Un poil de méthodo

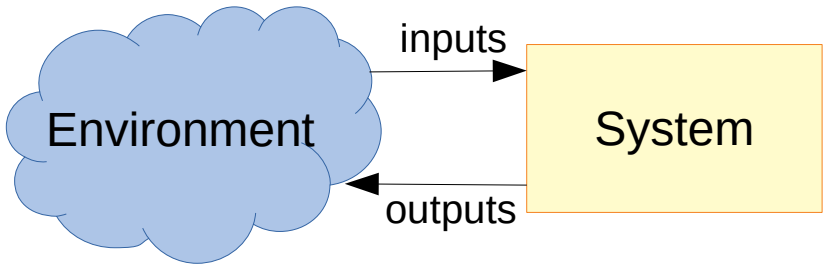
En lien avec le projet...

Identifier le système et son environnement

- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel

Identifier le système et son environnement

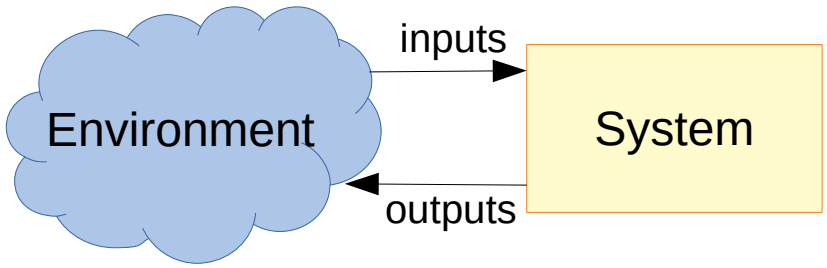
- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel
- Il faut identifier ce qui fait partie du système et ce qui fait partie de l'environnement du système.



Identifier le système et son environnement

- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel
- Il faut identifier ce qui fait partie du système et ce qui fait partie de l'environnement du système.Dans le cas de l'exemple du portail,...

We want to model the controller of an entry door by using a FSM.



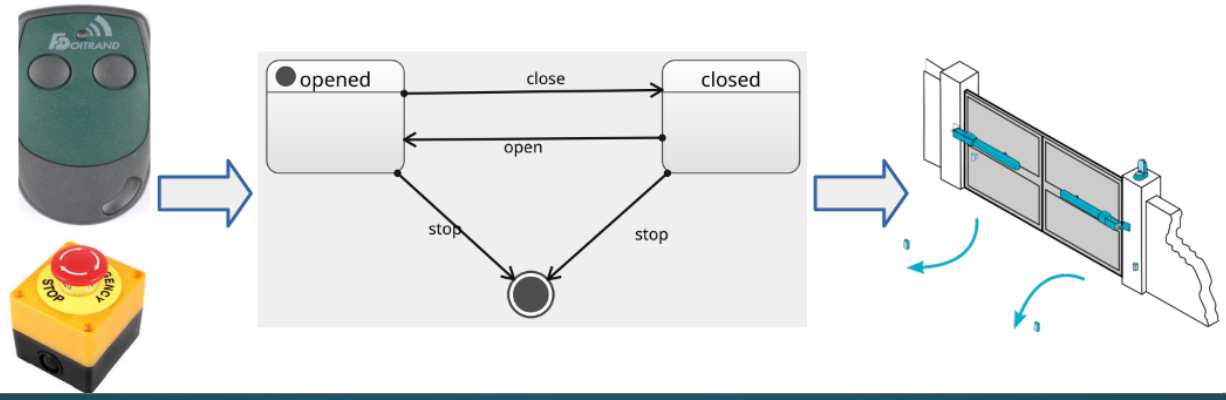
Identifier le système et son environnement

- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel
- Il faut identifier ce qui fait partie du système et ce qui fait partie de l'environnement du système. Dans le cas de l'exemple du portail, la télécommande et le bouton ne font pas partie du système. De plus, seuls les données/événements en relation avec le système sont intéressants.

