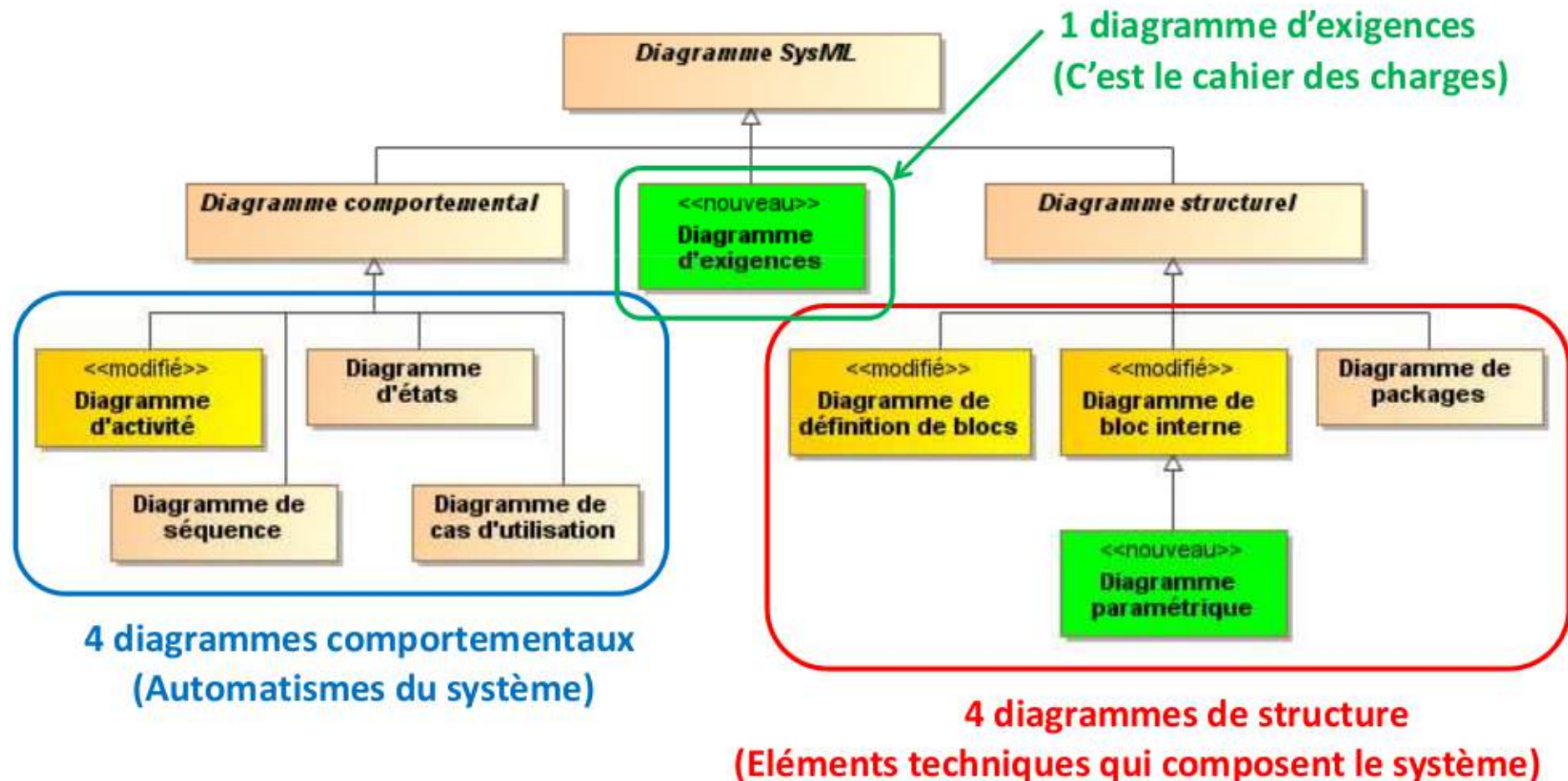




# Les 9 diagrammes SysML



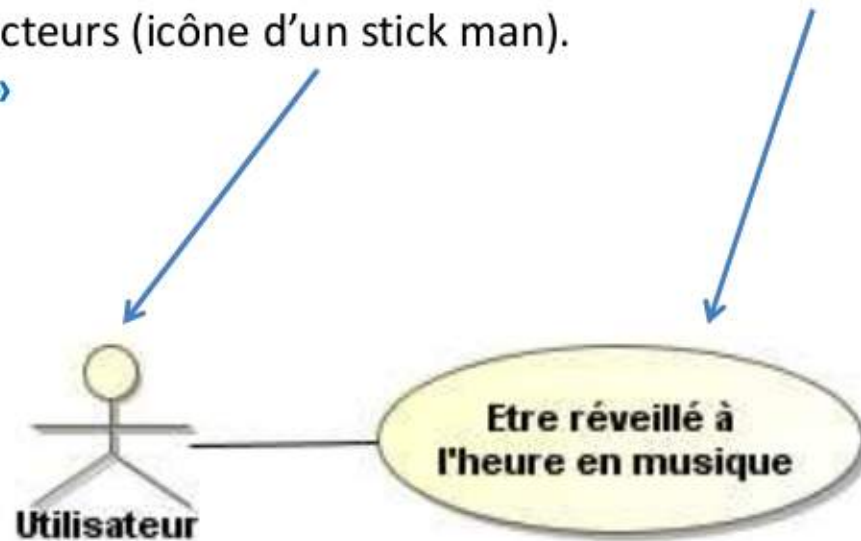
# Le diagramme de cas d'utilisation

Le **diagramme de cas d'utilisation** permet de représenter les **besoins attendus** par un système.


