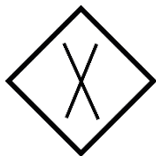
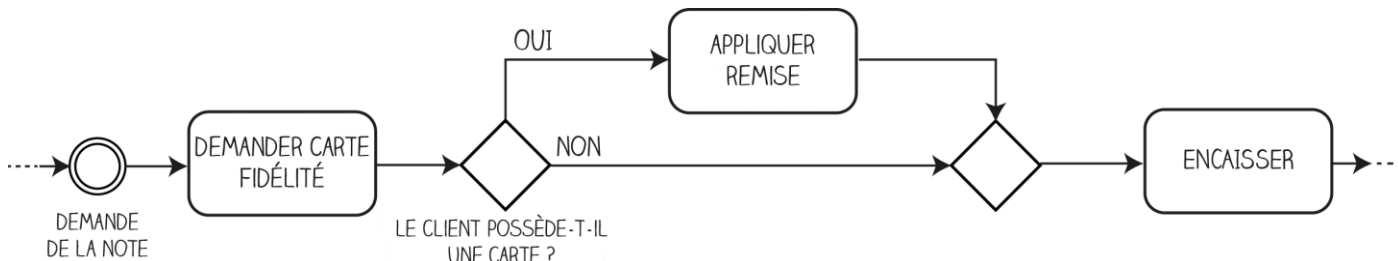


## C2.4 – Les passerelles (2)



Nous avons précédemment présenté les passerelles exclusives basées sur les données, celles permettant d'effectuer des alternatives dans un processus en fonction d'une condition. C'est la passerelle par défaut. Elle peut être représentée soit vide, soit avec une croix.

Rappelez-vous, dans notre exemple. Une fois que le client a demandé la note, on lui demande s'il possède une carte de fidélité. S'il possède une carte, une remise est alors appliquée sur le montant du repas.



On souhaite donc **diviser** le flux en deux, en fonction de : si le client possède ou non cette carte. Coté formalisme, on remarque plusieurs choses :

- Les passerelles sont reliées aux autres concepts par des flux de séquence.
- On indique la question correspondant à la condition voulu.
- On note sur les flux de séquence les critères des différents chemins possibles.

Dans la norme BPMN, il existe d'autres types de passerelles pour modéliser d'autres chemins de séquence que nous allons vous présenter.

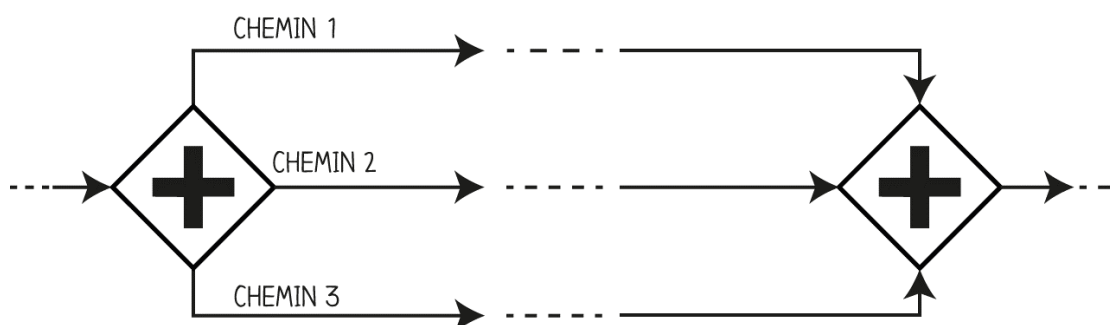
### 1. PASSERELLE PARALLELE



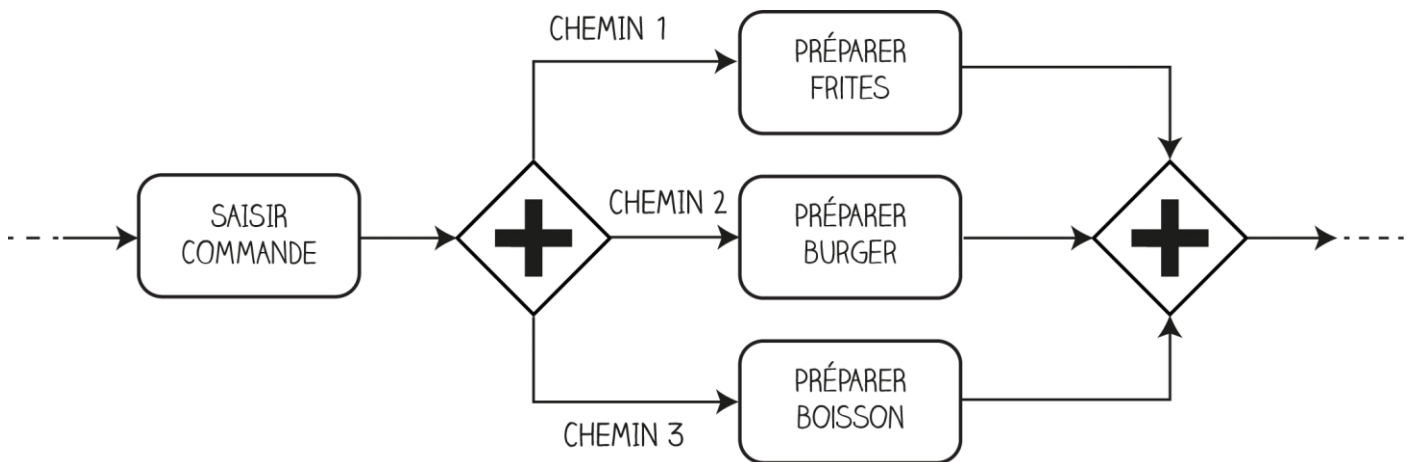
La passerelle parallèle permet comme son nom l'indique de modéliser des flux de séquence en parallèle. Le signe + permet de spécifier ce type de passerelle.