We want to model the controller of an entry door by using a FSM.

Σ_I = {open, close, stop}

Σ_O = {doOpen, doClose}



Identifier le système et son environnement

- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel
- Il faut identifier ce qui fait partie du système et ce qui fait partie de l'environnement du système. Si on modélisait le contrôleur de la télécommande ce serait différent…

We want to model the controller of a remote commande.

Identifier le système et son environnement

- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel
- Il faut identifier ce qui fait partie du système et ce qui fait partie de l'environnement du système. Si on modélisait le contrôleur de la télécommande ce serait différent…

We want to model the controller of a remote commande.

Σ_I = {b1, b2}

Identifier le système et son environnement

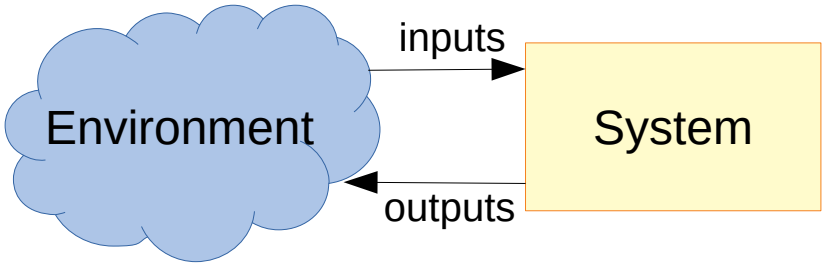
- Pour tout ce qui suit, cela peut être fait de différentes manières selon les sensibilités. Ici peu importe nous resterons très informel
- Il faut identifier ce qui fait partie du système et ce qui fait partie de l'environnement du système. Si on modélisait le contrôleur de la télécommande ce serait différents…

We want to model the controller of a remote commande.