On se place du **point de vue de l'utilisateur**.

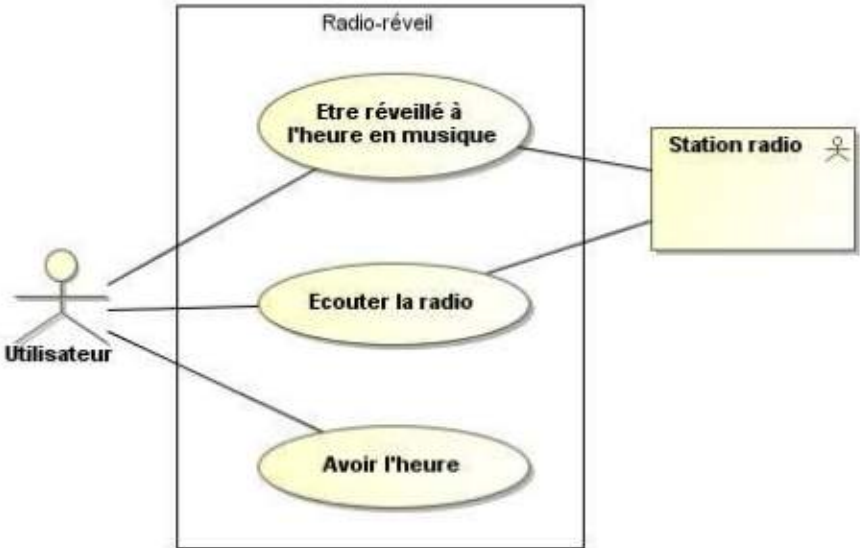
Le **diagramme de cas d'utilisation** est un schéma qui montre les cas d'utilisation (ovales) reliés par des associations (lignes) à leurs acteurs (icône d'un stick man).  
Chaque association signifie : « **participe à** »

**Exemple : radio-réveil à projecteur**



Les **acteurs principaux** sont représentés **à gauche** des cas d'utilisation, et les **acteurs secondaires** à droite. Un acteur **non-humain** est représenté par un **rectangle**.

uc [Modèle] Modèle RadioReveil [  Cas d'utilisation du radio-réveil (suite) ]



```
graph LR
    Utilisateur((Utilisateur)) --- UC1(Etre réveillé à l'heure en musique)
    Utilisateur --- UC2(Ecouter la radio)
    Utilisateur --- UC3(Avoir l'heure)
    subgraph RadioReveil [Radio-réveil]
        UC1
        UC2
        UC3
    end
    UC1 --- StationRadio[Station radio]
```



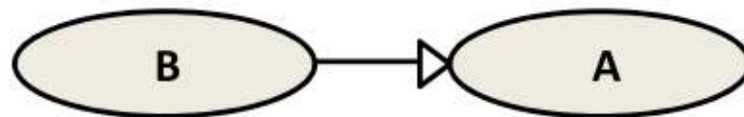


# Le diagramme de cas d'utilisation

On peut imaginer les cas d'utilisation selon que l'utilisateur est réveillé ou endormi.

