

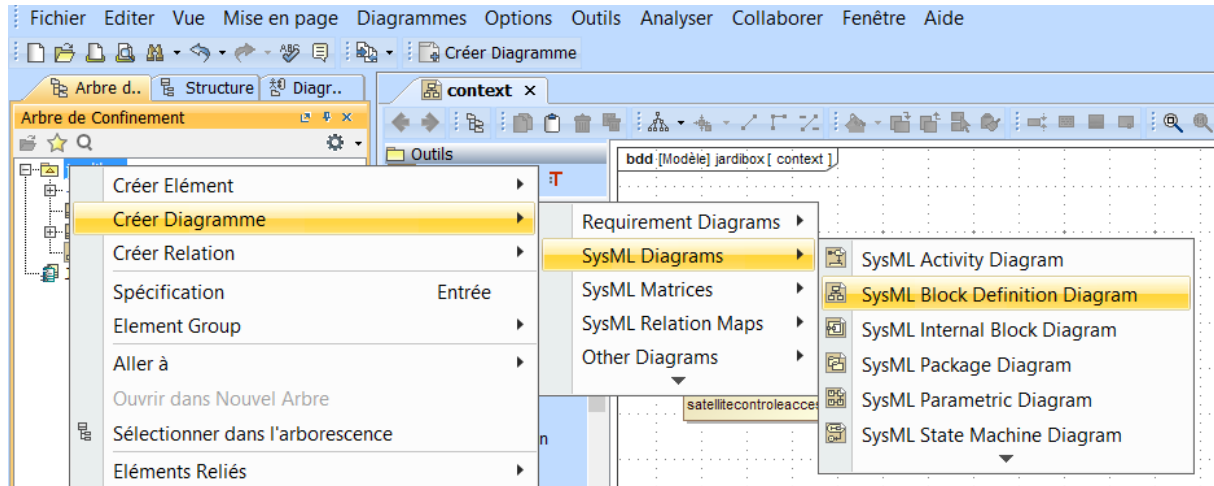
UTILISATION DE MAGIDRAW : CRÉATION DU SYSML D'UN SYSTÈME BY S.B

Objectifs : Être capable de modéliser un objet technique.

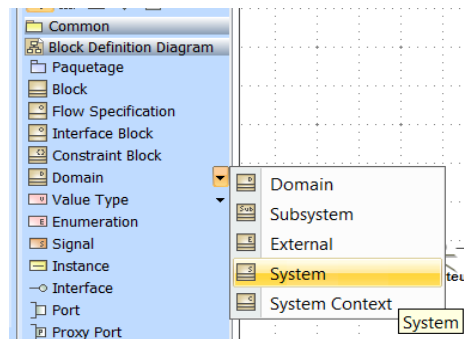
1. Création du diagramme de contexte du système.....	2
2. Création des BDD (diagrammes de définition blocs).....	5
3. Création des ibd à partir du bdd.....	8
3.1. Liaisons entre blocs.....	9
3.2. Propriétés : contraintes, valeur,.....	10
4. Création des idb (diagramme de bloc interne).....	11
4.1. Construction de l'IBD.....	11
4.2. Liaisons entre blocs.....	11
4.3. Propriétés : contraintes, valeur,.....	12
5. Création d'un diagramme de séquence (SR).....	14
5.1. Créer le diagramme.....	14
5.2. Créer des lignes de vie.....	14
5.3. Créer les liens entre les lignes de vie.....	15
6. Création du diagramme d'exigence.....	16
7. Diagramme de cas d'utilisation "uc : user case".....	18

1. CRÉATION DU DIAGRAMME DE CONTEXTE DU SYSTÈME

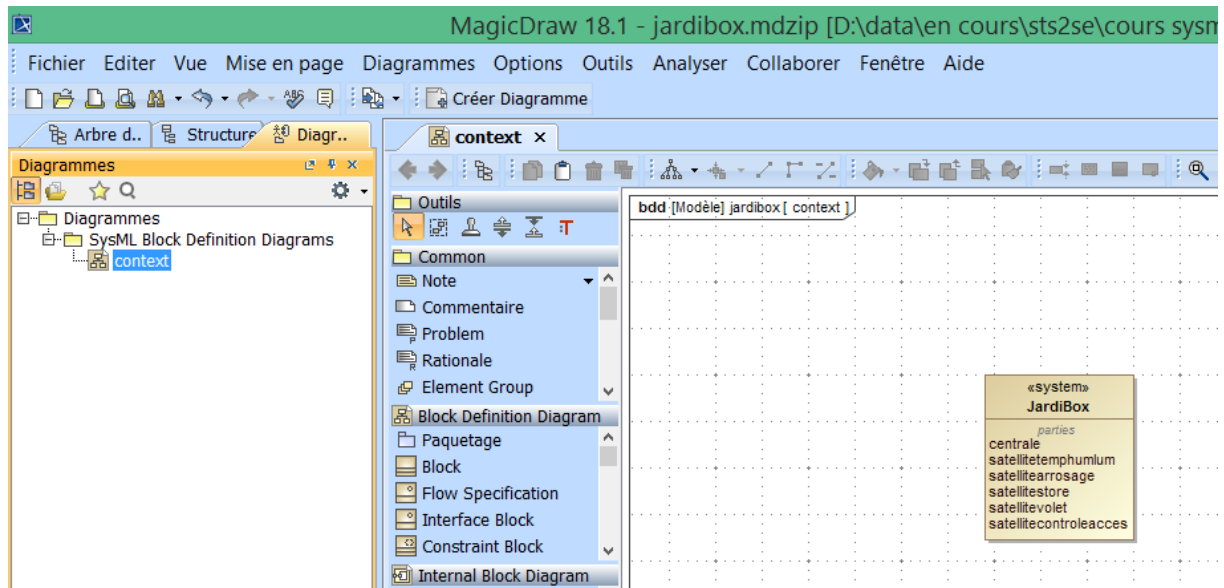
On doit créer un bdd et insérer un "block" du type "system " :



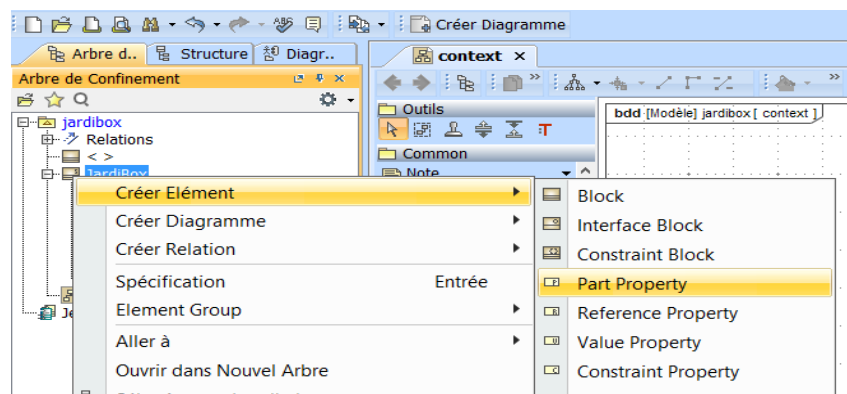
Pour que le block soit "system " il faut utiliser :
domain + system



J'appelle ce diagramme « context » (j'aurai pu directement faire un block "system context " :

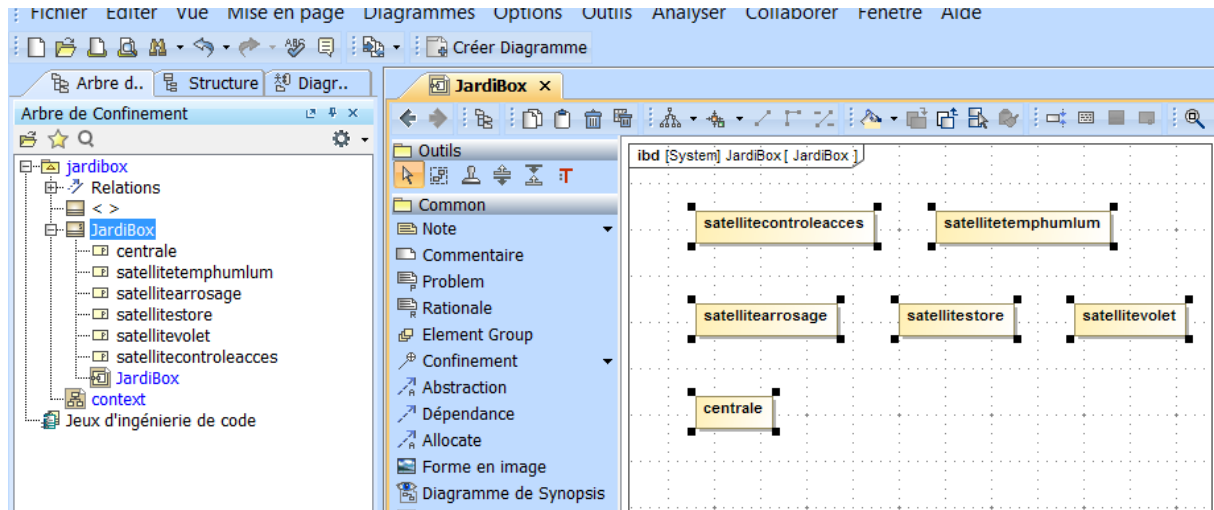


La jardibox se compose de plusieurs parties, j'ajoute donc des "parts" :
se placer sur l'arbre de confinement :



