

SEQUENTIEL

1) Introduction

2) Le diagramme des cas d'utilisation

3) Le diagramme de séquence

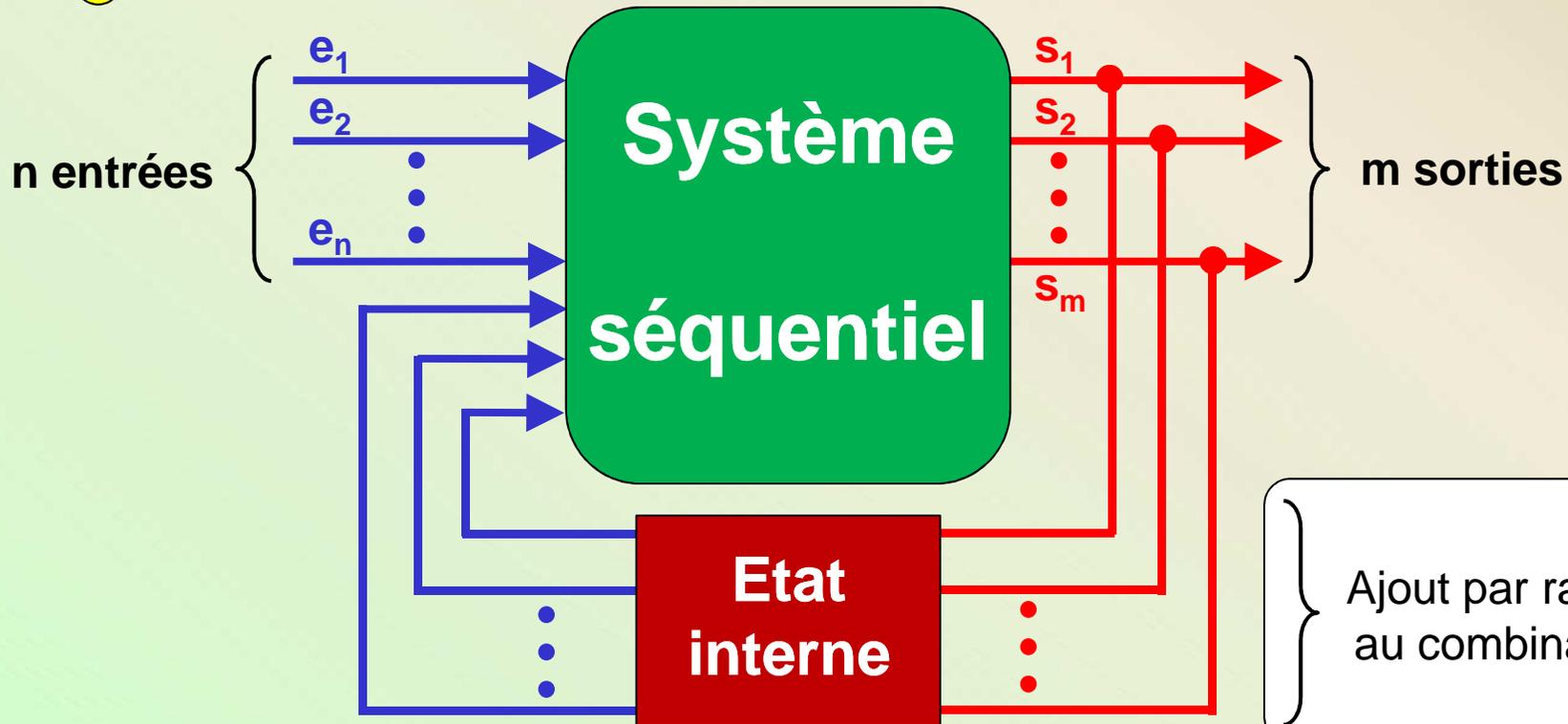
4) Le diagramme d'état

← Sera vu dans un prochain cours.

1) Introduction

► Définition :

Un système séquentiel est un système à évènements discrets dans lequel l'état des variables de sortie dépend de l'état des variables d'entrée et des évolutions passées.



Il existe un état interne qui est fonction des évolutions passées, donc des variables de sortie.

Un système séquentiel évolue étape par étape (« état »)



nécessité de mémoires

Autrement dit :

À une même combinaison des variables d'entrée peut correspondre des situations différentes pour les variables de sortie.