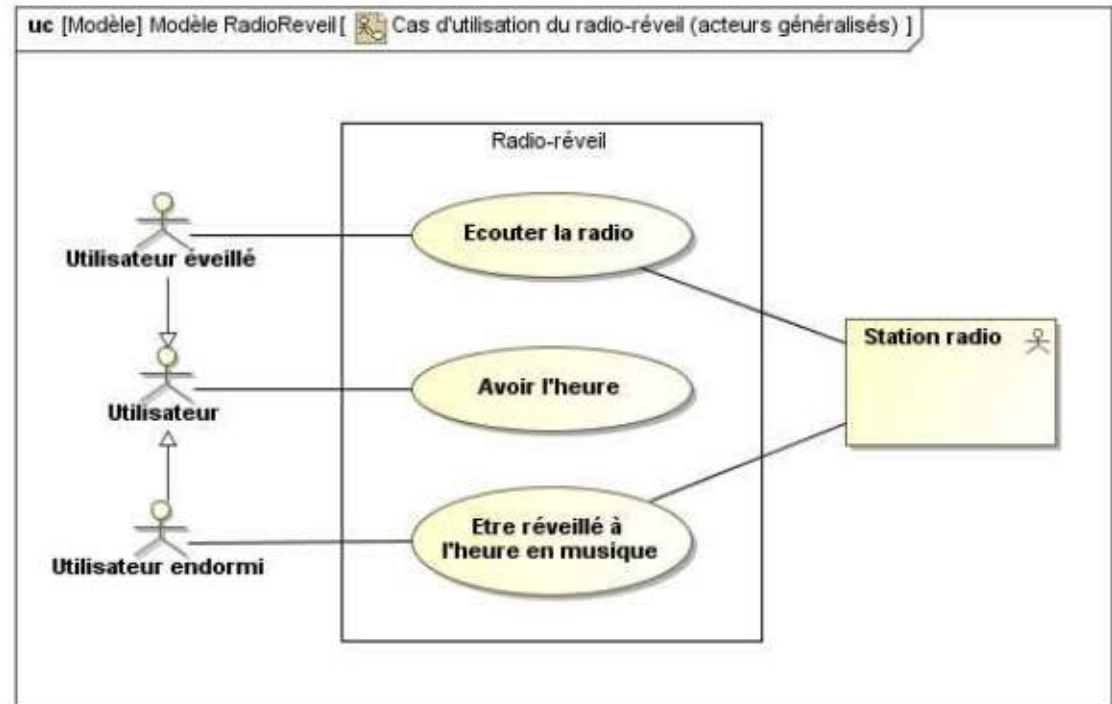
La relation entre les stick man est une relation de **généralisation**.

La relation est symbolisée par une flèche évidée : **B est un cas particulier de A.**



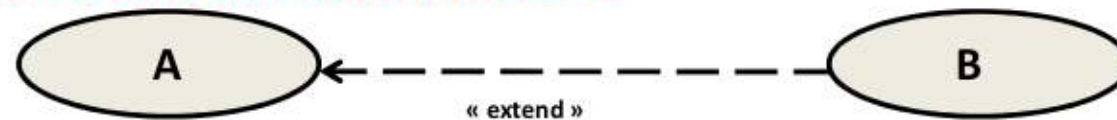
Le diagramme peut être complété :

Exemple : radio-réveil à projecteur



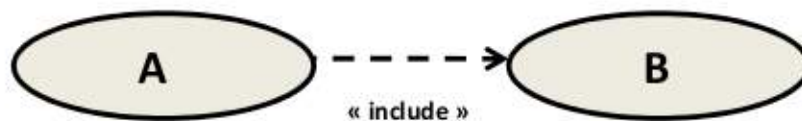
Une **relation d'extension** est formalisée par le mot « **extend** » et une flèche en pointillés.

Le cas B est une extension du cas A si : le cas B peut être appelé au cours de l'exécution du cas A.