L'intérêt de créer ces parties est que l'ibd se crée tout seul à partir du bdd en incluant les block de chaque partie créée :

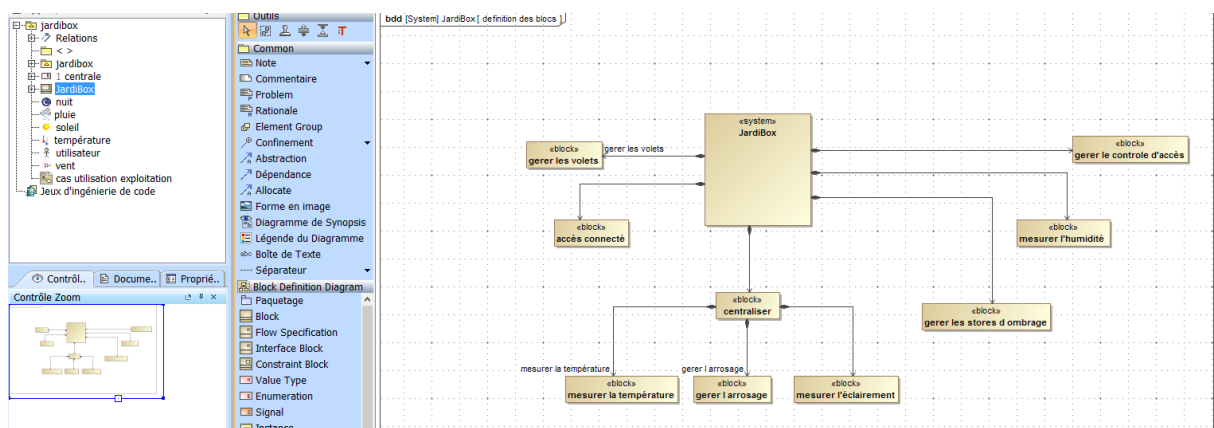
ceci donne l'ibd suivant :



Attention pas du tout judicieux à ce niveau.

Rmq : les éléments en bleu indiquent qu'ils n'ont pas été sauvegardé (solution : ctrl +s)

Il vaut mieux créer un bdd de définition de blocs (FP) qu'on détaillera après en ibd :

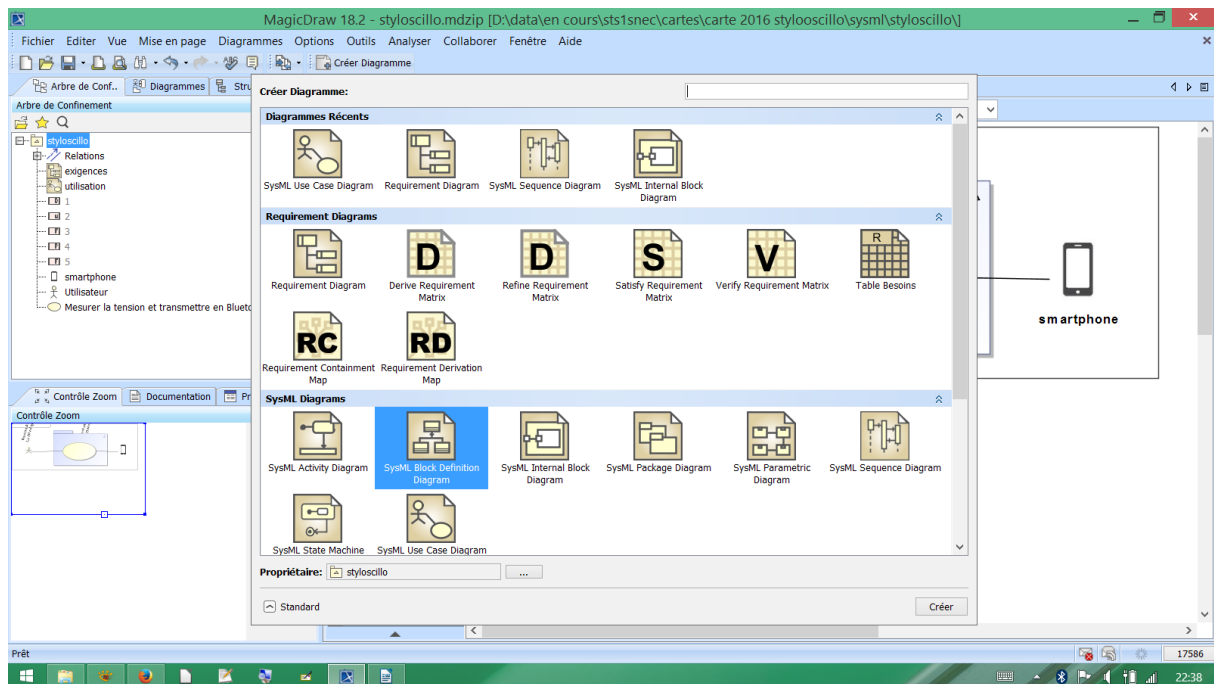


et maintenant on peut pour chaque bloc créer un ibd .

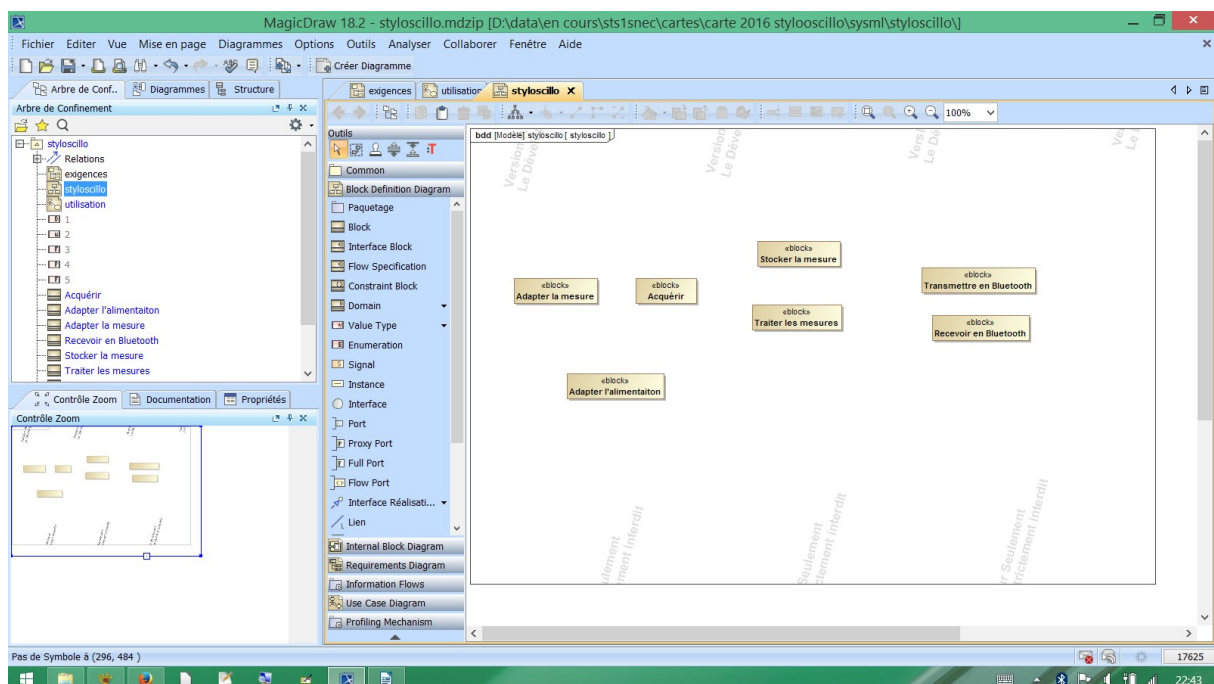
2. CRÉATION DES BDD (DIAGRAMMES DE DÉFINITION BLOCS)

Ces blocs représentent les fonctions principales du systèmes.