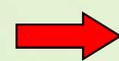
-  *Un état est actif ou non.*
-  *Plusieurs états peuvent être actifs en même temps.*
-  *Une variable (information booléenne) est à l'origine de l'activation (ou désactivation) d'un état.*

▶ Exemple : ➔ *distributeur de boissons chaudes*

*Le bouton poussoir (b) pour demander un espresso est monostable
(un seul état stable)*

➔ *quand on appuie dessus il passe de 0 à 1 et donne ainsi
l'information de la demande, mais quand on le relâche il
revient automatiquement à 0 (son état initial) alors que
l'espresso se prépare.*

➔ *donc le même état 0 de ce bouton correspond, soit à rien,
soit à une demande déjà effectuée (espresso en préparation).*



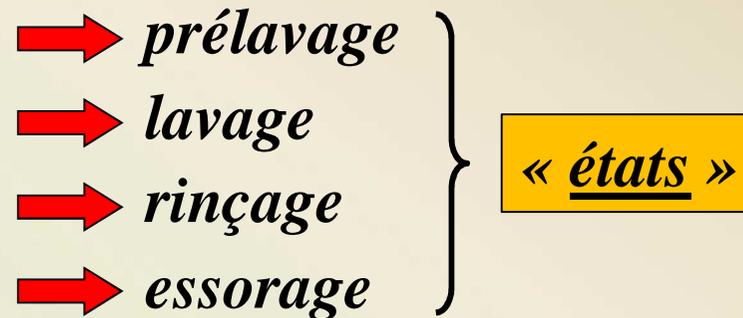
nécessité d'une mémoire



Idem pour la commande de l'éclairage dans une cage d'escalier :
bouton poussoir (différent d'un interrupteur à deux positions)
➔ une mémoire (appelée relais en électricité) mémorise
l'état allumé ou éteint des ampoules.

▶ Exemple d'un lave-linge :

La succession des opérations est la suivante :



L'évolution du cycle de fonctionnement se déroule selon une succession de plusieurs états (séquences d'où le nom séquentiel).

Un état représente une période de la vie du système durant laquelle une ou plusieurs actions (« activités ») se produisent.

- ▶ Description d'un système séquentiel : avant de décrire l'évolution des différents états que pourra prendre un système, il est nécessaire de partir du besoin donné par le commanditaire (le client).

➔ 3 étapes successives :

👉 le diagramme des cas d'utilisation (UCD)

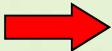
👉 le diagramme de séquence (SD)

👉 le diagramme d'état (SMD)

Langage
Sysml

2) Le diagramme des cas d'utilisation (UCD)

Modéliser le(s) service(s) rendu(s) par le système à un ou plusieurs acteurs.

 *Ce diagramme permet de répondre à la question :*

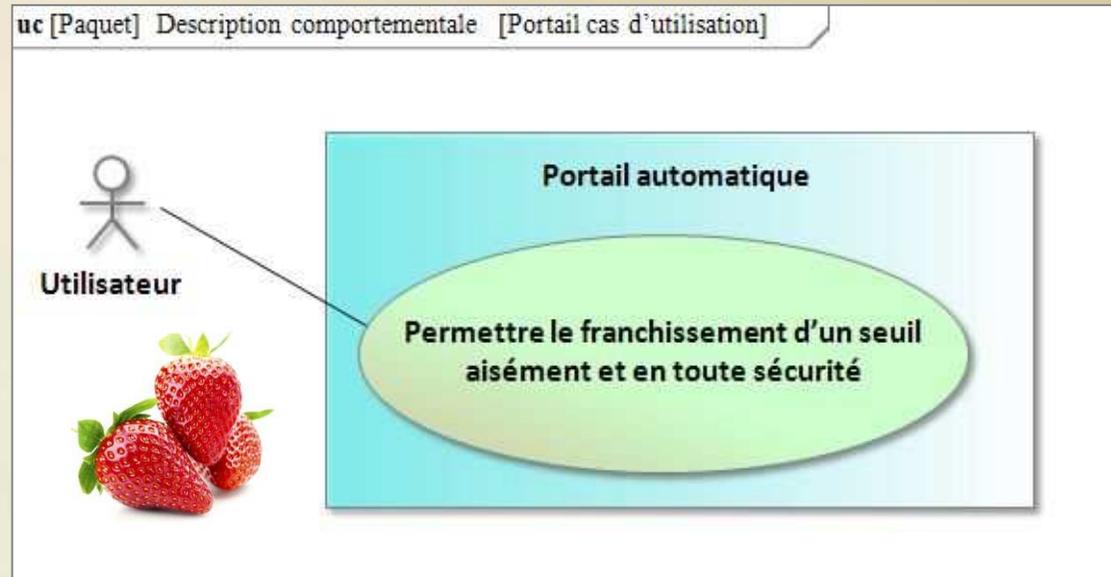
Quel(s) service(s) rend le système et avec qui il interagit ?