Contrairement à la passerelle exclusive qui permet de faire continuer le flux sur une et une seule branche, la passerelle parallèle fait continuer le flux sur l'ensemble des chemins.

Placé en **amont des tâches**, elle permet de synchroniser plusieurs chemins, c'est-à-dire que la séquence ne continuera qu'une fois l'ensemble des flux parallèles terminés.



Soyons concret : Si l'on reprend notre exemple du processus de service d'un client dans un restaurant, et que l'on se place dans un contexte de fastfood, une fois la commande saisie, les différents éléments composant le menu peuvent être préparés simultanément. On utilise donc une passerelle parallèle pour **diviser** le flux en différents chemins. Ensuite, pour servir le client, l'ensemble des composants du menu doivent être prêt. Une passerelle est alors positionnée afin de **synchroniser** les différents chemins, spécifiant ainsi que les trois activités doivent être terminées pour continuer le flux de séquence.

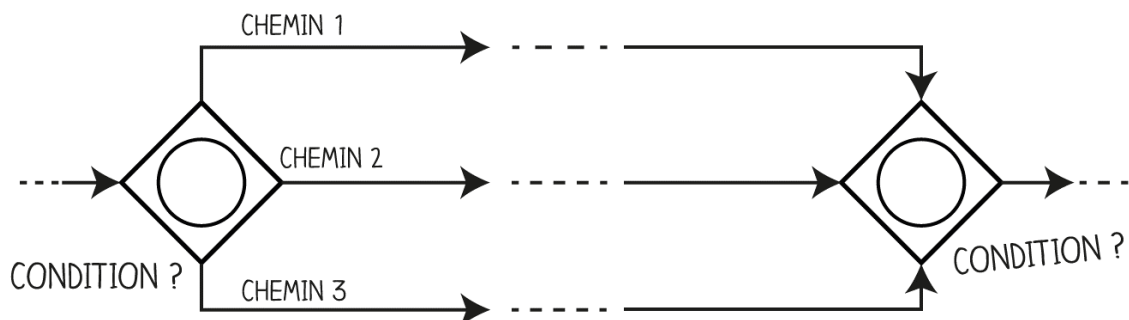


## 2. PASSERELLE INCLUSIVE



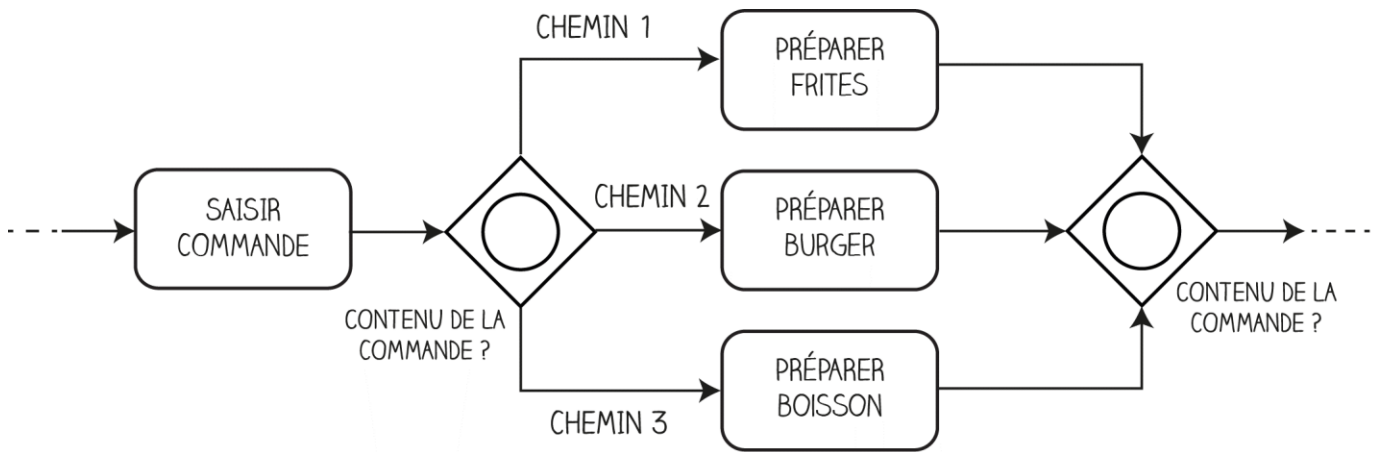
Voyons maintenant, la passerelle inclusive. Tout comme la passerelle exclusive, elle est conditionnelle, c'est-à-dire que le flux va continuer en fonction de certaines conditions. Mais, contrairement à la passerelle exclusive, il est possible de faire progresser les flux sur plusieurs chemins, si ceux-ci vérifient l'expression de la condition.