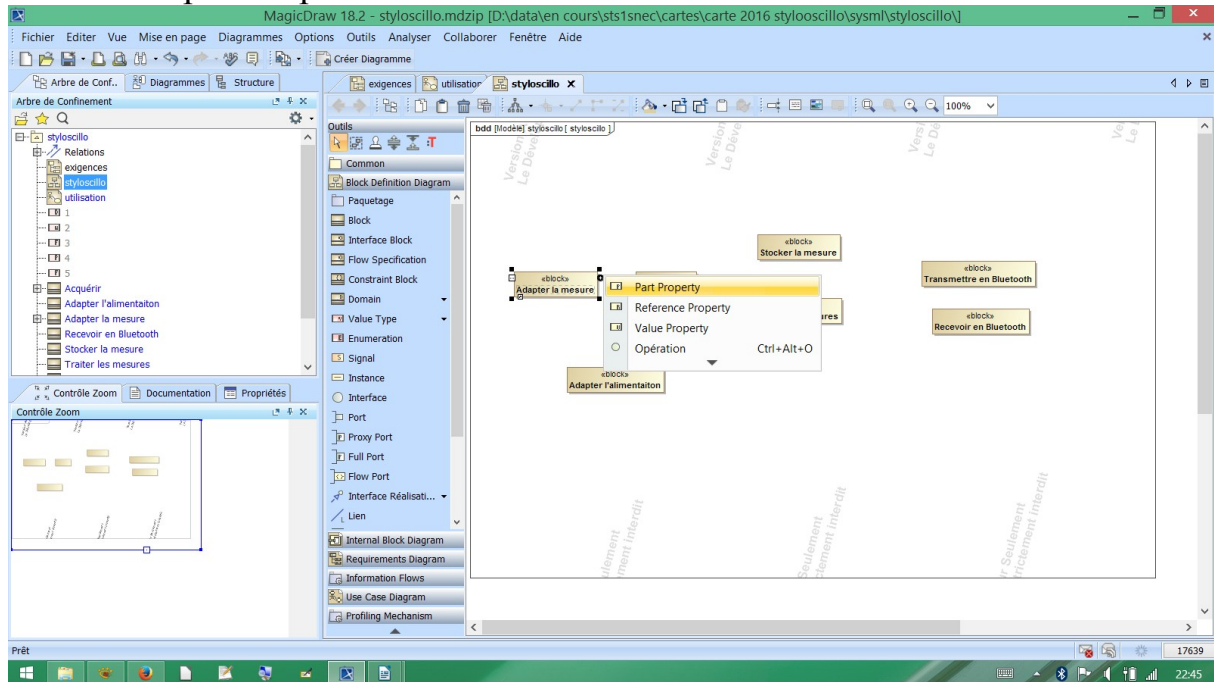
Créer le diagramme :



Créer des blocs dans le diagramme en utilisant les outils spécifiques au bdd :

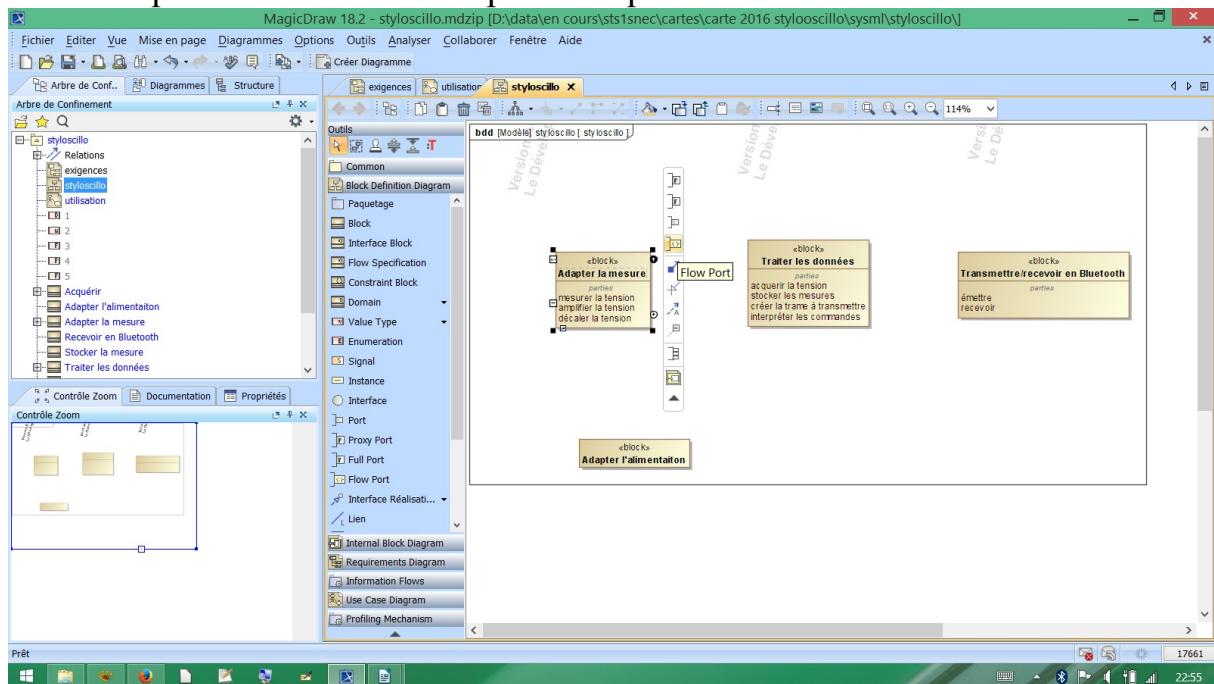


Inclure des parties 'parts' dans les différents blocs :

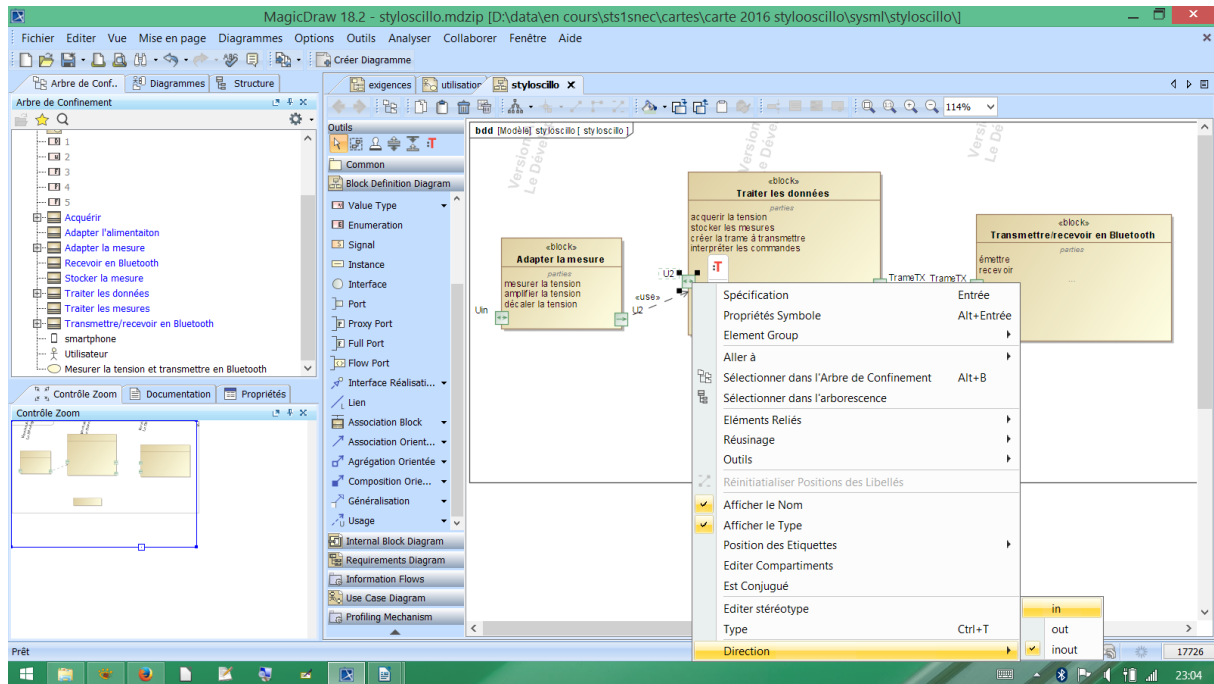


Attention : les parties permettent de créer automatiquement les IBD.