Nota :  *permet le dialogue entre concepteur du système et commanditaire (le client).*

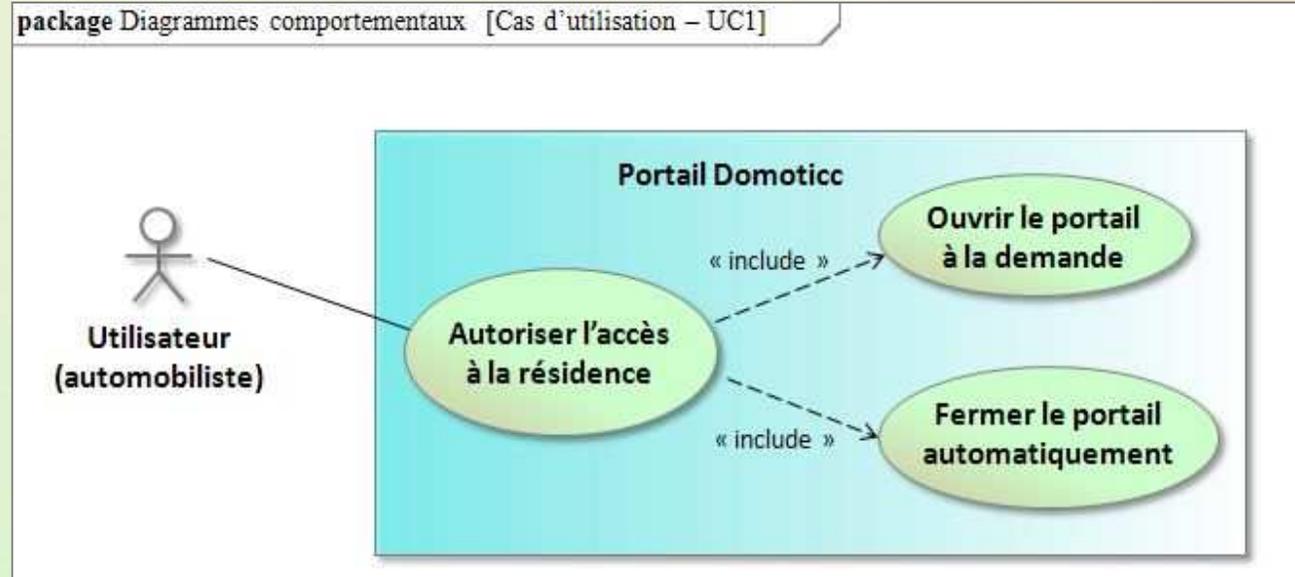
 *sous-entend éventuellement un scénario.*

► Exemple du portail automatique :

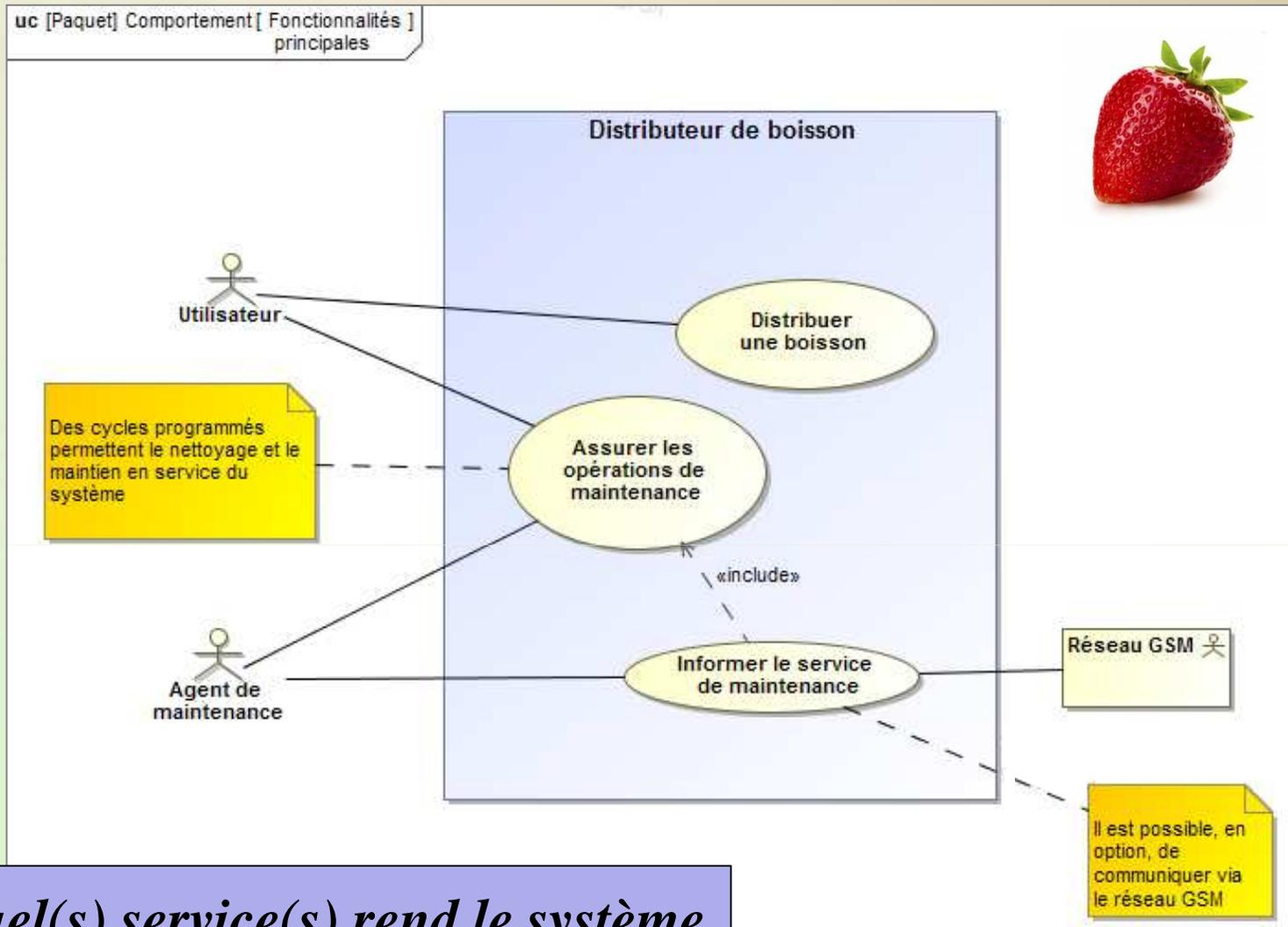
*Plusieurs présentations
sont possibles.*



