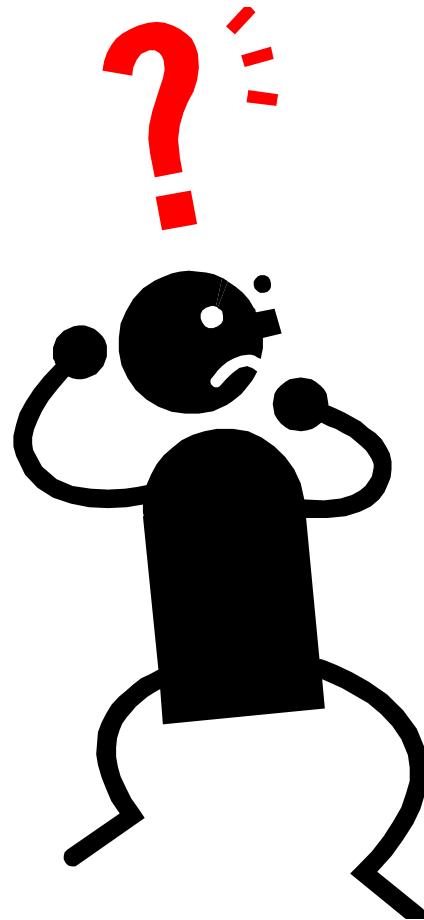


[Introduction à la gestion des processus métier]

[Business Process Model and Notation (BPMN)]

[ Qu'est ce qu'.... ]

Qu'est ce  
qu'un  
processus ?



# Processus

- Du latin procedere : « aller en avant »
- *Définition (Larousse): « Un **enchaînement ordonné de faits ou phénomènes, répondant à un certain schéma et aboutissant à quelque chose.** »*
- Définition de référence : (International Organization of Standardization (ISO) 9000:2000 : « *Ensemble d'activités corrélées ou interactives qui transforment des éléments d'entrée en éléments de sortie* »