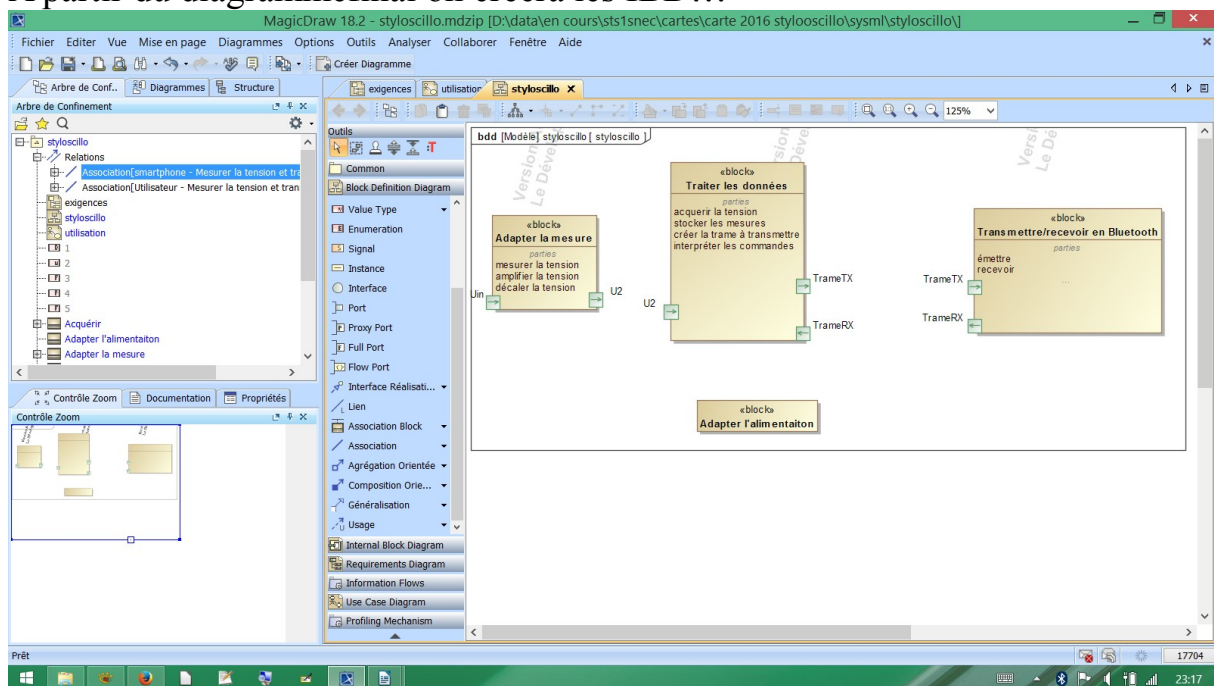
Créer les ports d'entrées/sorties pour chaque blocs :



Les ports peuvent être orientés :

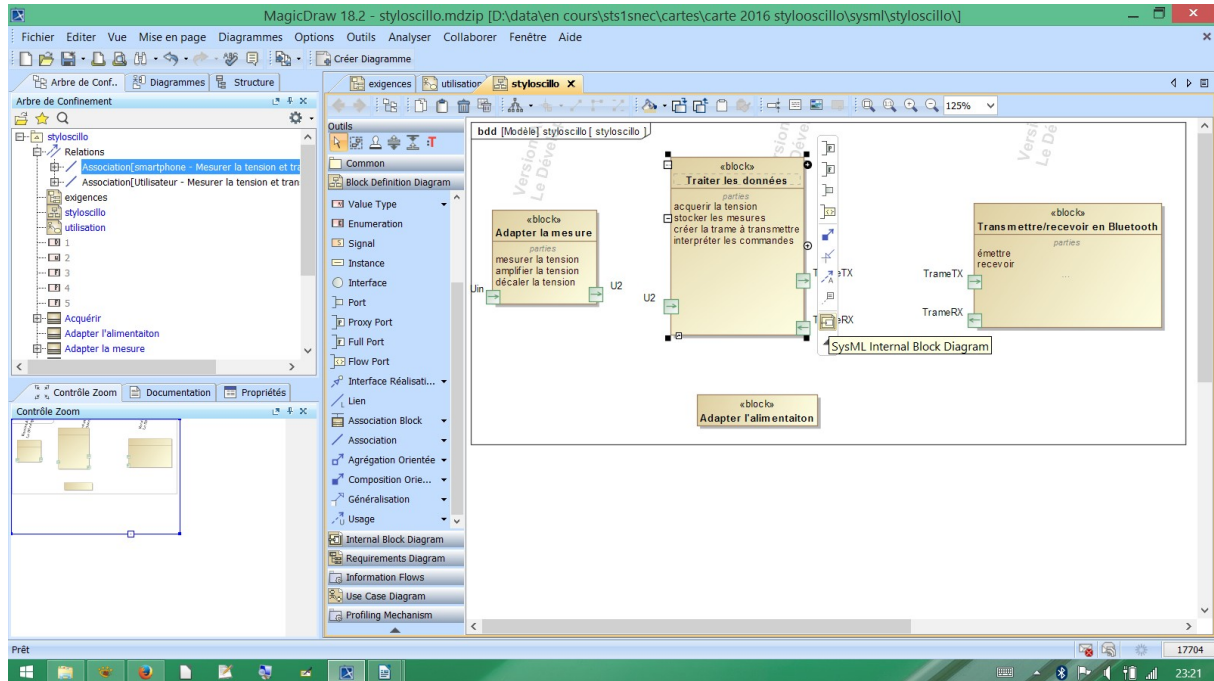


A partir du diagramme final on créera les IBD...