*Un scénario peut
ici être amorcé.*



► Exemple du distributeur de boissons chaudes :



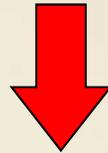
Quel(s) service(s) rend le système et avec qui il interagit ?

Ici: l'utilisateur, l'agent de maintenance et le réseau GSM.



3) Le diagramme de séquence (SD)

Le diagramme des cas d'utilisation permet de «décrire» ce que le système doit réaliser mais n'explique pas comment.



C'est le rôle du diagramme de séquence qui va décrire explicitement un scénario utilisé.

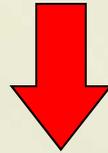
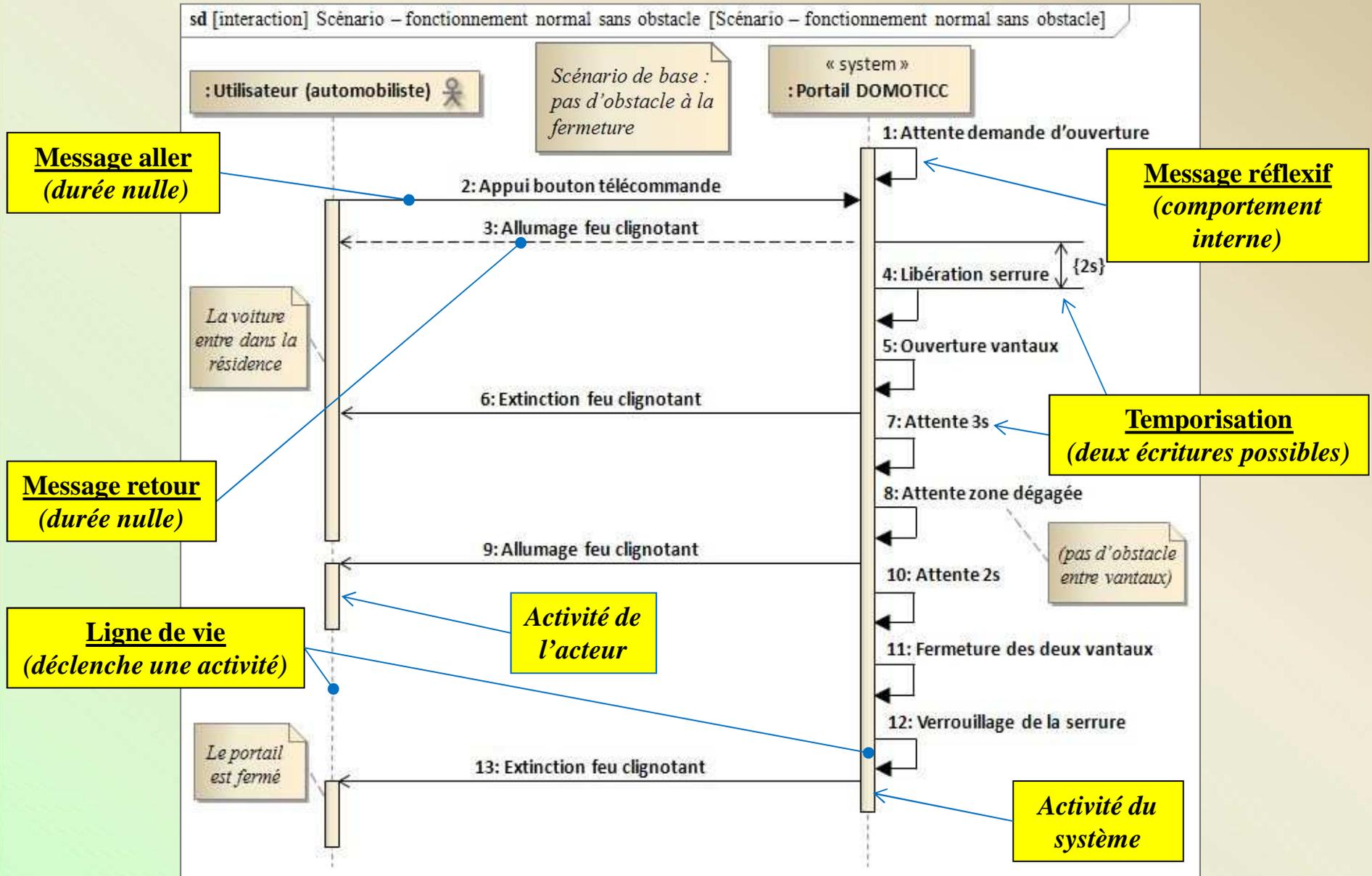


