de faire l'inventaire de tous les blocs qui composent l'objet.





Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



Activité

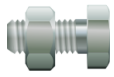
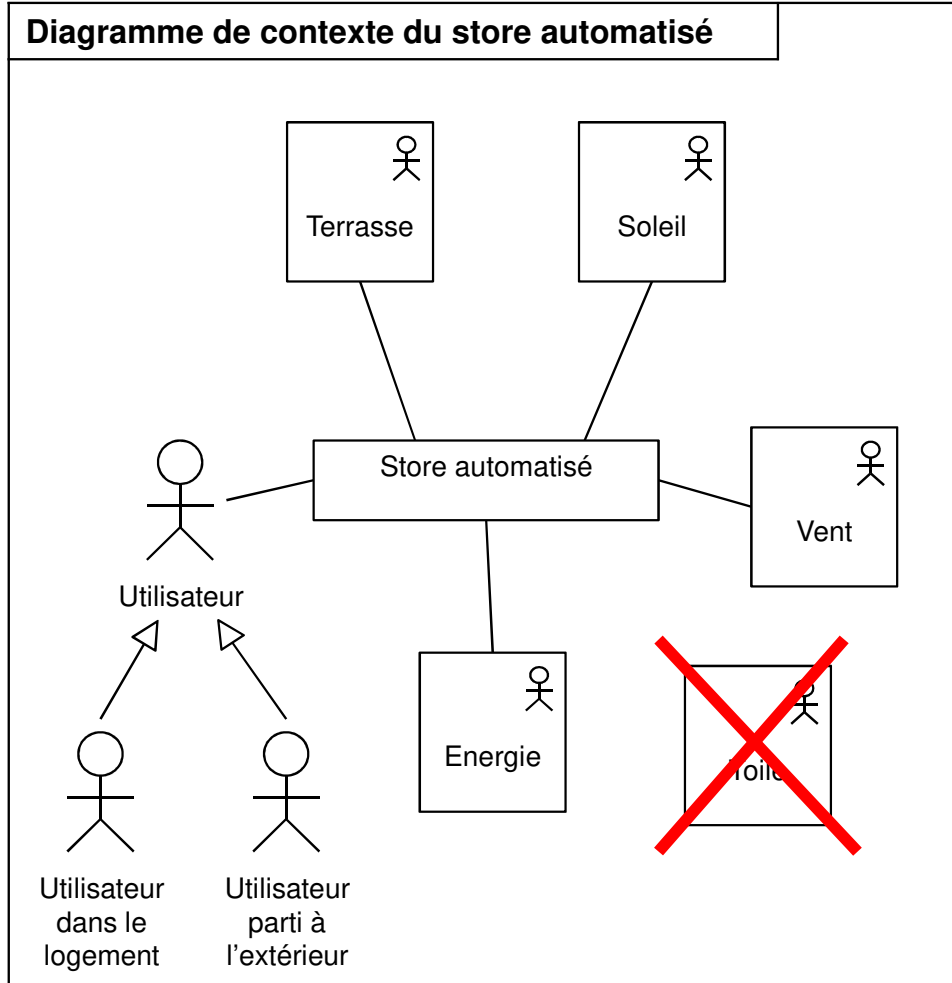


Diagramme de contexte (Context Diagram)



Il recense les éléments extérieurs (**acteurs**) qui interagissent avec l'objet.



1- Au centre, on place l'objet

2- On place tout autour, les acteurs qui interagissent avec l'objet (ce sont les mêmes que ceux du diagramme des cas d'utilisation)

Si besoin, un acteur peut être décomposé en deux sous acteurs .

i Il ne faut pas placer comme acteur des éléments qui appartiennent au système.

Exemple ici : toile, capteurs... ne sont pas à indiquer.

→ Ce diagramme permet de lister tous les éléments extérieurs en relation avec l'objet.



Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



Activité

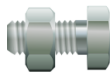
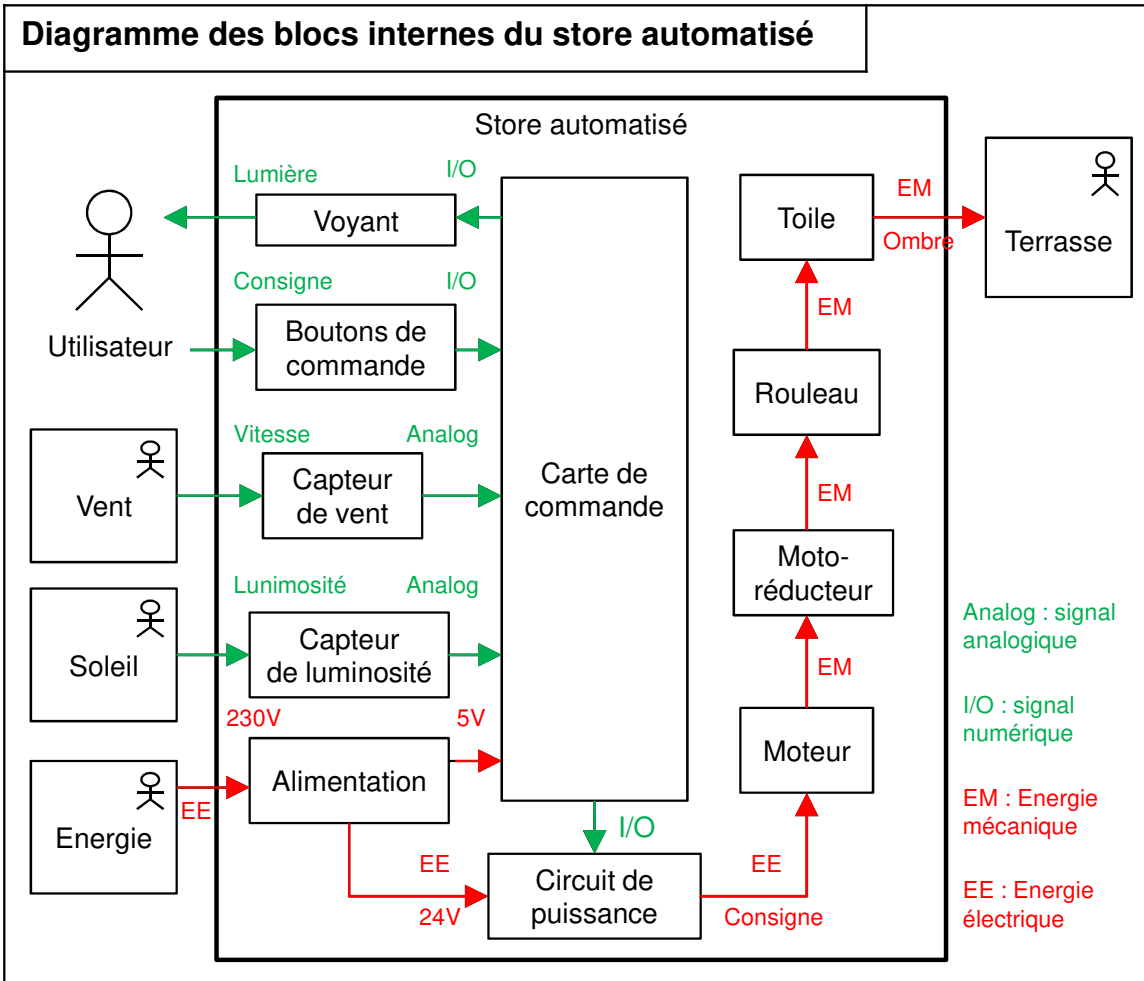


Diagramme des blocs internes (Internal Block Diagram)



Il permet de définir comment l'information, l'énergie et la matière circulent à travers l'objet.