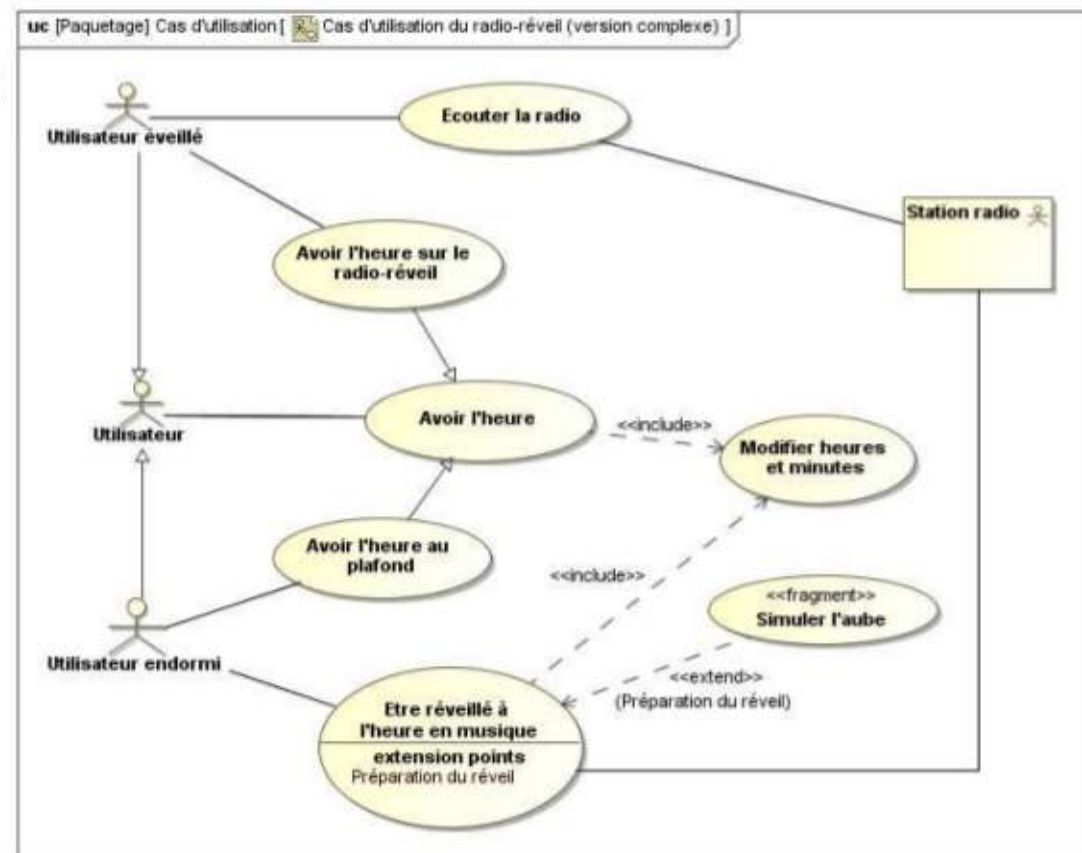


La **relation d'inclusion** est formalisée par le mot « **include** » et par une flèche en pointillés.

Un cas A inclut le cas B si : lorsque A est sollicité, B l'est obligatoirement.



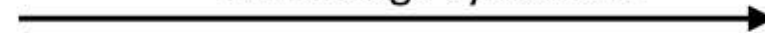
Le diagramme peut être complété :  
Exemple : radio-réveil à projecteur



# Le diagramme de séquence

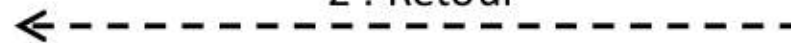
Un message qui est dans l'attente d'une réponse est un **message synchrone**. Il est représenté par une flèche pleine :

1: Message synchrone



La flèche pointillée représente un **retour**. Cela signifie que le message en question est le résultat direct du message précédent. La flèche est évidée.

2 : Retour



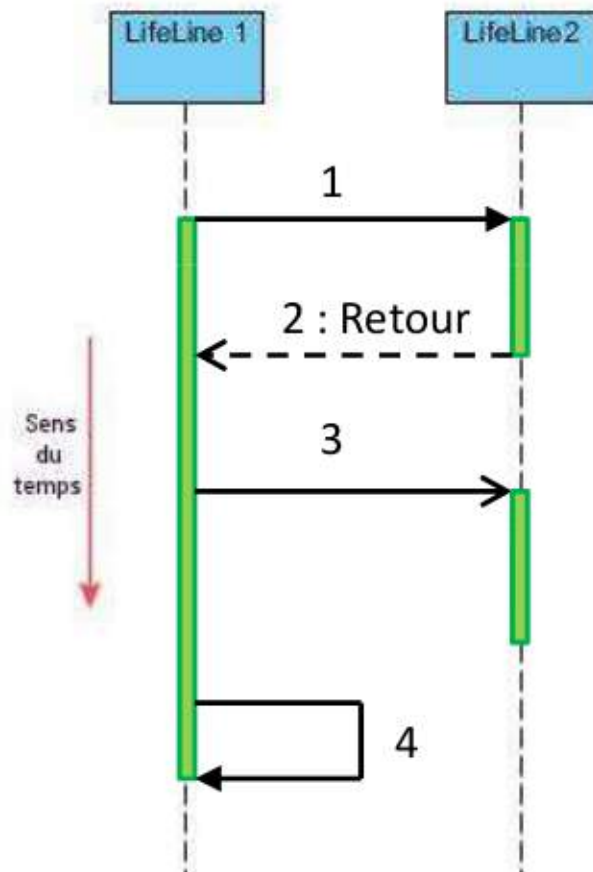
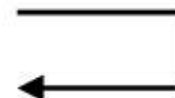
Un **message asynchrone** n'est pas dans l'attente d'une réponse. Il est représenté par une flèche pleine évidée :