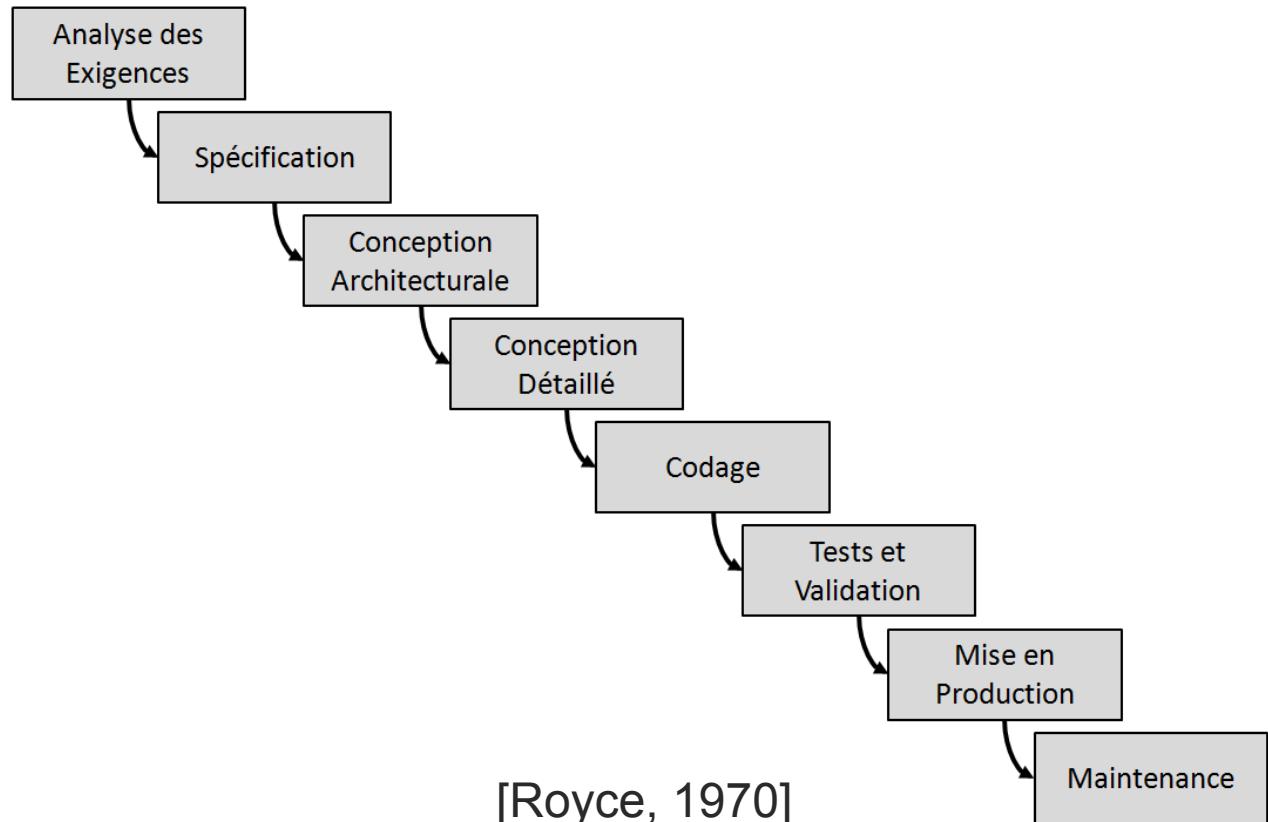


International  
Organization for  
Standardization

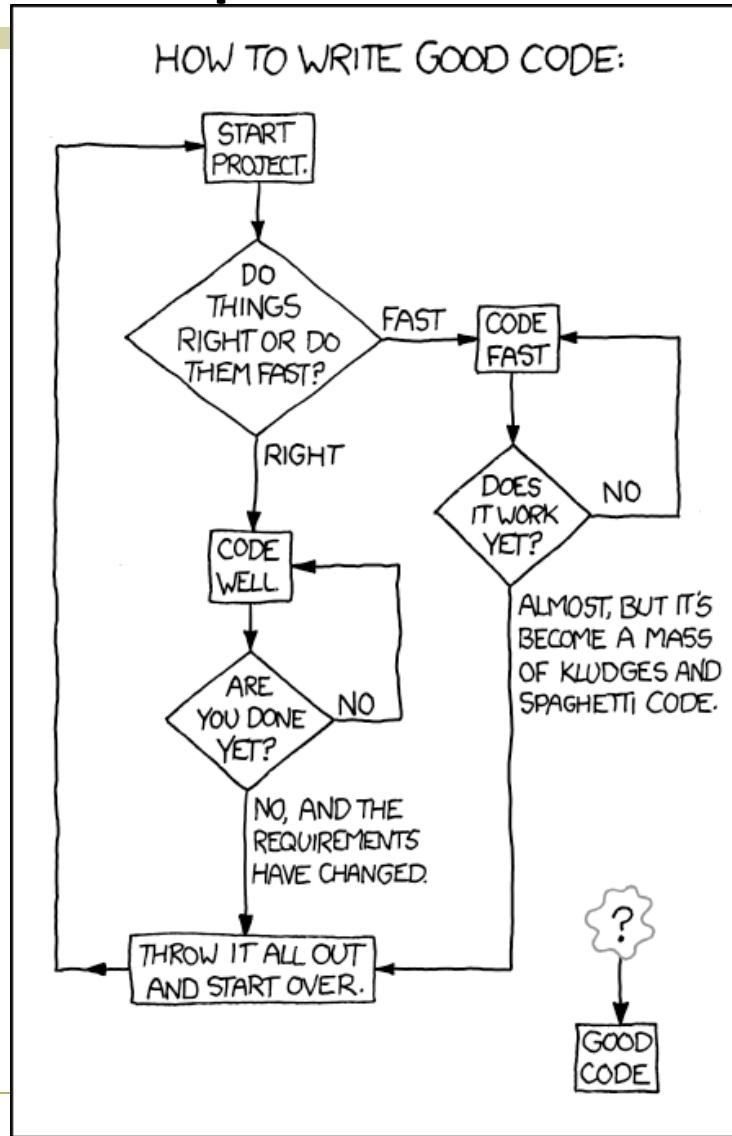
- ▶ Organisation fondée en 1947 qui a publié plus de 19500 normes internationales dans presque tous les domaines technologiques et économiques (sécurité des aliments et des ordinateurs, agriculture, santé, ...) <http://www.iso.org>