Σ_I = {b1, b2}

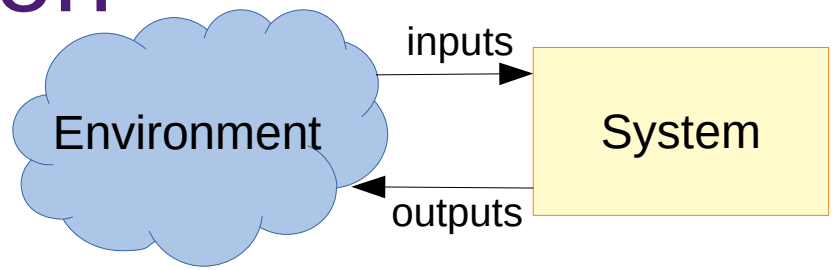
Σ_O = {openLeft, openAll, close}

Identifier le système et son environnement

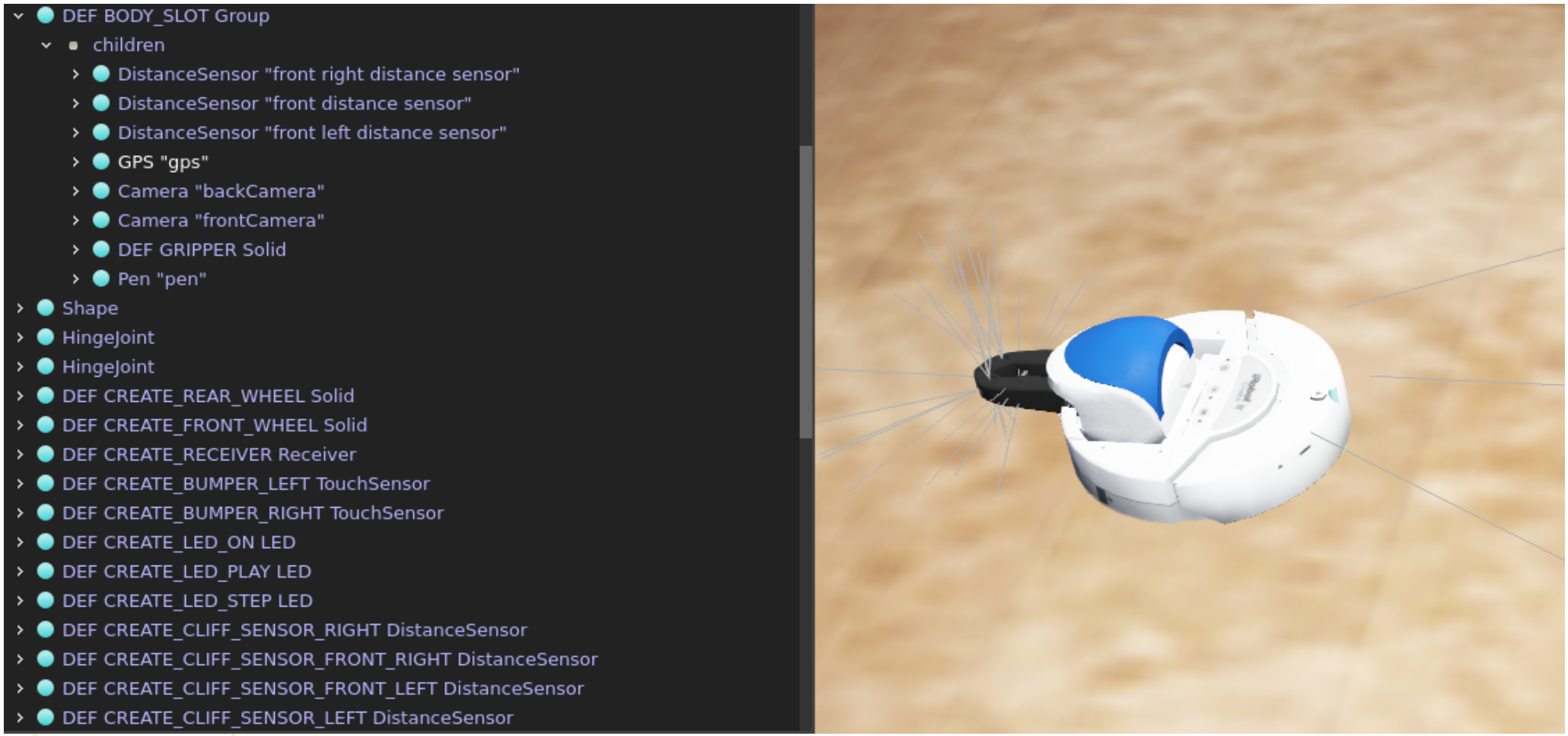


- Dans l'exemple du robot ménager, l'environnement est composé de…

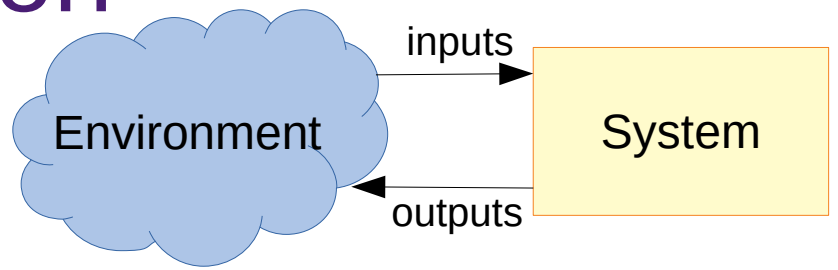
Identifier le système et son environnement