3 : Message asynchrone



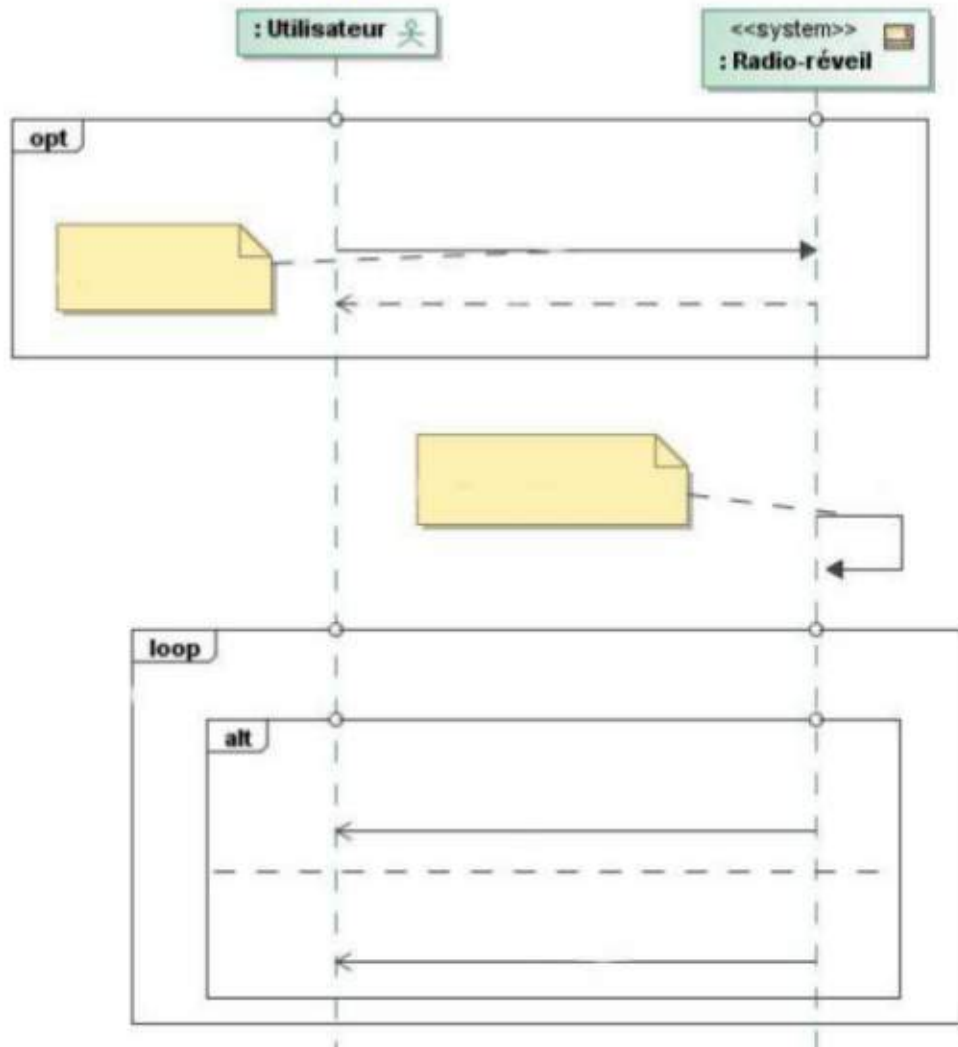
Un **message réflexif** permet de représenter une boucle interne. Il est représenté par une flèche pleine bouclée :

4 : Message réflexif





# Le diagramme de séquence



Opt : **optionnel**

Le fragment ne s'exécute que si la condition fournie est vraie .

Loop : **boucle**

Le fragment peut s'exécuter plusieurs fois .

Alt : **alternatif**

Le fragment s'exécute si la condition fournie est vraie, suivie de sinon : la condition est fausse .

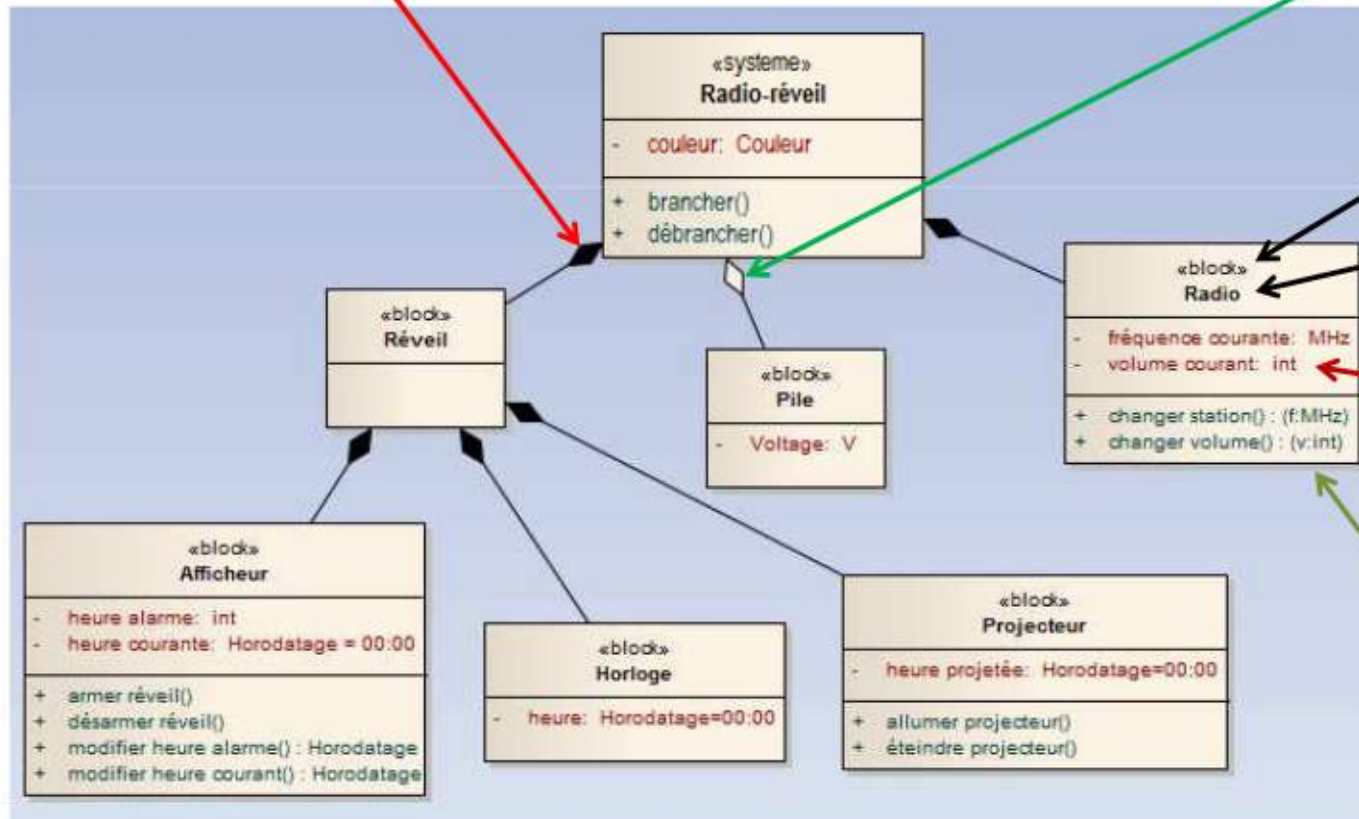
Compléter l'exemple :