# Exemple de processus (1)

## ▶ Processus de Développement Logiciel (en Cascade)



# Exemple de Processus (2)



[http://imgs.xkcd.com/comics/good\\_code.png](http://imgs.xkcd.com/comics/good_code.png)

# Processus Métier

## ■ Merise

Un processus est la réaction de l'entreprise face à un **événement**

## ■ Hammer & Champy

Un processus est un ensemble finalisé d' **activités** orienté vers la production d' **un résultat qui a une valeur**, une importance, pour le **client**.

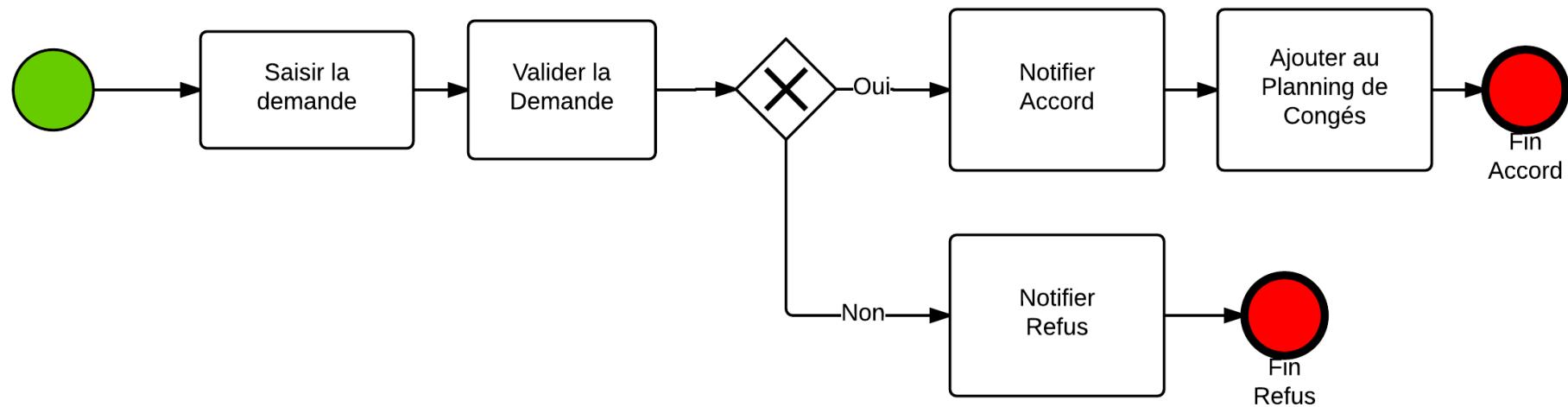
## ■ Morley & al

Processus = **organisation d'activités**

- ✓ Ayant un **but** commun (*par ex. répondre aux besoins d'un acteur « externe » par ex. le « client »*)
- ✓ Effectuées par des **acteurs** (*ici acteur au sens travailleur, acteur interne*)
- ✓ Mettant en jeu des **informations**
- ✓ Répondant à un type **d'événement**