Diagramme temporel qui permet d'identifier les différentes opérations nécessaires.

Nota : *on peut envisager un scénario pour le fonctionnement normal, un pour un dysfonctionnement, un autre en cas de panne énergétique...*

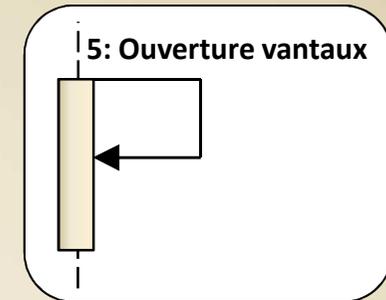
► Exemple du portail automatique à deux vantaux :



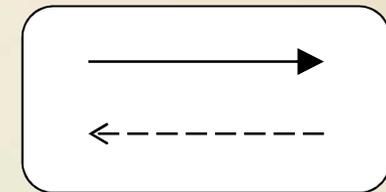


Les messages : une flèche représente un message (information, matière ou énergie)

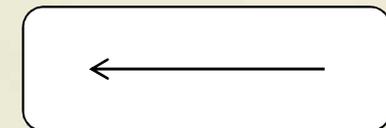
➔ **Message réflexif** : pour supporter une opération réalisée par le système et correspondant à une période d'activation du système.

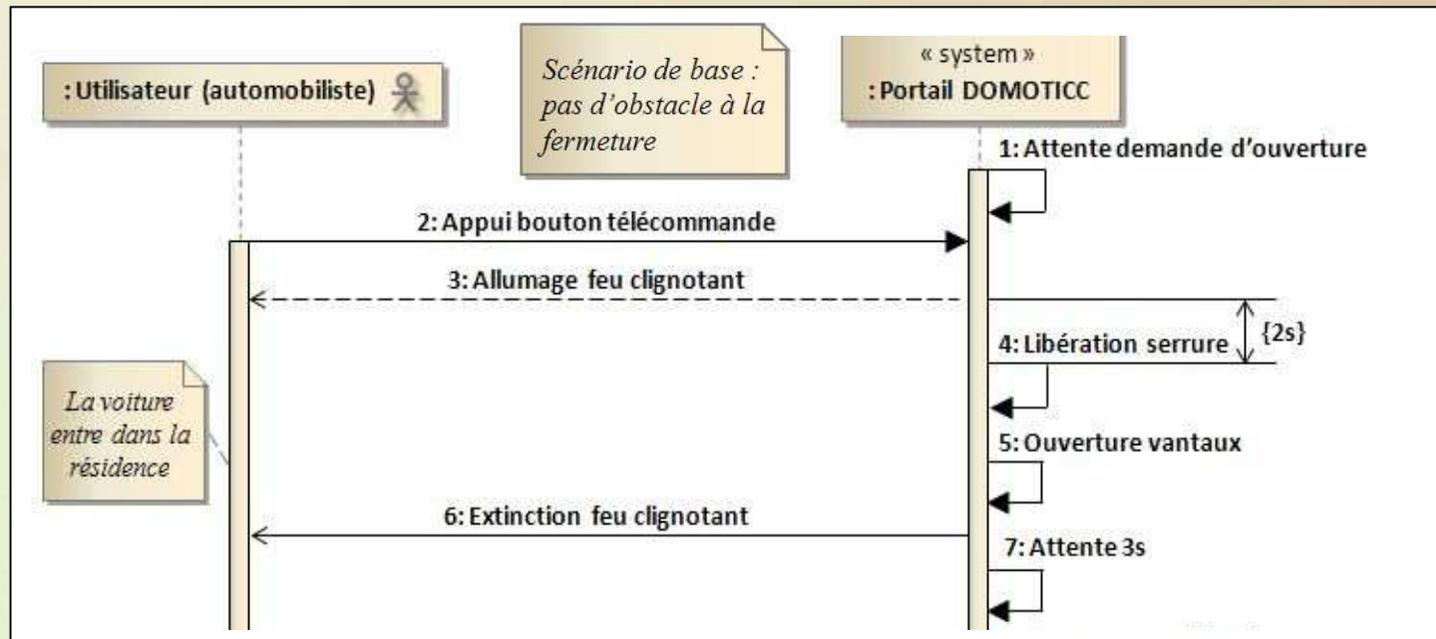


➔ **Message synchrone** : nécessite une réponse (retour) et représenté par une flèche pleine. La réponse est une flèche ouverte en pointillés.