- 1- Dans un grand rectangle, on délimite l'objet.
- 2- A l'extérieur de ce rectangle, on place les acteurs (définis dans le diagramme de contexte)
- 3- A l'intérieur du rectangle, on place les blocs de l'objet (définis dans le diagramme de définition des blocs)

i Ici, le Châssis n'est pas représenté car il n'a pas d'échange de flux dans l'objet

4- On représente par des flèches les flux d'information, d'énergie et de matière.

i On peut utiliser des codes couleurs et ajouter des informations sur les flux

i Diagr. séquence / ports SysML

→ Dans ce diagramme, on voit les échanges et interactions entre les différents blocs.

Il remplace la chaîne d'énergie et d'information.



Exigences



Cas d'utilisation



Séquence



Définition de blocs



Contexte



Blocs internes



États



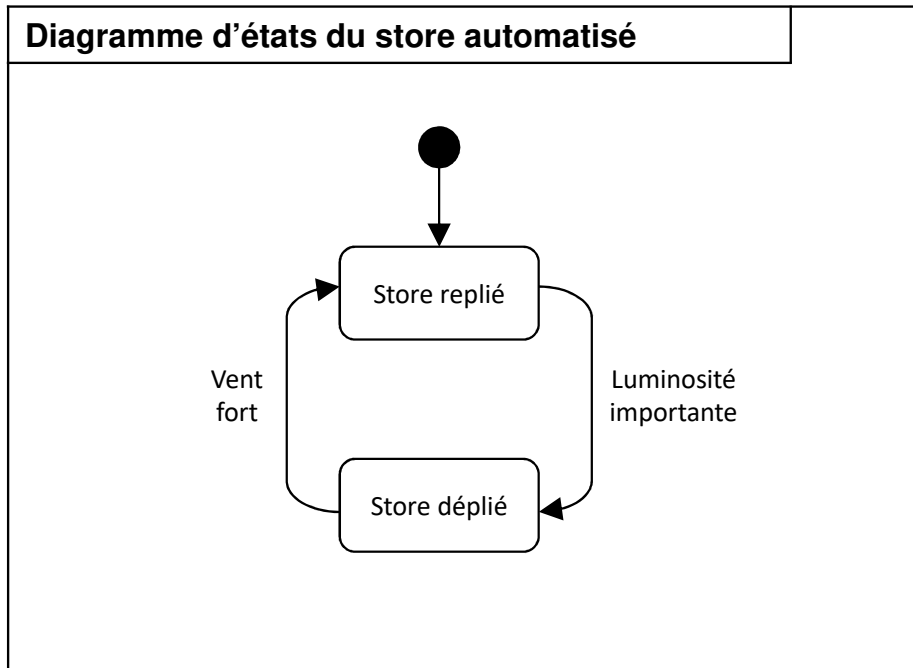
Activité



Diagramme d'états (*States Machine Diagram*)



Il présente les **différents états possible de l'objet.**



- 1- Le point noir représente l'**état initial** de l'objet.
- 2- On indique en dessous l'état de départ
- 3- On indique les autres **états observables** de l'objet
- 4- On précise par des **flèches** les **événements** qui conduisent aux changements d'états.

→ Si ces diagrammes ne permettent pas de comprendre globalement le fonctionnement du système, ils sont directement transposables en algorithme.

i Ici, ce diagramme ne comporte pas d'état final. Si il y en a un, il est représenté par le symbole :