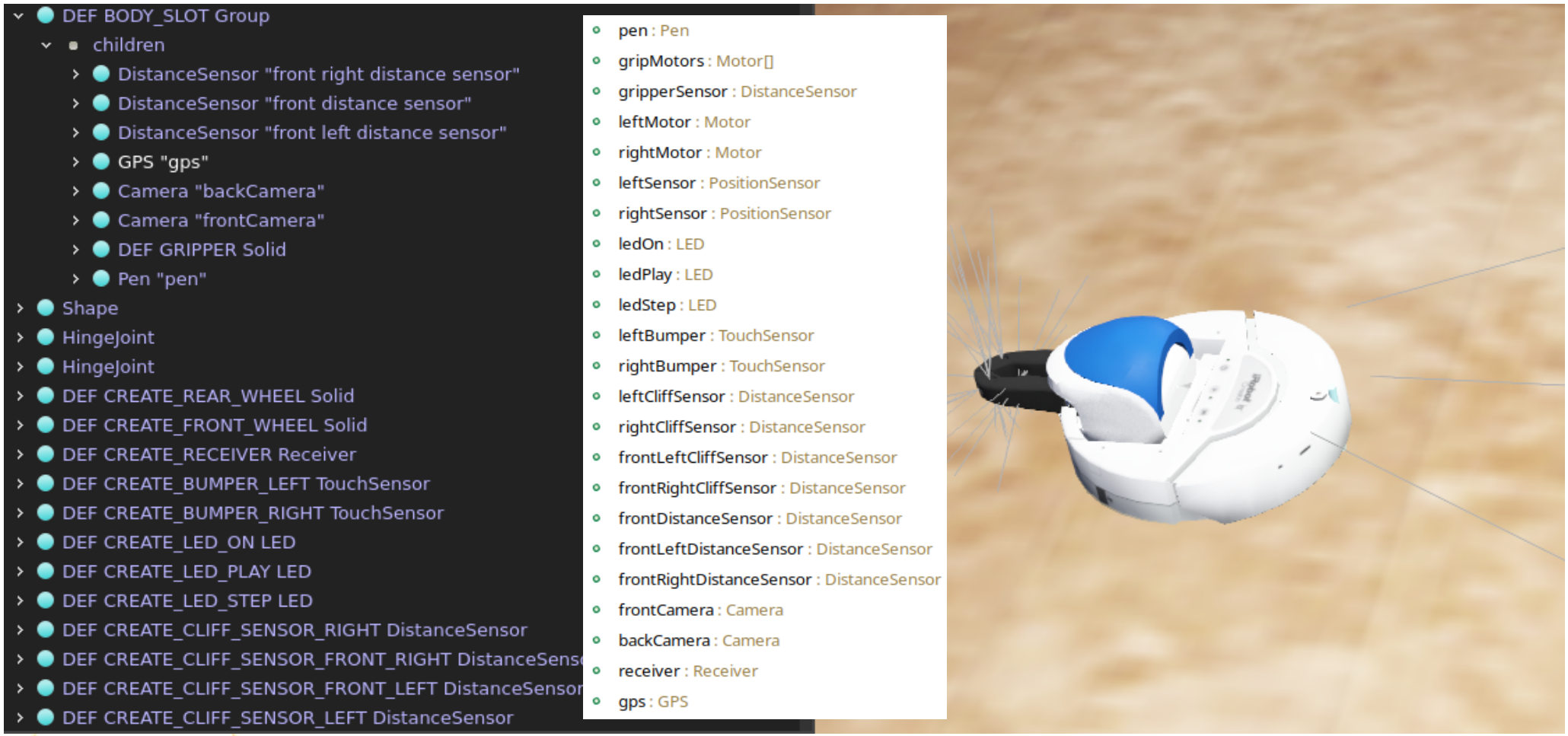
- Dans l'exemple du robot ménager, l'environnement est composé de l'ensemble capteurs/actionneurs liés au robot



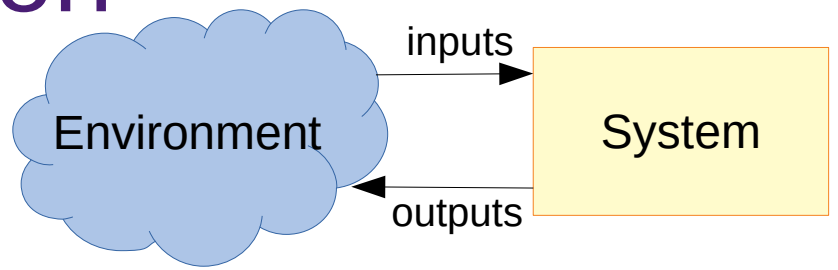
Identifier le système et son environnement