**Radio-réveil à projecteur**



# Le Diagramme de Définition des Blocs

Les losanges pleins indiquent l'agrégation avec **un bloc en particulier** et le losange vide indique l'agrégation d'un bloc **partagé** entre plusieurs blocs du diagramme : La pile sera en effet utilisée pour garder en mémoire l'heure courante **et** l'heure de l'alarme.



La **définition** du bloc présente son nom et son type.

Les **attributs** sont les propriétés du bloc.

Les **fonctions** sont les opérations que l'on peut demander au bloc. (verbes)



# Le Diagramme de Définition des Blocs

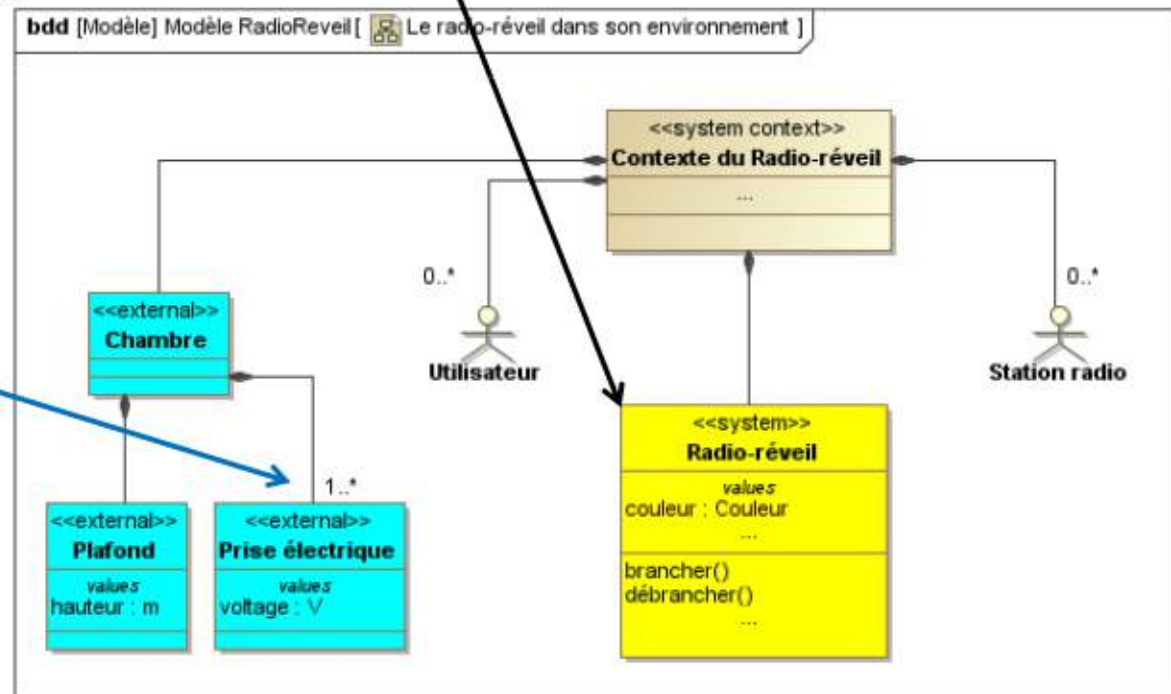
## Diagramme de contexte

D'un point de vue méthodologique, il est souvent intéressant de remonter d'un cran et de modéliser le contexte du **bloc principal** : « **System** » Radio-réveil