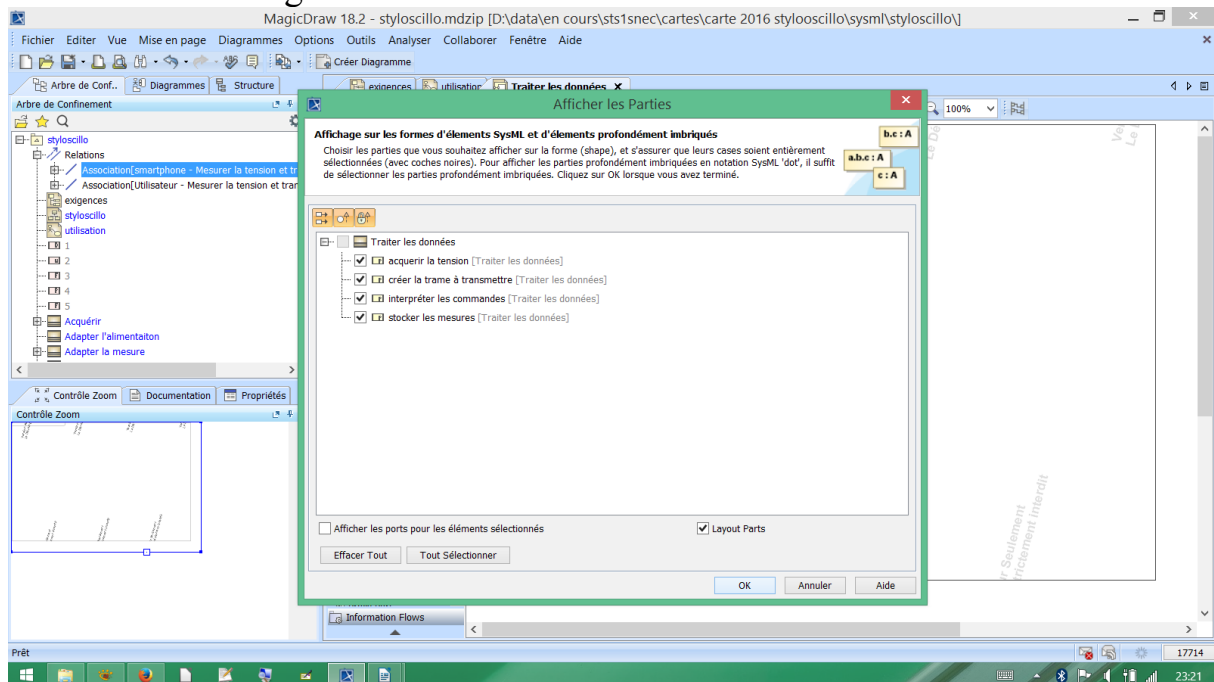
3. CRÉATION DES IBD A PARTIR DU BDD

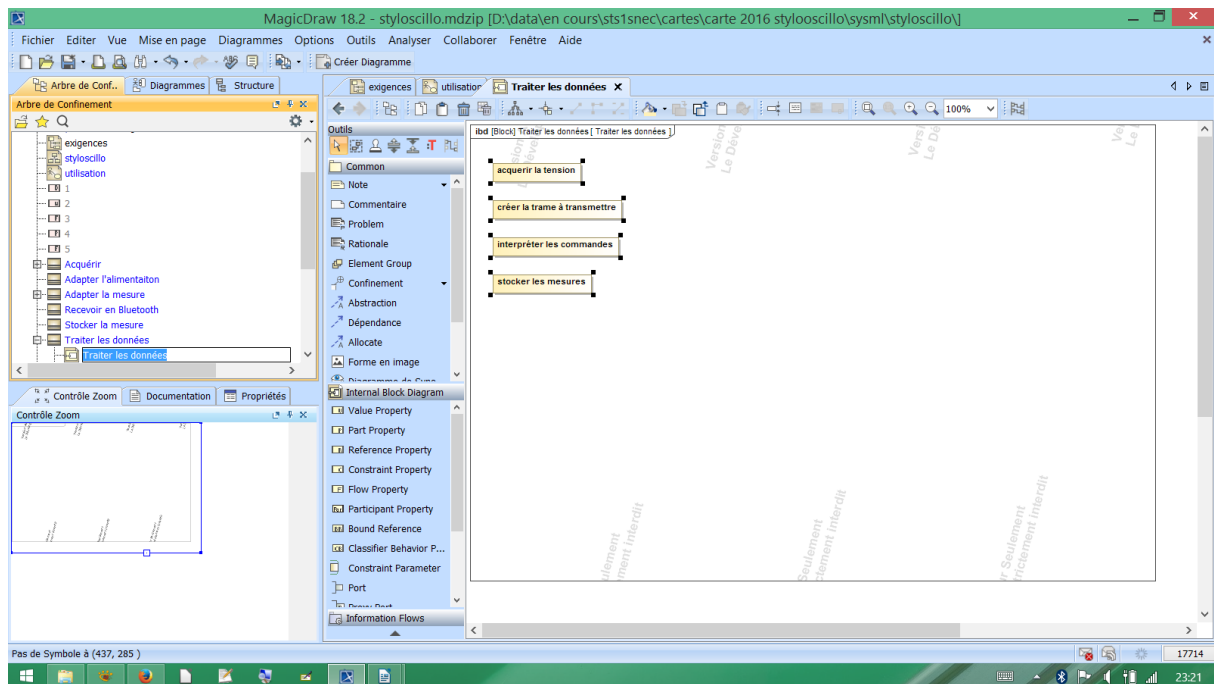
Choisir le bloc :



Cliquer sur l'outil : idb

Une boîte de dialogue s'ouvre :



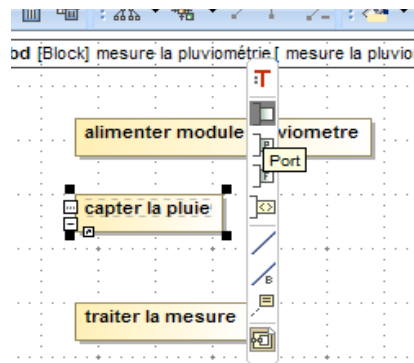


Placer les liaisons entre les blocs internes :

3.1. Liaisons entre blocs

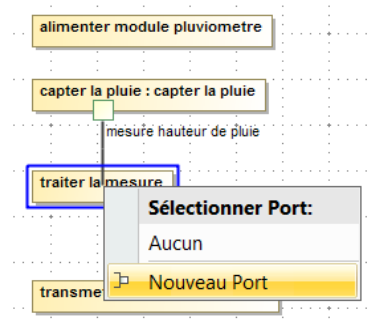
La liaison des blocs se fait en créant des ports.

Créer un port sur le bloc



Nommer le port

Sélectionner le nouveau port + cliquer droit + choisir une liaison + cliquer sur le bloc à relier.

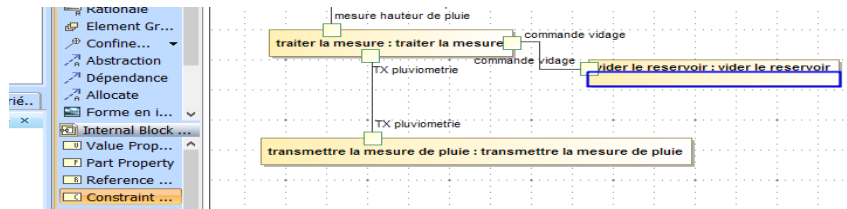


La liaison se fait.

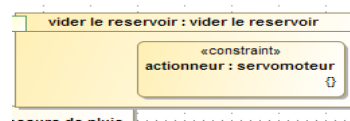
3.2. Propriétés : contraintes, valeur,

Ajouter une contrainte :

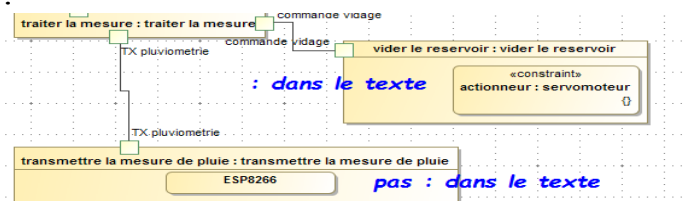
Choisir "constraint property" et glisser vers le bloc :



une contrainte est ajouter :



si dans le texte il y a " : " l'information « constraint » apparaît comme ci-dessus. Sinon on obtient :



Ajouter une partie :

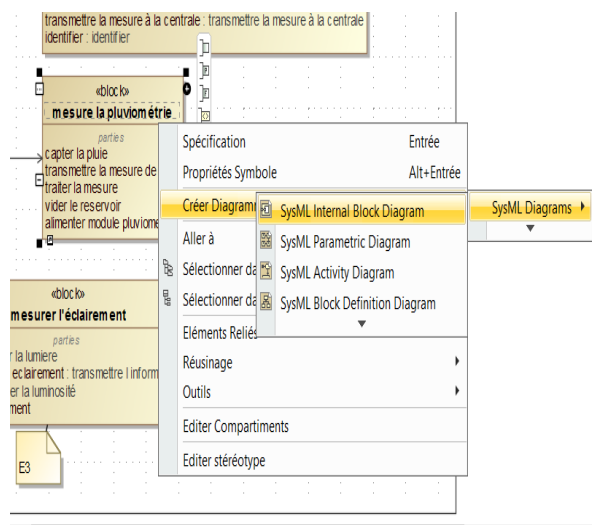
même principe en choisissant dans la barre d'outil de gauche. Le cadre est rectangulaire pour Part .

4. CRÉATION DES IDB (DIAGRAMME DE BLOC INTERNE)

4.1. Construction de l'IBD

L'IBD se fait à partir du BDD.

Sélectionner le BDD + cliquer droit et faire + Créer diagramme SysML + Choisir IBD



L'IBD se construit avec les blocs définis dans le BDD.

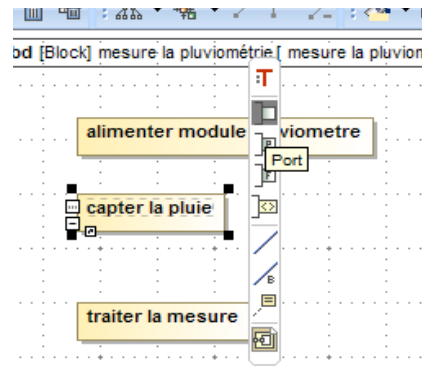
Je n'ai pas trouvé comment actualiser l'IBD si on modifie le BDD une fois l'IBD construit !

Si vous modifiez votre BDD après avoir généré l'IBD, il faut aller chercher dans l'arborescence l'élément bloc "part" nouvellement créé et l'insérer dans l'IBD correspondant.

4.2. Liaisons entre blocs

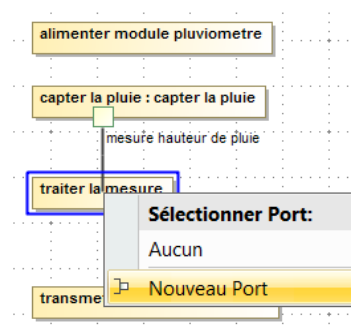
La liaison des blocs se fait en créant des ports.

Créer un port sur le bloc



Nommer le port

Sélectionner le nouveau port + cliquer droit + choisir une liaison + cliquer sur le bloc à relier.

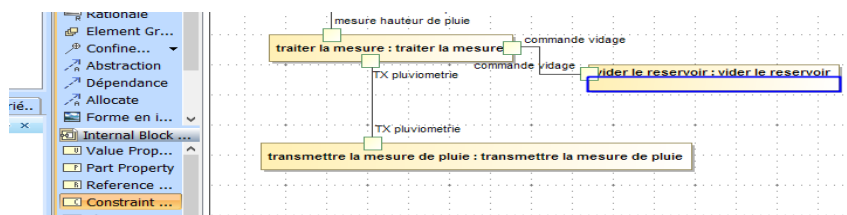


La liaison se fait.