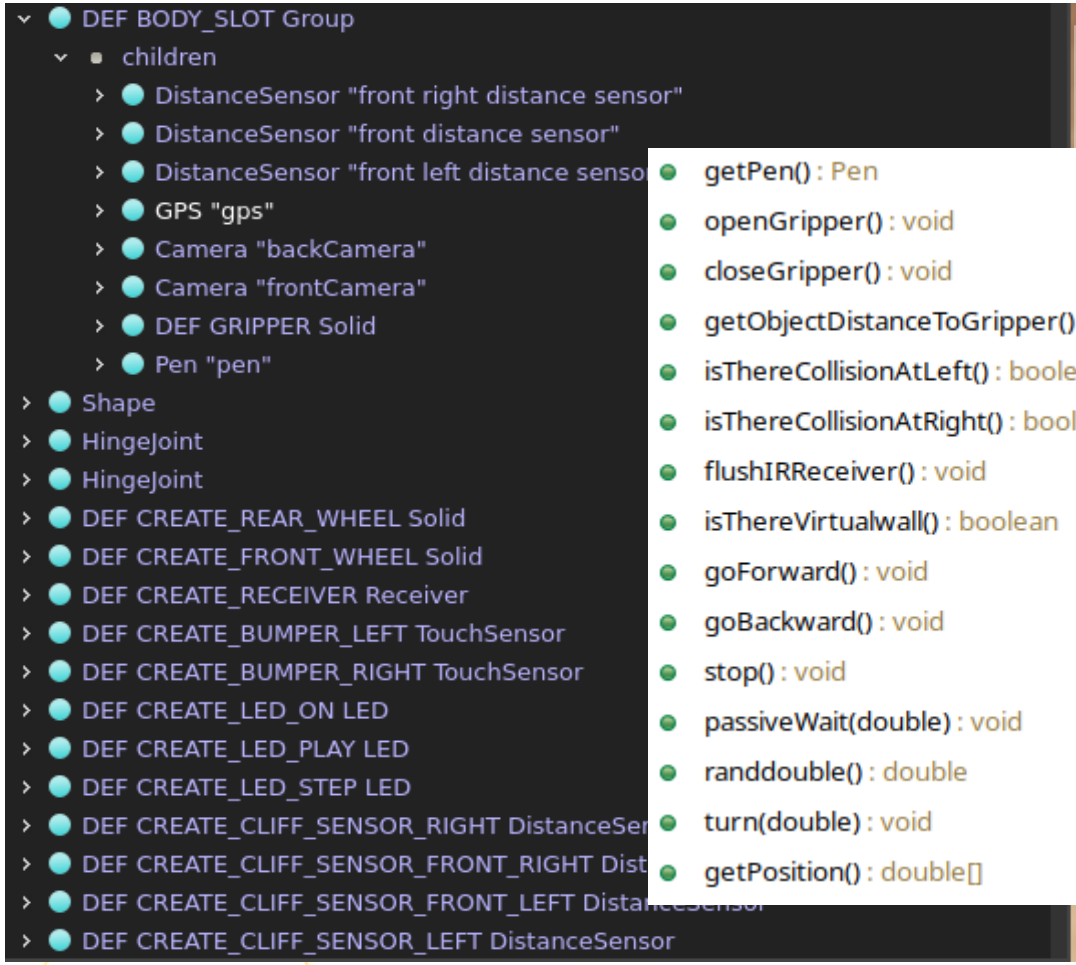
- Dans l'exemple du robot ménager, l'environnement est composé de l'ensemble capteurs/actionneurs liés au robot