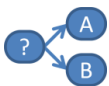
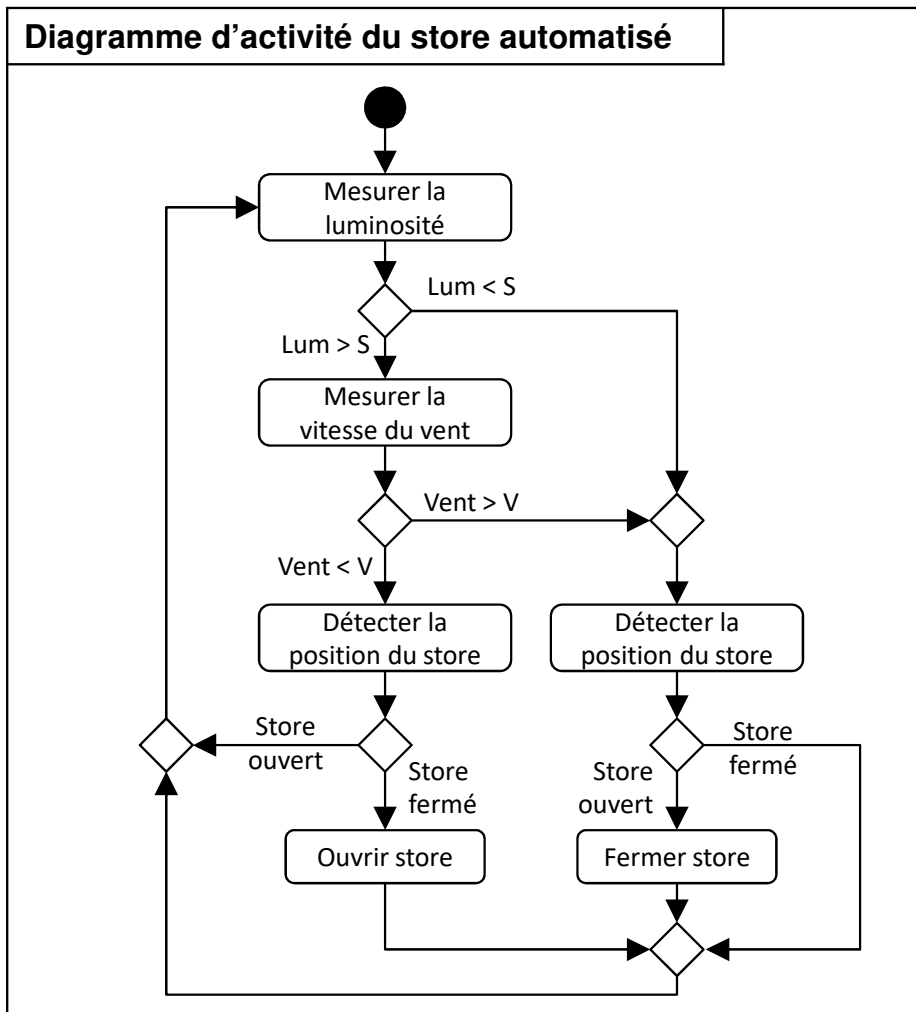


Diagramme d'activité (*Activity Diagram*)



Il présente le **comportement de l'objet** (les choix des **actions** en fonction de **décisions**)



- 1- Le point noir représente l'**état initial** de l'objet.
 - 2- On indique les **actions dans des rectangles**
 - 3- On indique dans des losanges les **nœuds** (là où les flèches se séparent ou se retrouvent)
 - 4- On relie les éléments avec des flèches selon le comportement de l'objet
- Ce diagramme remplace les schémas des algorithmes.

i Ici, ce diagramme ne comporte pas d'état final. Si il y en a un, il est représenté par le symbole : ●

Sources :

- Description exemple SysML Aspirateur autonome (Guillaume MARTIN, ac-grenoble)
- Boite à outils SysML : les diagrammes, Eduscol (Didier FAGNON, Stéphane GASTON)
- Des outils de l'analyse fonctionnelle au « SysML », ATDTec (Abdelkrim EL MARRAKCHI)
- Ingénierie Système en SysML appliquée à la rédaction du cahier des charges (Y. LE GALLOU, ac-lille)
- Représentations symboliques : Le langage SysML, Pedalite (Olivier GUILLOTIN)
- UML 2 De l'apprentissage à la pratique (laurent AUDIBERT)

Crédits :

- Toutes les images sont réalisées par l'auteur ou libre de droit (site d'image libres de droits Pixabay)
- Les logos des diagrammes SysML ont été inspirés du diaporama d'Abdelkrim EL MARRAKCHI et redessinés par l'auteur avec le logiciel Inkscape.

Réalisation :

👉 Igor FERRIEU – 2018 sous licence CC-BY-SA (réutilisation libre à condition de citer l'auteur et de distribuer sous cette même licence libre)