Ainsi, l'éclatement du flux pourra être fait sur une, certaines ou toutes les branches ; de même que pour la fusion des flux, il pourra concerner un, certains ou tous les chemins.



Reprenons notre exemple : dans la version où nous utilisons une passerelle parallèle, cela impliquait que le menu devait être composé de ces trois éléments, car on continuait vers les 3 activités réalisées simultanément.

Cependant, il peut arriver qu'une personne ne commande qu'une boisson, ou qu'un burger, juste une boisson et un burger, ou le menu complet. Il faut s'adapter à ces différents cas de figure. Aussi, une passerelle inclusive peut être utilisée. En fonction du contenu de la commande, notre condition, le flux de séquence continuera sur un, plusieurs ou l'ensemble des chemins ; et il en va de même pour la réunion du flux.



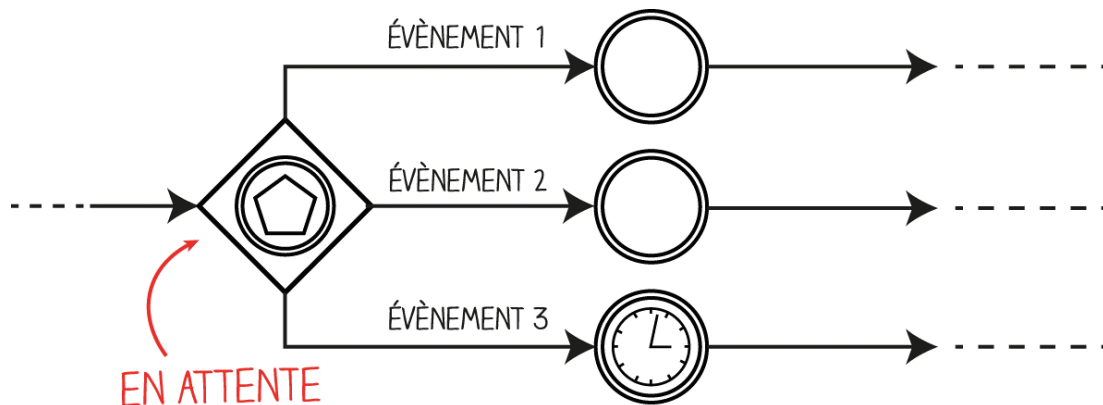
### 3. PASSERELLE EVENEMENTIELLE EXCLUSIVE