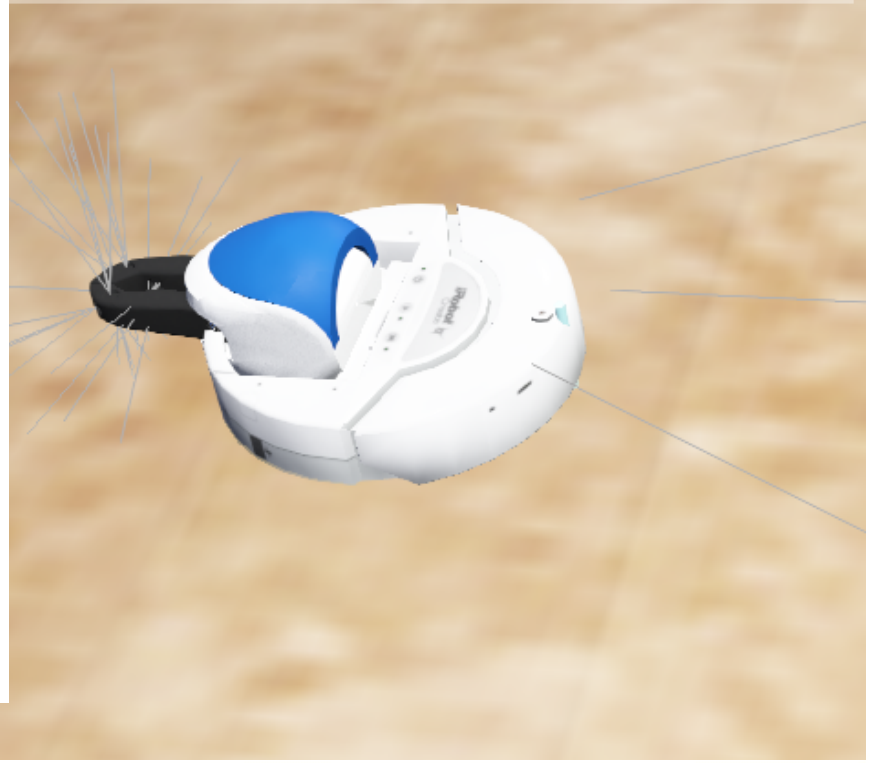
Identifier le système et son environnement



- Dans l'exemple du robot ménager, l'environnement est composé de l'ensemble capteurs/actionneurs liés au robot



+ exactement c'est l'API permettant d'utiliser les capteurs/Actionneurs qui sera intéressante



Plus globalement...

- Vous avez le choix pour cumuler vos différents state chart:
  - un state chart par fonctionnalité gérant les différentes préoccupations ?
  - Un par préoccupation gérant toute les fonctionnalités ?
  - Plein de state charts en réseau ?
  - Un mélange approprié des points précédents ?

Plus globalement...

- N'hésitez pas à commencer par réfléchir indépendamment de l'outil
- N'hésitez pas à faire de petits tests
  - De petits state charts permettant de tester certaines de vos idées de conceptions.
  - Tester les différentes stratégies précédentes.
  - …

Plus globalement...