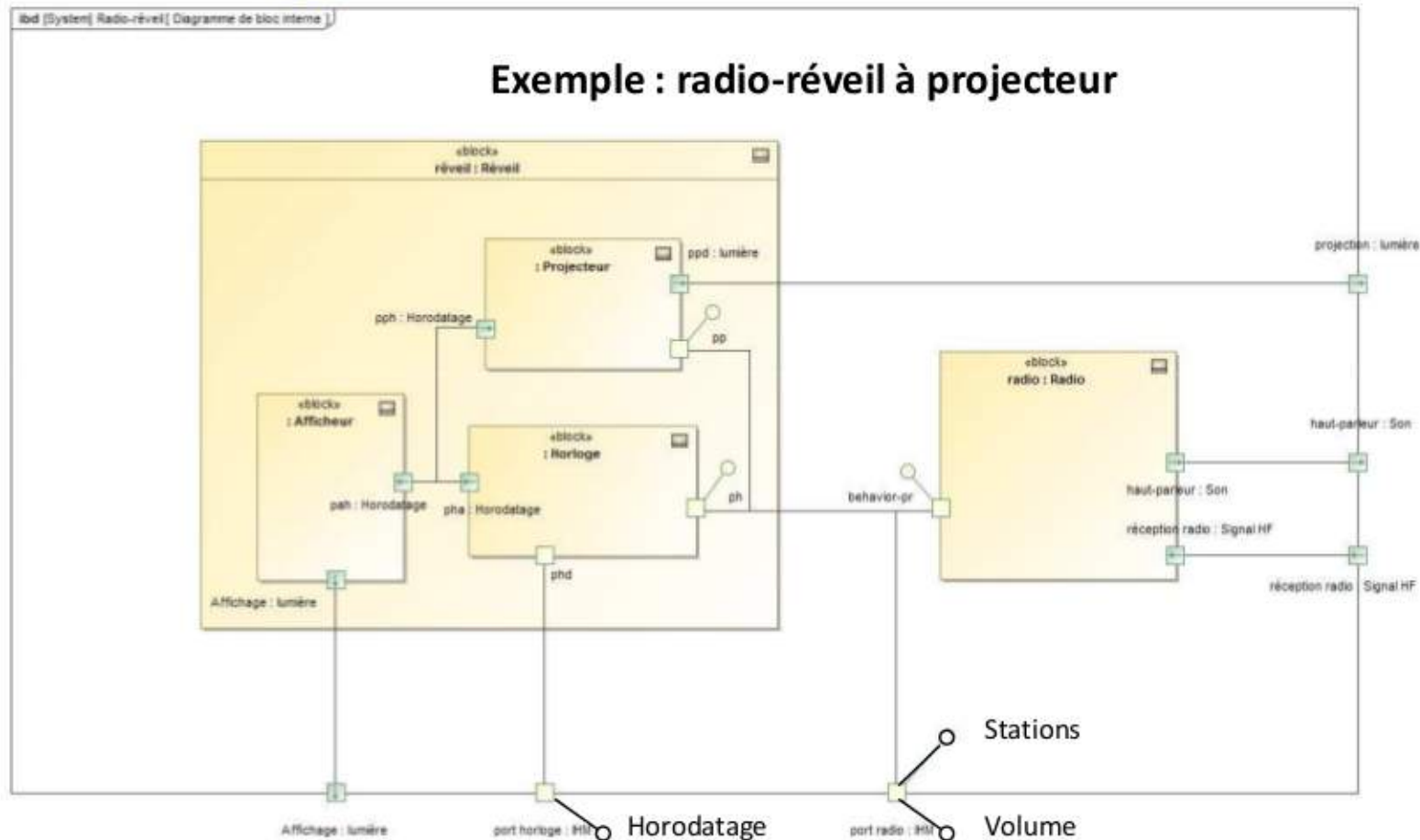
C'est celui qui porte le mot-clé « system » dans le bdd précédent. On appelle ce diagramme le **diagramme de contexte**.

### Exemple : radio-réveil à projecteur

A chaque extrémité d'un lien, on peut indiquer la **multiplicité** d'un bloc :  
Il y a 1 à n prises dans la chambre.

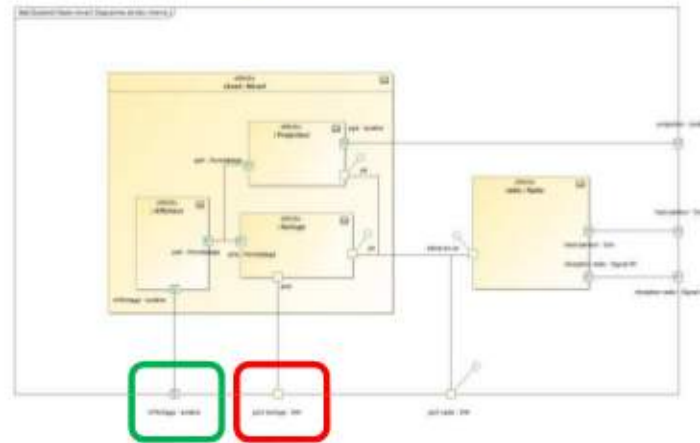


# Le diagramme de bloc interne « ibd »



Les interfaces sont représentées par un cercle et un trait : volume, stations ...