4.3. Propriétés : contraintes, valeur,

Ajouter une contrainte :

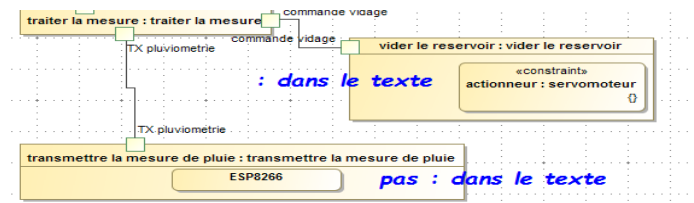
Choisir "constraint property" et glisser vers le bloc :



une contrainte est ajouter :



si dans le texte il y a " : " l'information « constraint » apparaît comme ci-dessus.
Sinon on obtient :



Ajouter une partie :

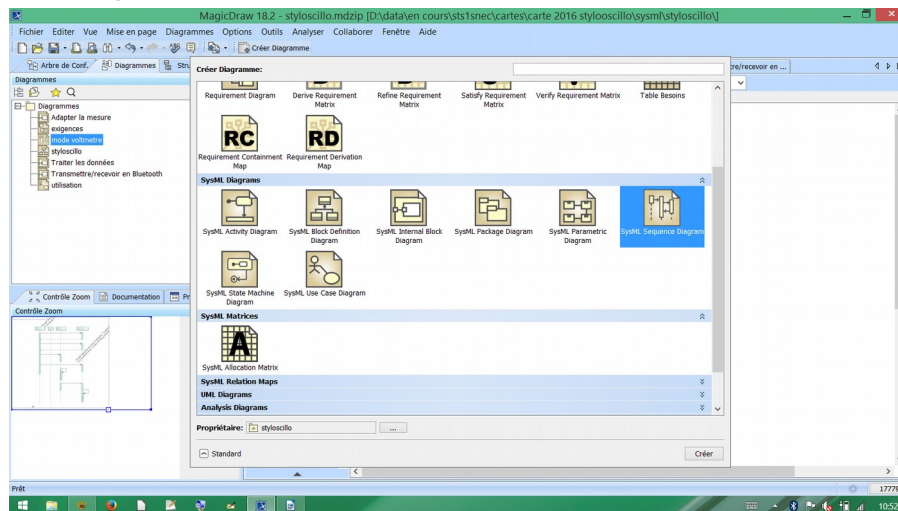
même principe en choisissant dans la barre d'outil de gauche.
Le cadre est rectangulaire pour Part .

5. CRÉATION D'UN DIAGRAMME DE SÉQUENCE (SR)

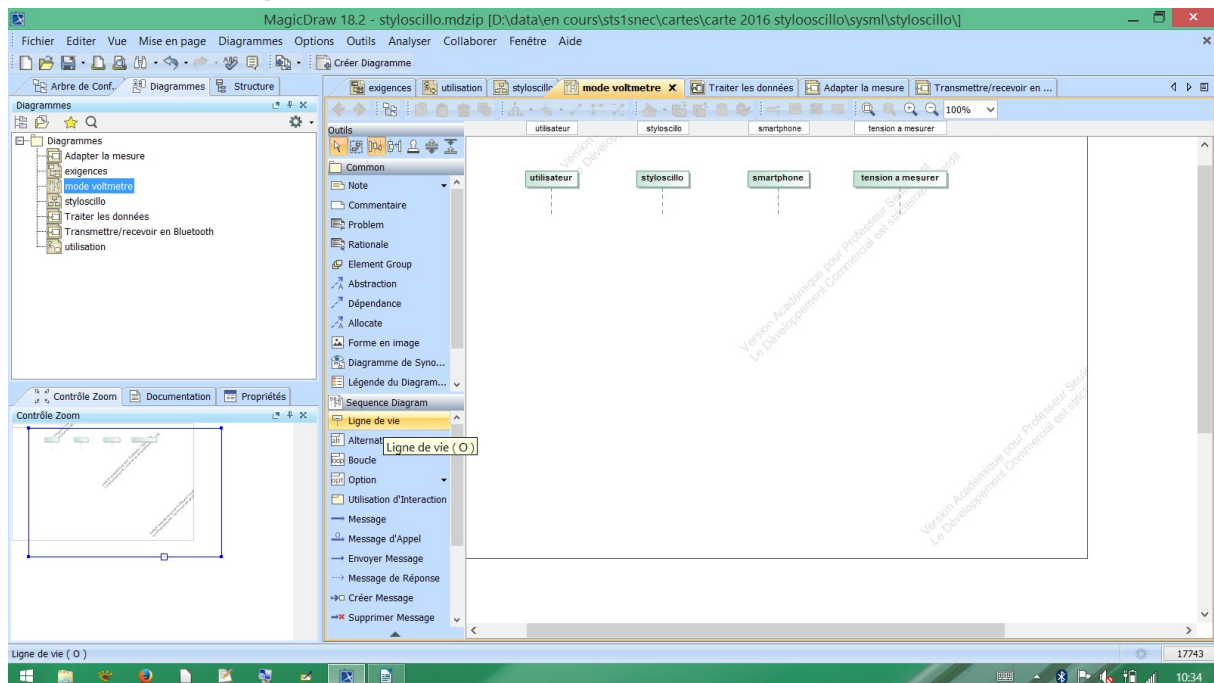
Les diagrammes de séquences fournissent le déroulement du comportement du système.

Ils peuvent servir pour de décrire les "algorithmes" du système.