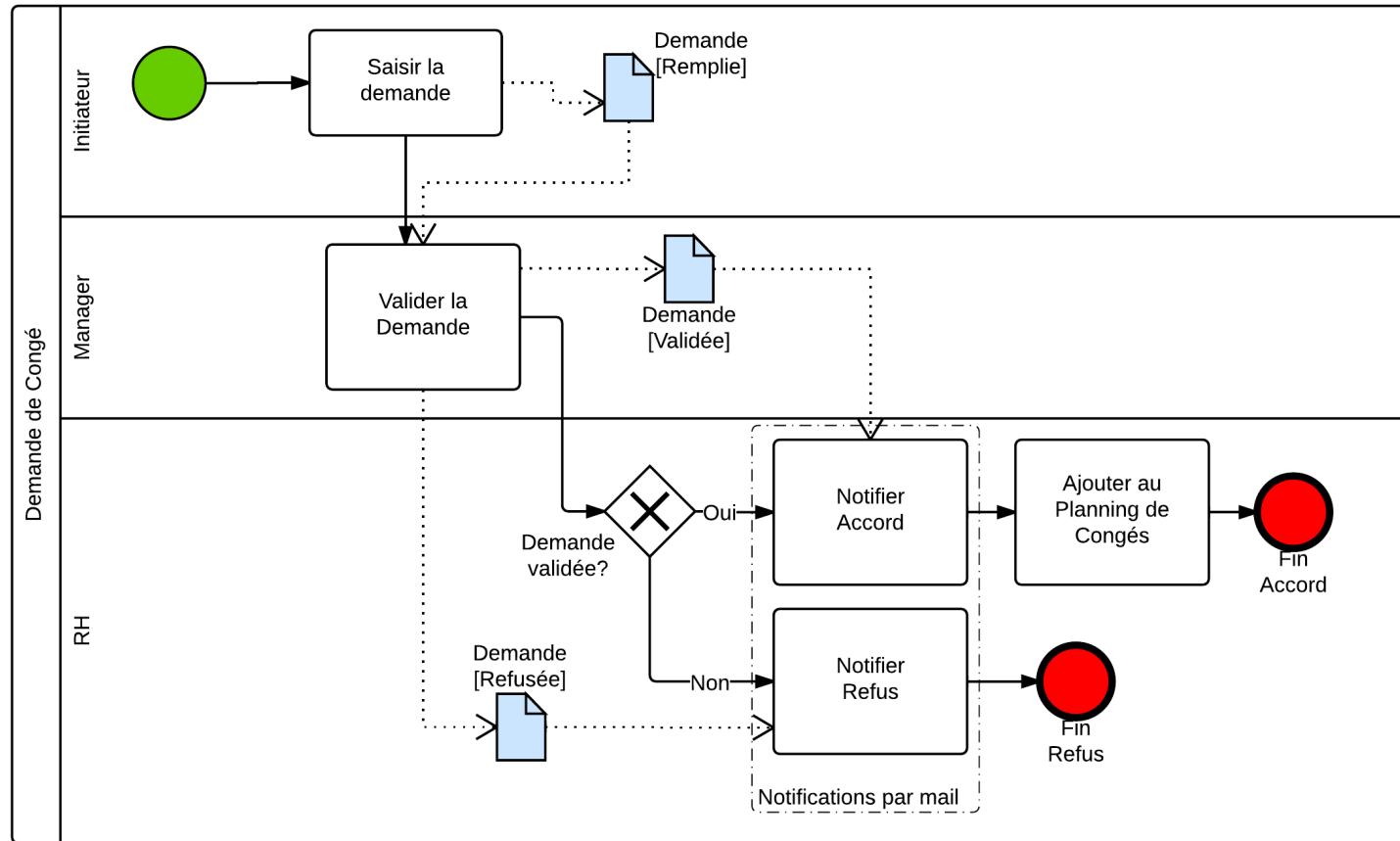
# [ Exemple de Processus Métier (1) ]