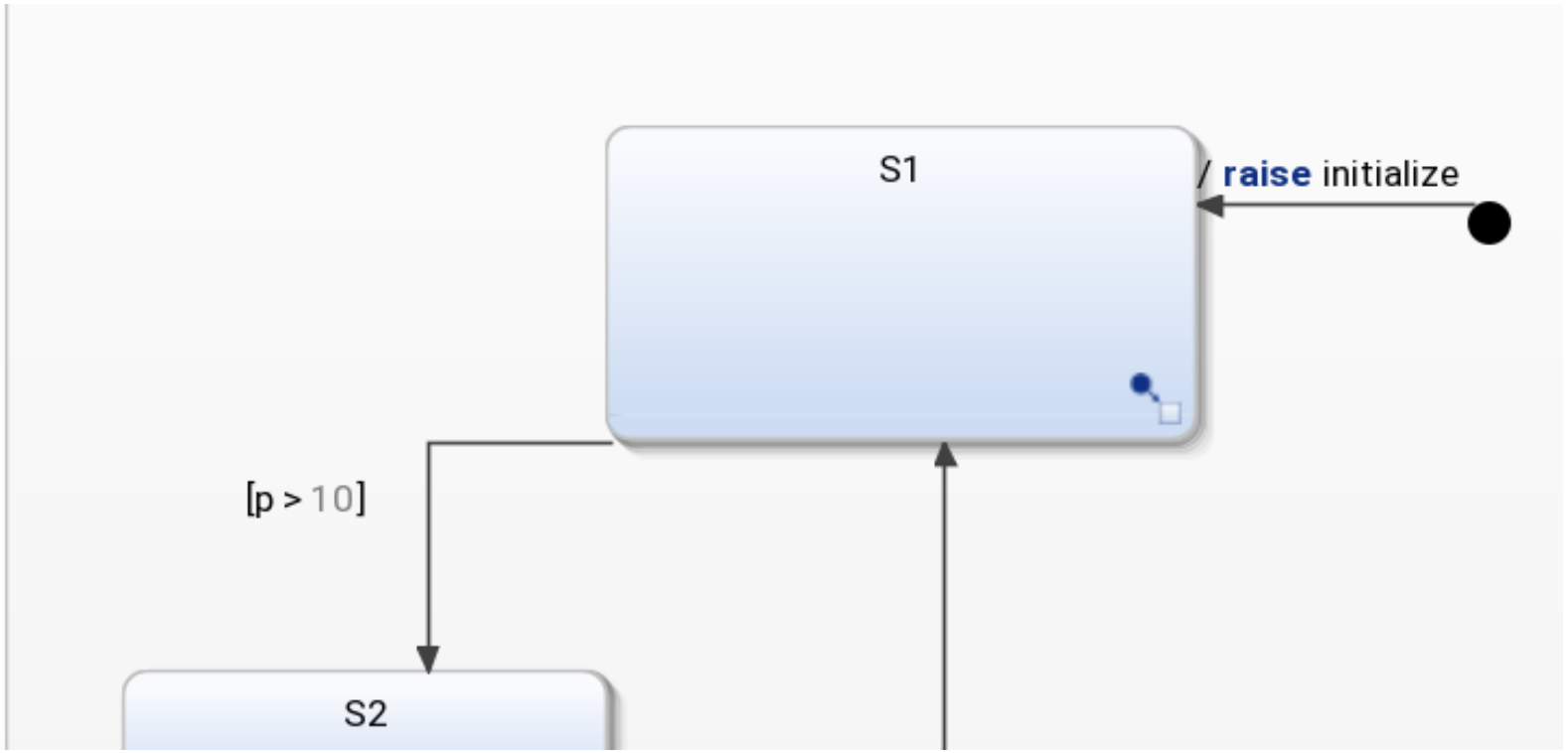
- Dans certaines mesures, il est possible de lire/écrire des données définies dans le state chart depuis le code.
  - À priori à éviter mais cela peut s'avérer pratique si l'on veut utiliser des gardes boolénnes sur les transitions

```
@EventDriven
// Use the event driv
// Runs a run-to-cor
// each time an eve
// Switch to cycle ba
// by specifying '@C
// instead.

@ChildFirstExecutio
// In composite stat
// child states first.
// @ParentFirstExec
interface:

var p : integer

in event bombRea
in event bombGra
```



Plus globalement...

- Dans certaines mesures, il est possible de lire/écrire des données définies dans le state chart depuis le code.
  - À priori à éviter mais cela peut s'avérer pratique si l'on veut utiliser des gardes boolénnes sur les transitions

```
@EventDriven
// Use the event driv
// Runs a run-to-cor
// each time an eve
// Switch to cycle ba
// by specifying '@C
// instead.

@ChildFirstExecutio
// In composite stat
// child states first.
// @ParentFirstExec
interface:

var p : integer

in event bombRea
in event bombGra
```

```
private long p;

public long getP() {
    return p;
}

public void setP(long value) {
    this.p = value;
}
```