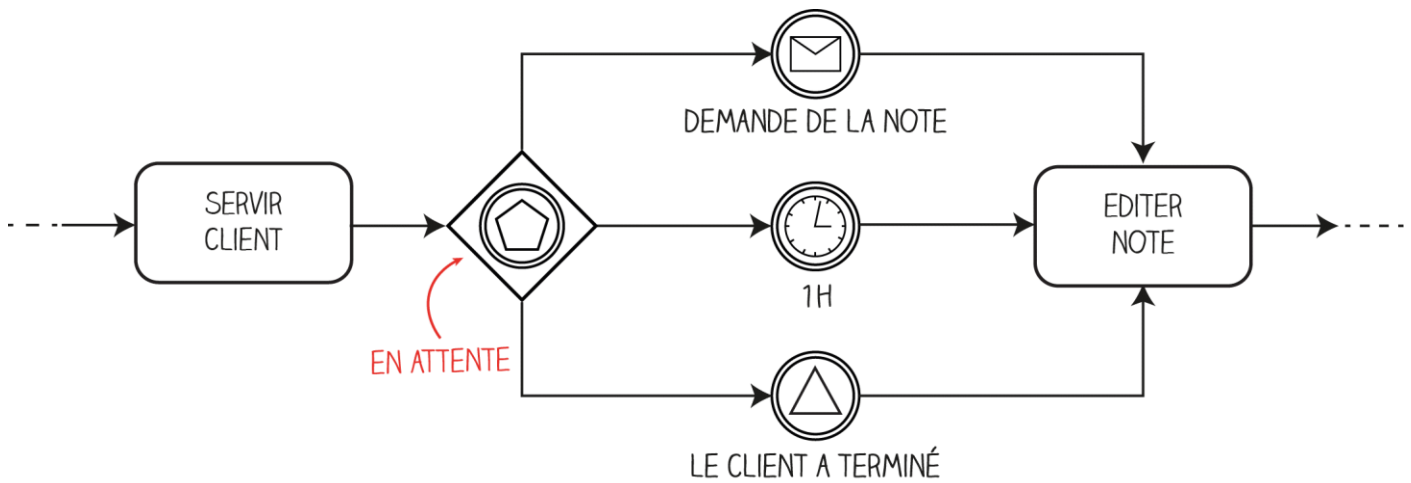
Nous avons présenté précédemment la passerelle exclusive dont les conditions se basent sur des informations. Il existe le même type de passerelle pour les conditions se basant sur des événements. La passerelle événementielle exclusive est notée par un pentagone dans un cercle.

Cette passerelle s'utilise pour mettre le processus en attente d'événements. L'arrivée d'un événement déclenchera un des chemins du processus.

Tout comme la passerelle exclusive basée sur les données, il ne faut pas que le processus puisse se retrouver bloqué, dans notre cas se serait si aucun événement ne venait à se produire. Aussi, une des bonnes pratiques veut que l'on rajoute un événement de type timer pour spécifier les activités dans le cas où les événements attendus ne se produiraient pas dans un délai acceptable.



Par exemple, dans un restaurant classique, une fois le client servi, nous pouvons attendre que le client nous demande la note (ce qui correspond à notre événement de type réception de message) ou la lui apporter automatiquement 1h après avoir été servi (correspondant à notre timer) ou encore une fois que l'on a observé que le client a terminé (avec notre événement de type signal).

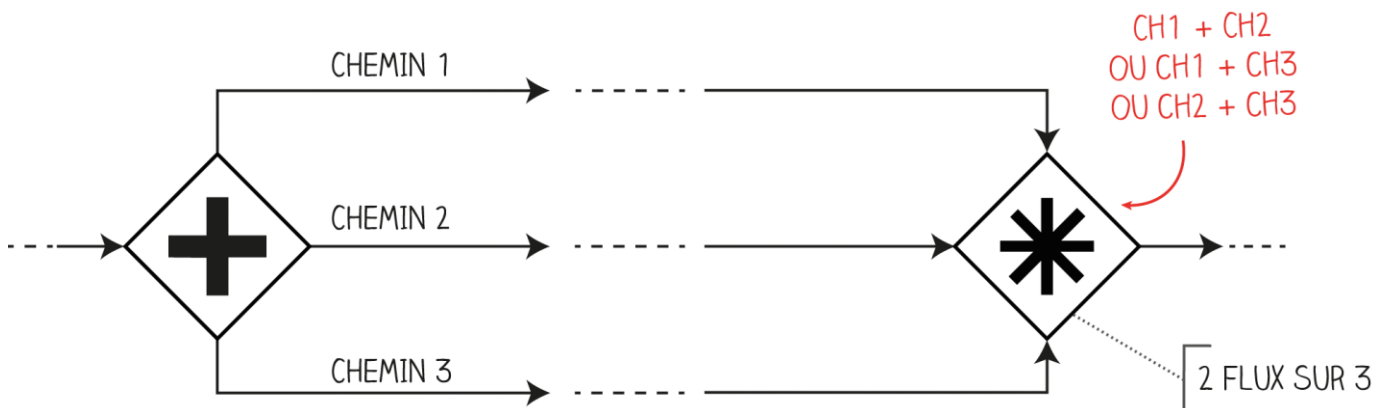


#### 4. PASSERELLE COMPLEXE



La passerelle complexe est utilisée lorsque le comportement du flux ne peut pas être exprimé par un autre type de passerelle. On la représente par un **astérisque**. Elle permet de modéliser une synchronisation complexe, donc basée sur une condition ni exclusive, ni inclusive, ni parallèle, qu'il faut préciser en commentaire.

Prenons comme exemple générique une passerelle parallèle générant trois chemins de séquence simultanés ; grâce à une passerelle complexe, nous pourrions modéliser que le flux continuera lorsque seulement deux des trois flux seront terminés.



Grâce à toutes ces passerelles, vous pouvez maintenant modéliser différents comportements de flux de séquence de vos processus.