## Processus de demande de congé (simplifié)

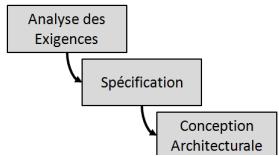


# Exemple de Processus Métier (2)

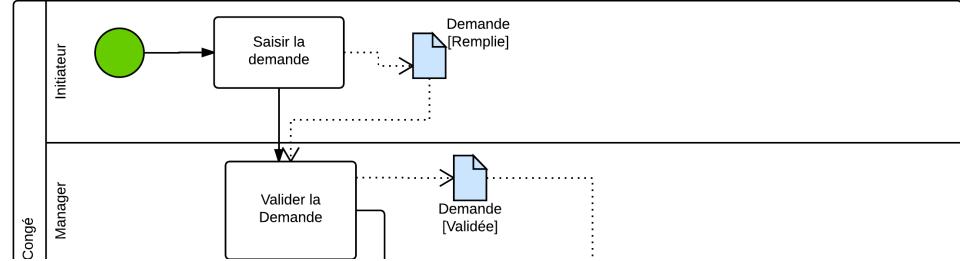
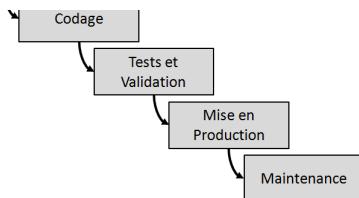
## Processus de demande de congé (plus détaillé)



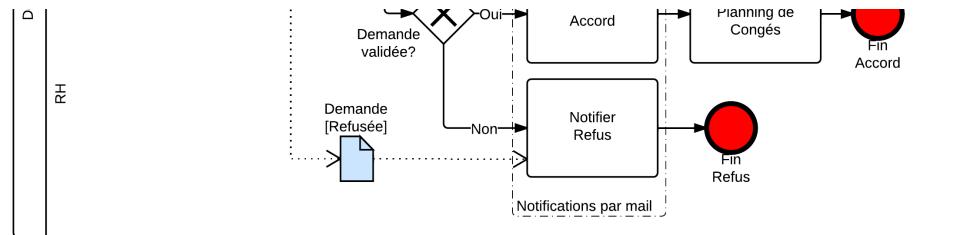
# Les modèles



Ceci n'est pas un processus logiciel



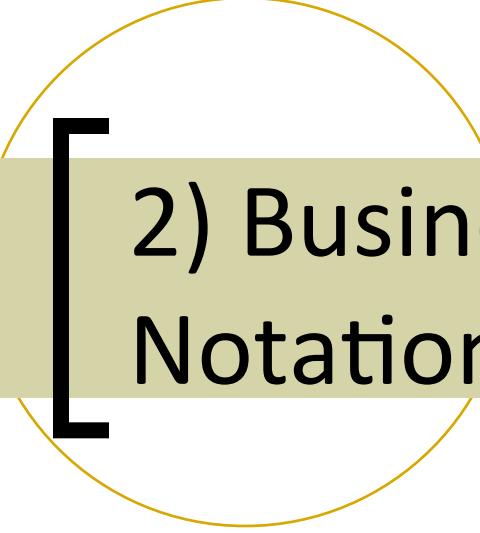
Ceci n'est pas un processus de demande de congés



# Modèle : définitions

- Un modèle est une représentation abstraite de la réalité.  
Exemple : une carte routière, un plan d'architecture, un modèle d'analyse, etc
- *Pour un opérateur O, un objet M est un modèle d'un objet A dans la mesure où O peut utiliser M pour répondre aux questions qui l'intéressent au sujet de A (Minsky)*
- Un modèle doit :
  - ✓ correctement représenter la pensée du modélisateur,
  - ✓ répondre aux problèmes de l'opérateur
  - ✓ permettre de communiquer sans ambiguïté

⇒ être décrit à l'aide d'une **notation** bien définie et connue, standardisée....