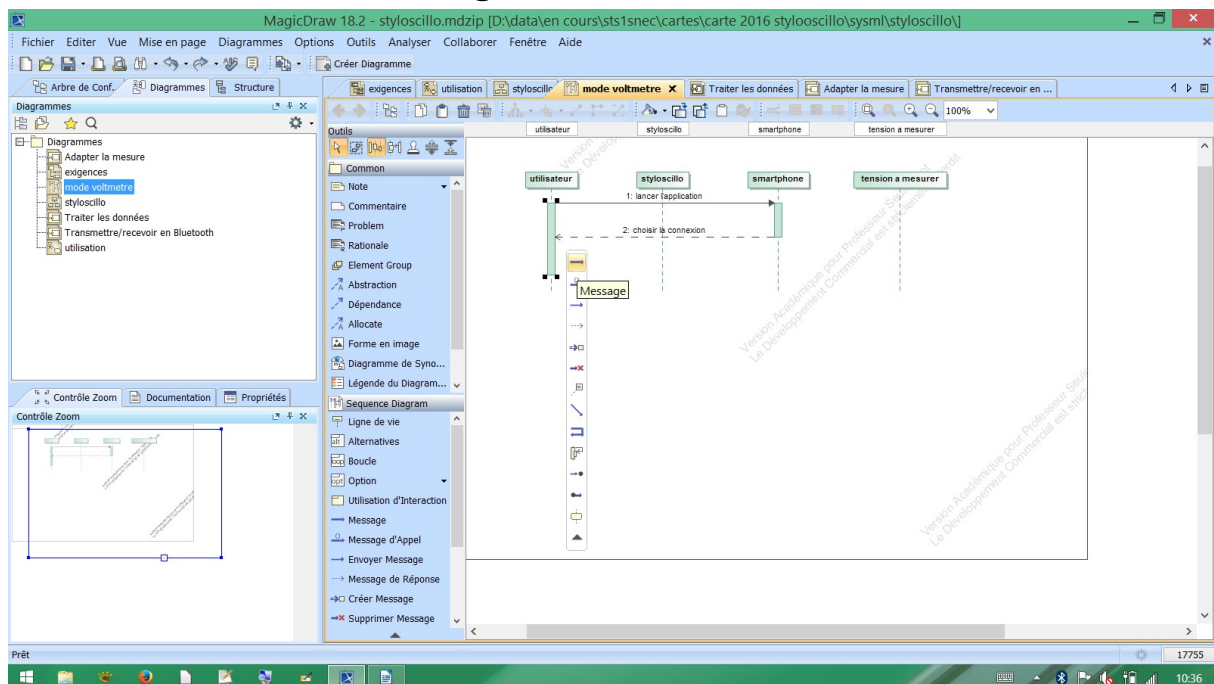
5.1. Créer le diagramme



5.2. Créer des lignes de vie

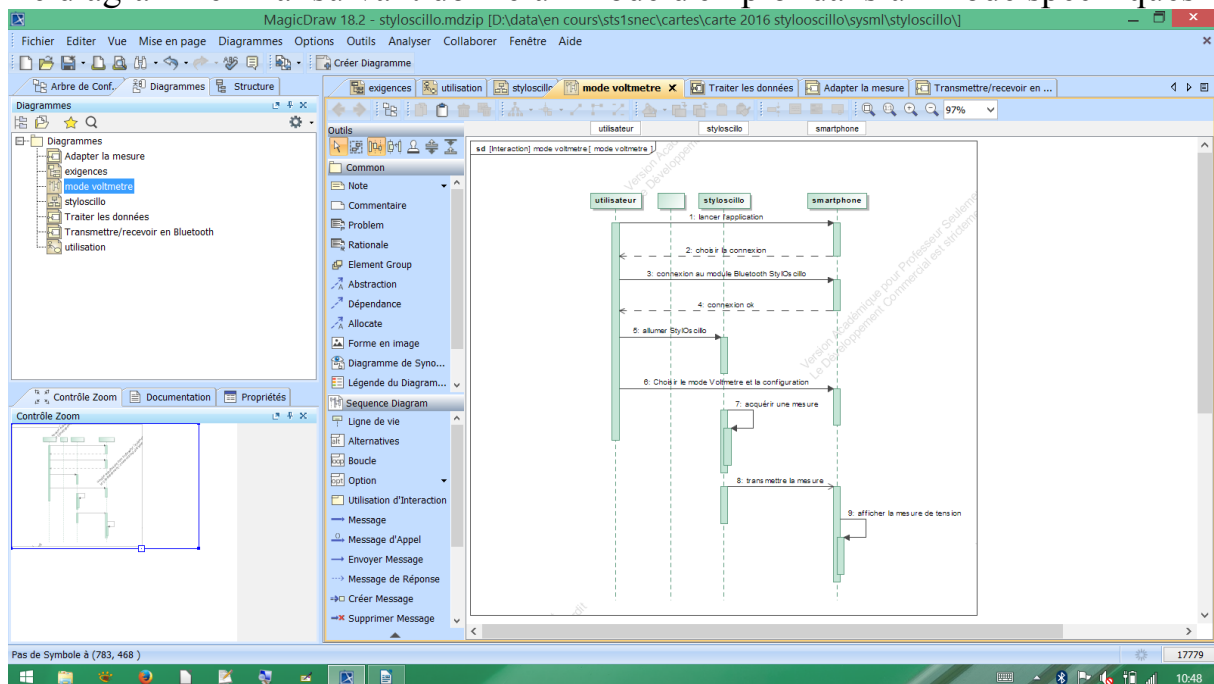


5.3. Créer les liens entre les lignes de vie



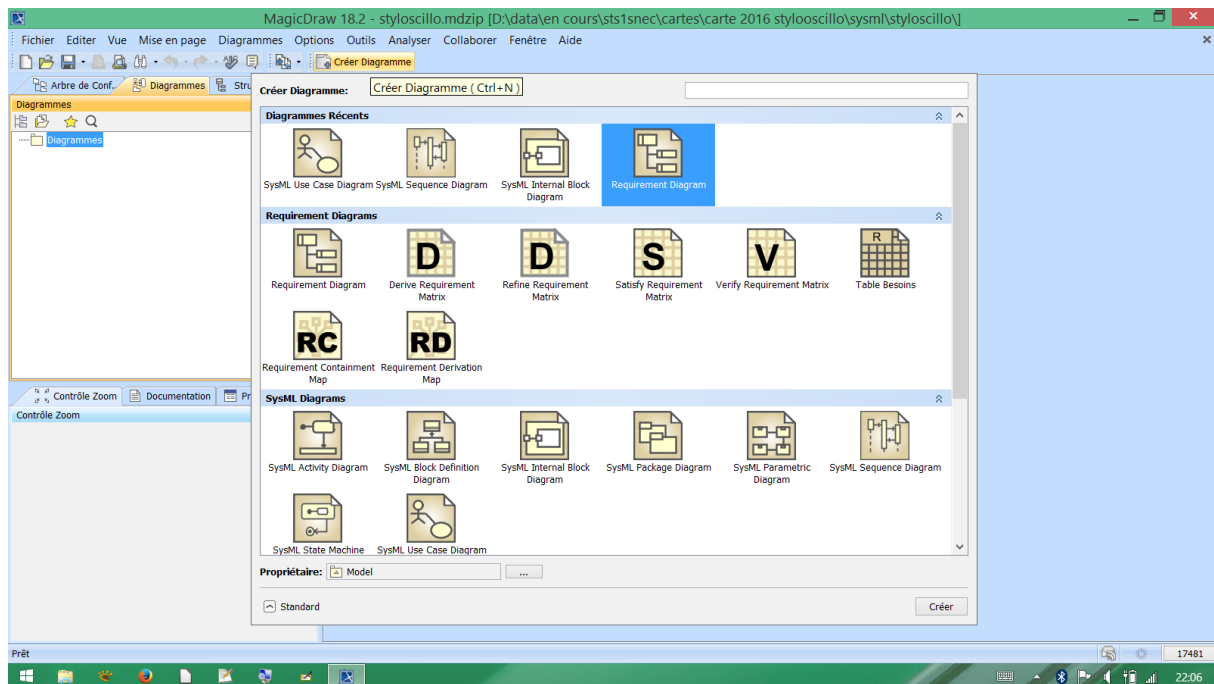
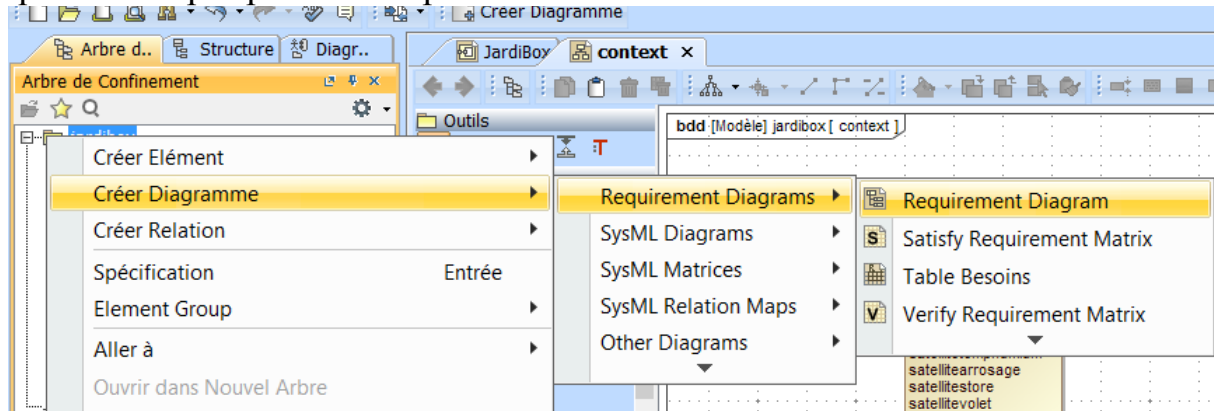
Plusieurs liens sont possible :
message, réponse, répétition...