# Le diagramme de bloc interne « ibd »



Le diagramme de bloc interne (ibd : « internal block Diagram ») décrit la vue interne d'un bloc, et se base sur le BDD (Diagramme de Définition des Blocs) pour représenter l'assemblage des blocs.

Un bloc peut avoir plusieurs ports qui spécifient des points d'interaction différents.

**Les ports peuvent être de deux natures :**



port horloge : IHM

**Standard :** ce type de port autorise la description de services logiques entre les blocs, au moyens d'interfaces regroupant les opérations.  
Exemple : IHM (Interface Homme – Machine)

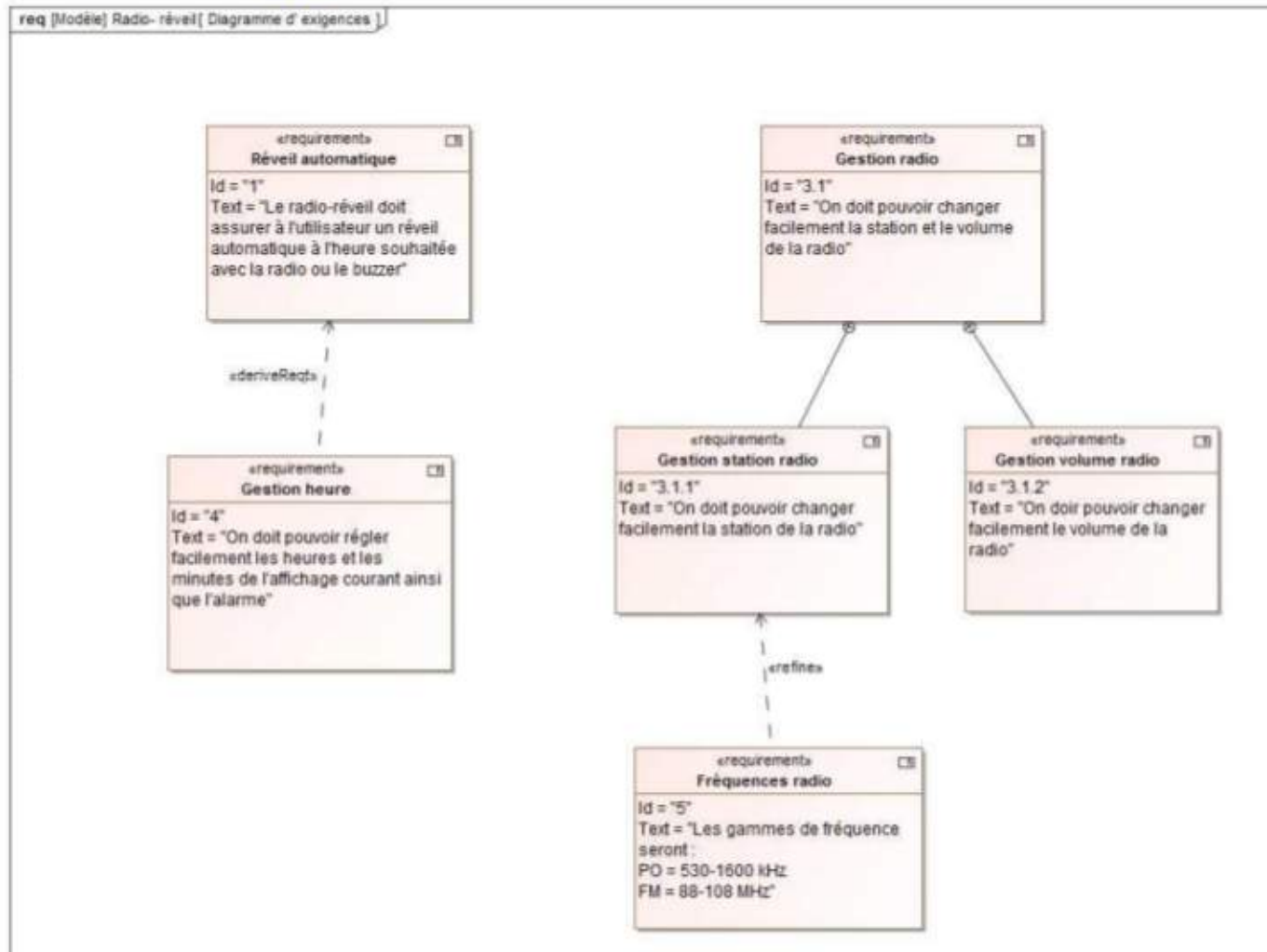


Affichage : lumière

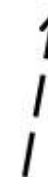
**Flux (flow port) :** ce type de port autorise la circulation de flux physiques entre les blocs. La nature de ce qui peut circuler va des fluides aux données, en passant par l'énergie.



# Le diagramme d'exigences



Lié à,  
inclus dans...



« deriveReq » =  
relier des  
exigences de  
niveaux différents



« refine » = ajout  
de compléments