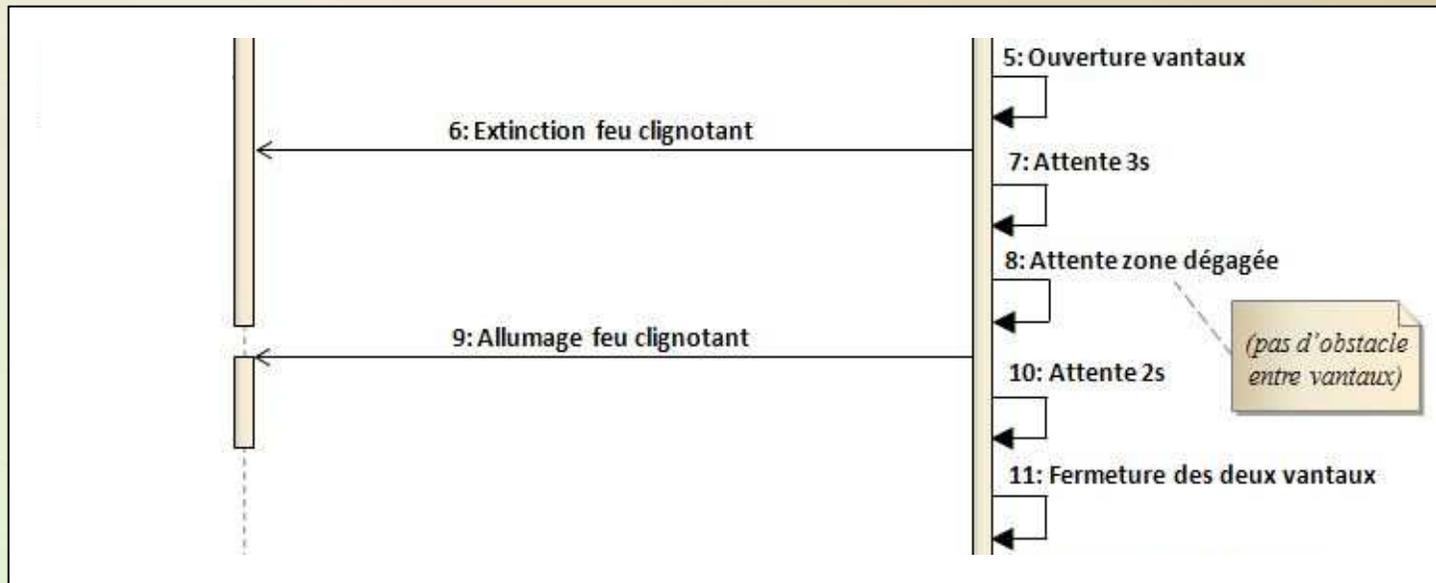


➔ **Message asynchrone** : l'émetteur n'attend pas de réponse. La flèche est ouverte.