Le diagramme final suivant donne un mode d'emploi dans un mode spécifiques

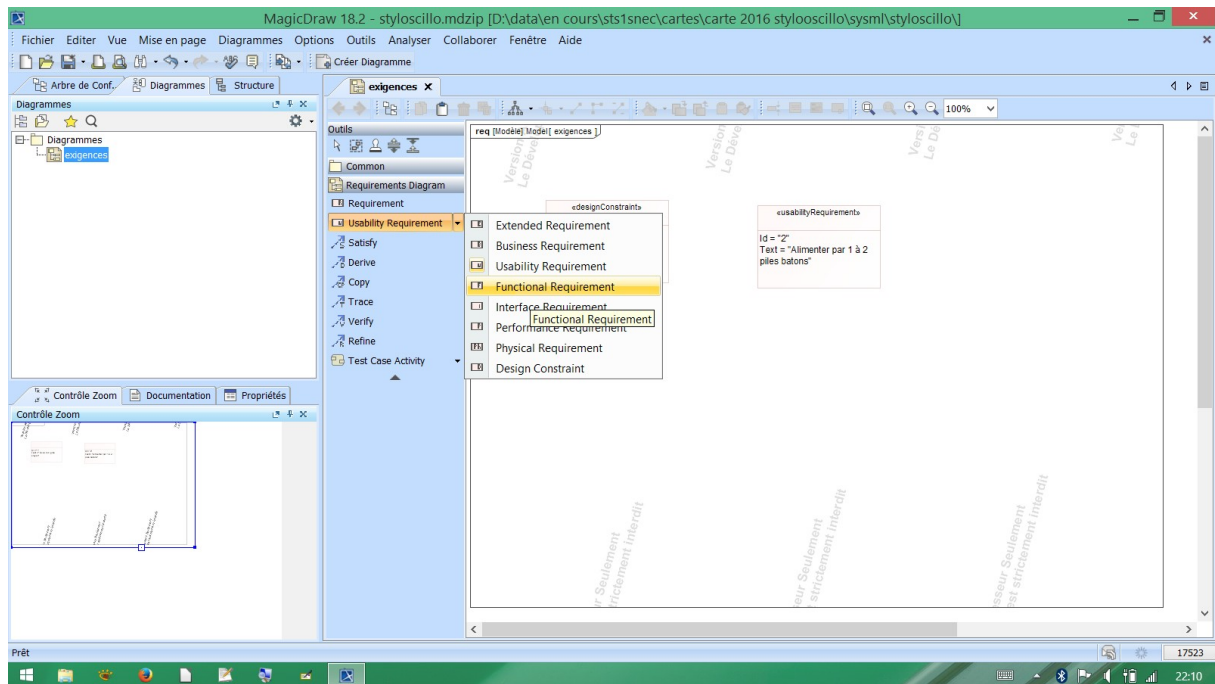


6. CRÉATION DU DIAGRAMME D'EXIGENCE.

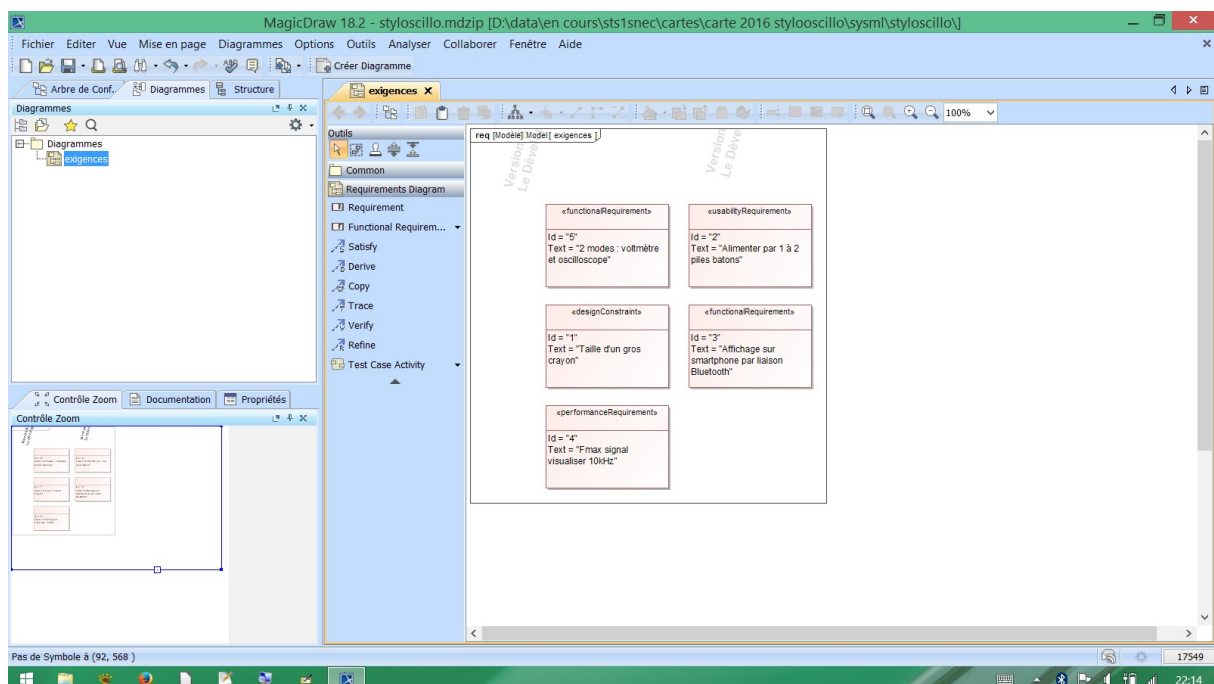
Ce diagramme ne peut être créé qu'à partir de la racine du projet sûrement parce qu'il ne fait pas partie de la partie structure au contraire du "bdd" et des "ibd".



Placer les blocs d'exigences en choisissant le type d'exigences parmi celles fournies dans le menu déroulant :



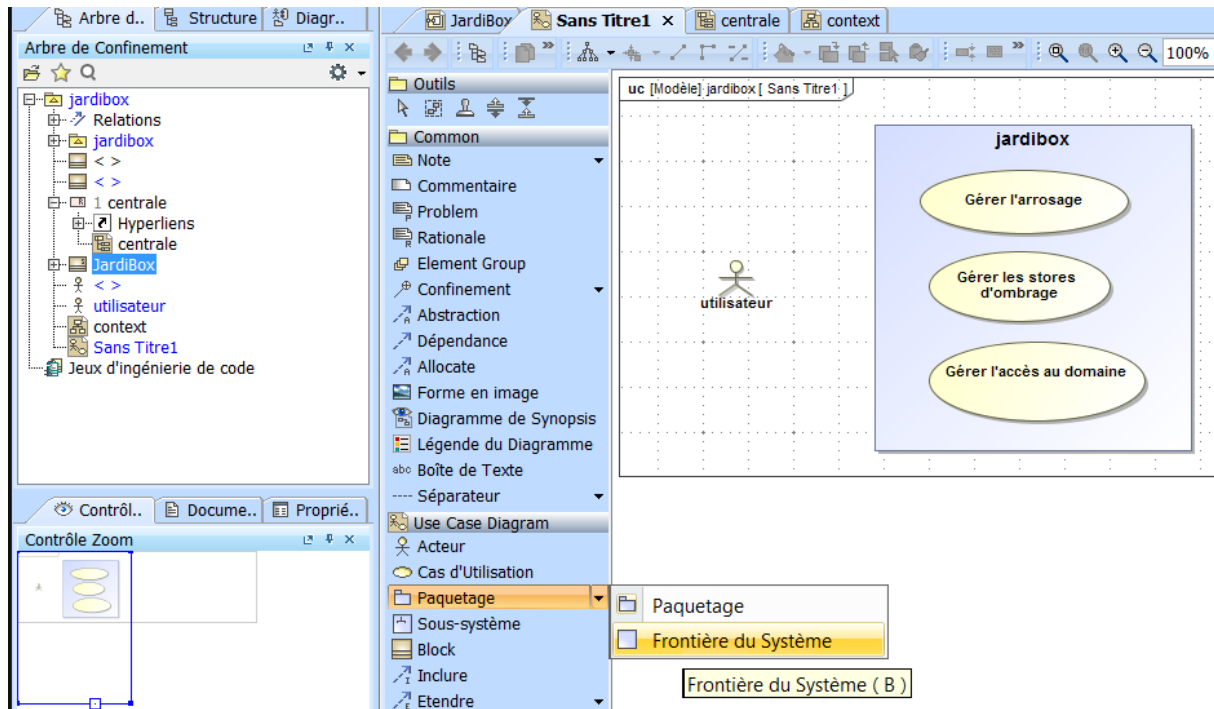
Les exigences sont numérotées et contiennent un texte descriptif.



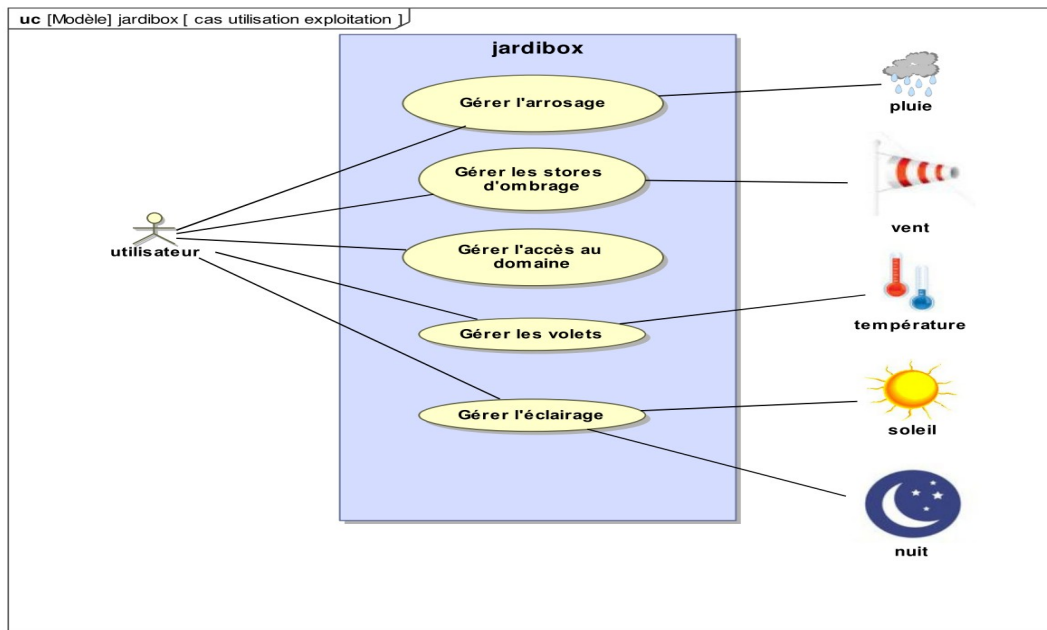
Le diagramme sera complété ultérieurement avec les diagrammes blocs permettant de satisfaire les exigences associées.

7. DIAGRAMME DE CAS D'UTILISATION "UC : USER CASE"

A partir de la racine du projet on peut créer des diagrammes de cas d'utilisation. Il est conseillé de ne créer qu'un diagramme d'exploitation.



La frontière du système se trouve dans :
use case diagram + paquetage + frontière du système.



Rmq : En mode "expert" (DCG sur acteur + propriété : expert) vous pouvez associer une image à l'acteur (exemple : acteur = soleil = image associé)