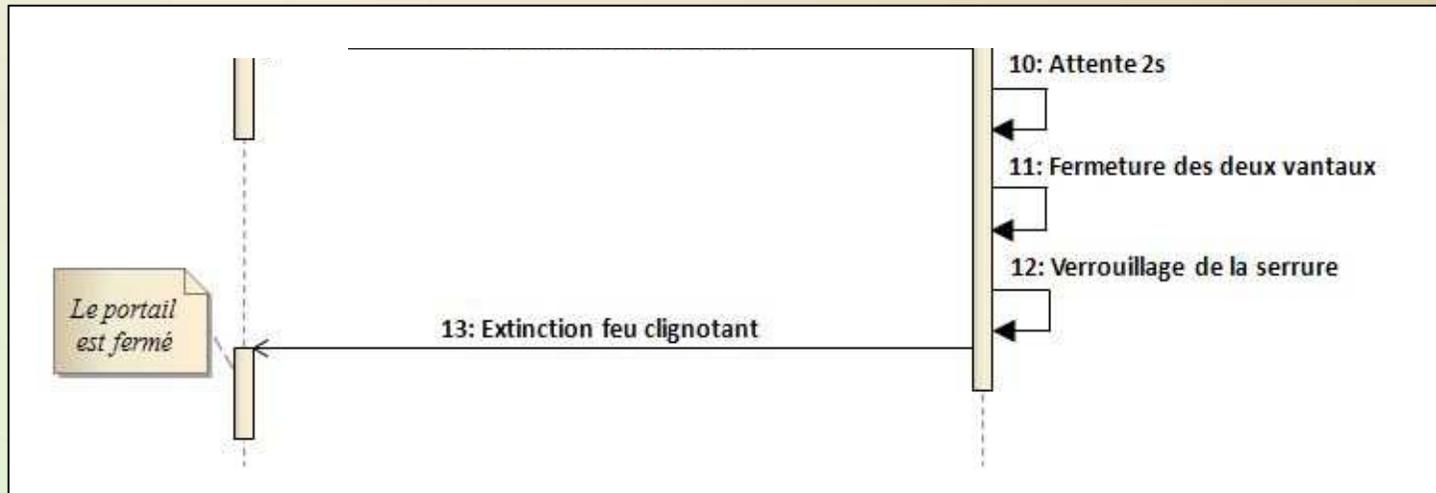




- 1 : début du cycle, pas d'action commandée, le système est à l'arrêt.
- 2 : l'utilisateur appui sur bouton télécommande \Rightarrow message à durée théoriquement nulle demandant une réponse (flèche pleine), de l'utilisateur vers le système.
- 3 : ordre d'allumage immédiat du feu clignotant (réponse du système vers l'utilisateur, l'ordre est de durée théoriquement nulle) et lancement d'une tempo de **2 s**.
- 4 : après **2 s** (depuis l'appui sur télécommande) \Rightarrow action de libération de la serrure.
- 5 : après action de libération serrure effectuée \Rightarrow début ouverture des vantaux.
- 6 : à la fin de l'ouverture des vantaux \Rightarrow ordre d'extinction du feu (message retour, ordre à durée nulle).



- 7** : lancement d'une tempo de **3 s** sans action de la part du système (dès la fin de l'ouverture des vantaux puisque la durée du message retour **6** est nulle).
- 8** : attente tant qu'il y a un obstacle (cellule photoélectrique de détection d'un obstacle) ➡ aucune action de la part du système.
- 9** : dès que la zone est dégagée ➡ ordre d'allumage du feu clignotant et lancement d'une tempo de **2 s**.
- 10** : attente de l'écoulement de la tempo de **2 s** sans aucune action du système.
- 11** : après l'écoulement de la tempo de **2 s** ➡ début de fermeture des vantaux.

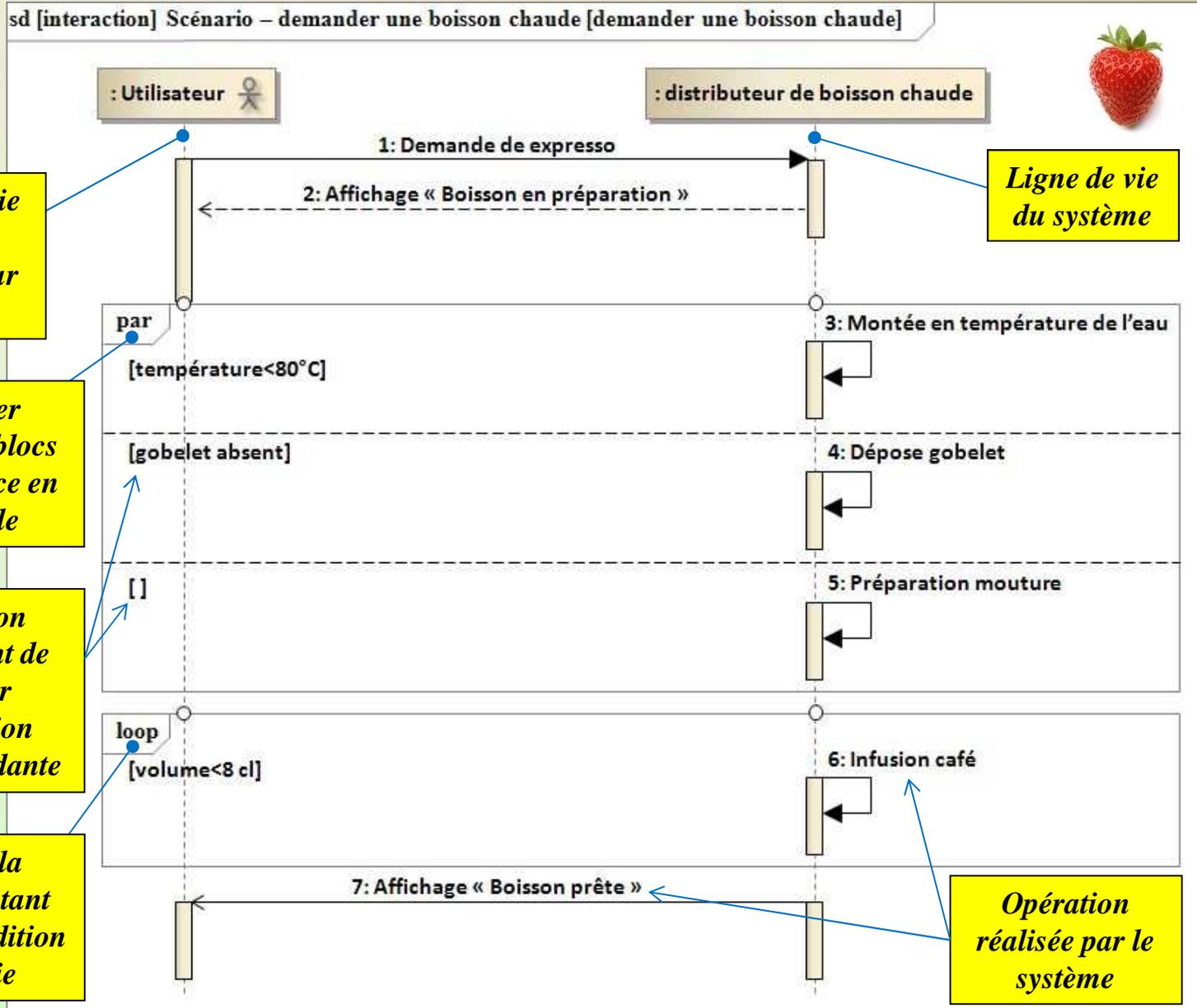


11 : début fermeture des vantaux après l'écoulement de la tempo de **2 s**.

12 : verrouillage après la fermeture des vantaux.

13 : ordre d'extinction du feu clignotant dès la fin du verrouillage (message retour).

Exemple du distributeur de boissons chaudes :



Ligne de vie de l'utilisateur (acteur)

Exécuter plusieurs blocs de séquence en parallèle

Condition permettant de réaliser l'opération correspondante

Répète la séquence tant que la condition est vraie

Ligne de vie du système

Opération réalisée par le système

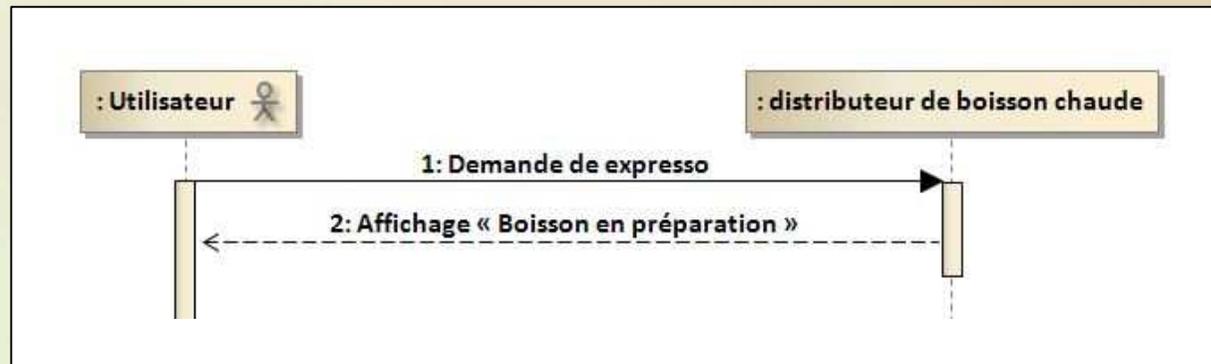




Les fragments combinés :

utilisés pour montrer des variantes dans le scénario.

- ➔ « par » : exécute plusieurs blocs de séquence en parallèle.
- ➔ « loop » : répète la séquence tant que la [condition] est vraie.
- ➔ « opt » : exécute le contenu du bloc si la [condition] est vraie.
- ➔ « alt » : exécute le premier bloc si la [condition] est vraie, sinon c'est le deuxième.



1 : action de demander un espresso : jeton, bouton ou autre (non précisé).

➡ message aller demandant une réponse (flèche pleine, durée nulle).

2 : ordre d'affichage du message «*boisson en préparation*».

➡ message retour en réponse au précédent (flèche ouverte en pointillés, durée nulle).



loop : ce fragment comporte la séquence **6** (action de faire passer l'eau chaude au travers de la mouture) qui commence dès que l'on quitte le fragment précédent et se répète (dure) tant que le volume n'atteint pas **8 cl** (condition de garde).

7 : dès que la condition de garde «*volume < 8cl*» n'est plus vraie (donc qu'on a **8 cl** dans le gobelet) on quitte le fragment «*loop*» (donc arrêt de l'infusion) et on donne l'ordre d'afficher «*boisson prête*».

Ce qu'il faut avoir retenu (minimum « vital »...)

- ▶ Différence entre système combinatoire et système séquentiel.
- ▶ Savoir lire et comprendre un diagramme des cas d'utilisation (« **ucd** »).
- ▶ Savoir lire et comprendre un diagramme de séquences (« **sd** »).