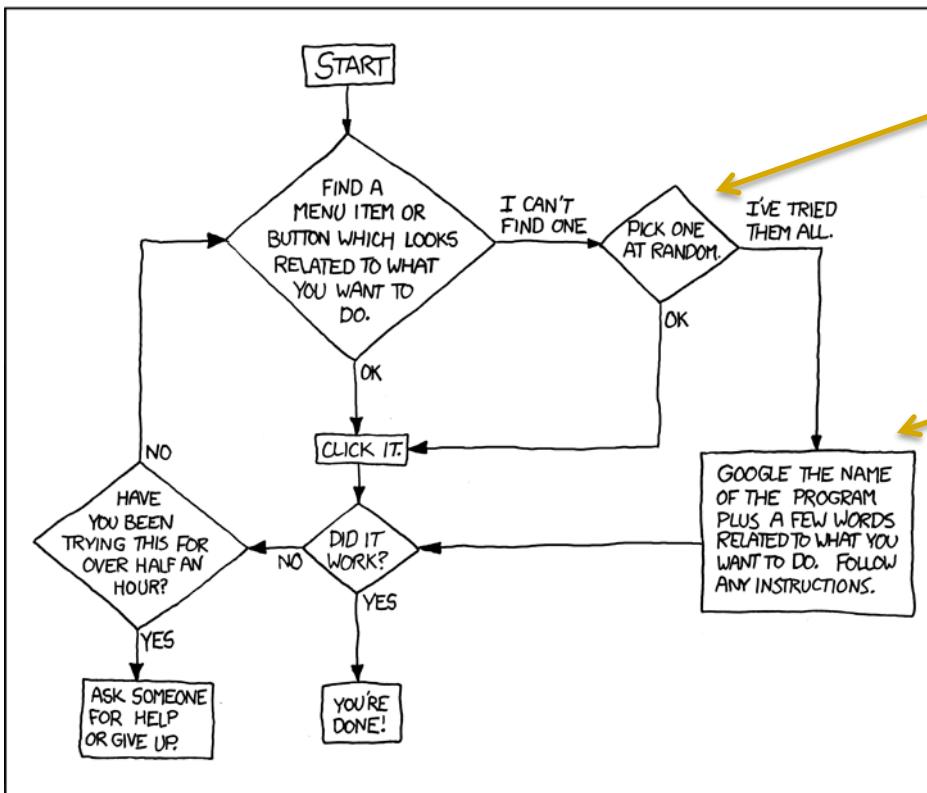
## **2) Business Process Model and Notation (BPMN)**



# L'origine de BPMN -> Flowcharts

DEAR VARIOUS PARENTS, GRANDPARENTS, CO-WORKERS,  
AND OTHER "NOT COMPUTER PEOPLE."

WE DON'T MAGICALLY KNOW HOW TO DO EVERYTHING IN EVERY  
PROGRAM. WHEN WE HELP YOU, WE'RE USUALLY JUST DOING THIS:



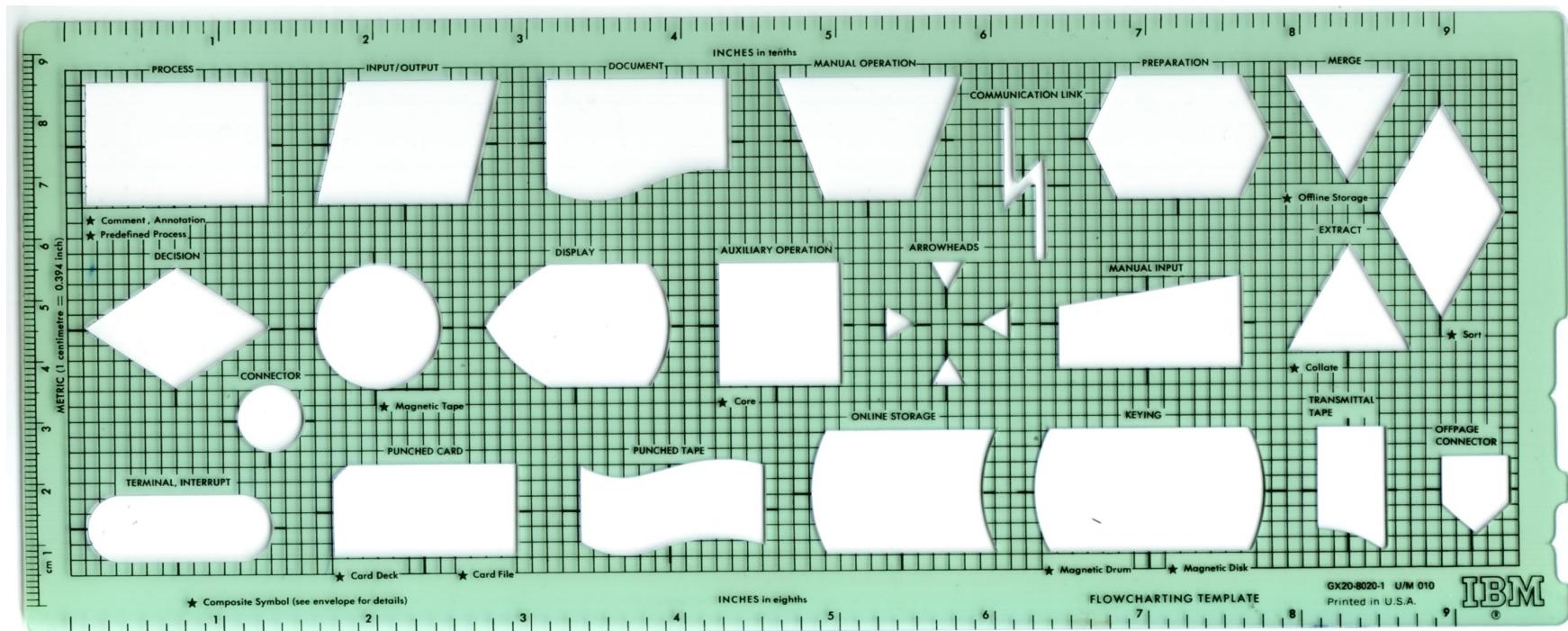
Un organigramme de programmation,  
algorigramme, logigramme ...

branchement

traitement

PLEASE PRINT THIS FLOWCHART OUT AND TAPE IT NEAR YOUR SCREEN.  
CONGRATULATIONS; YOU'RE NOW THE LOCAL COMPUTER EXPERT!

# « Outil » de Modélisation des Flowcharts



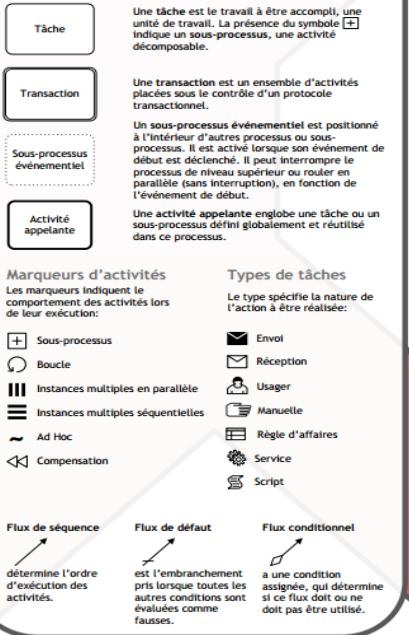
<http://upload.wikimedia.org/wikipedia/en/4/4d/Flowchart-template.jpg>

Notation BPMN pour les processus métiers :  
standardisée, bien formalisée, complète  
.....mais complexe

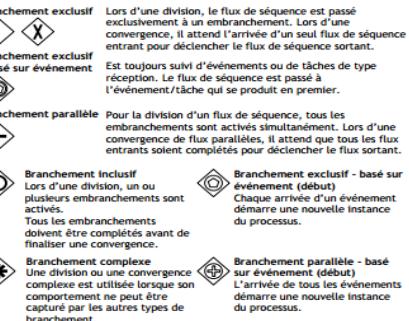
BPMN 2.0 - Business Process Model and Notation

<http://bpmb.de/poster>

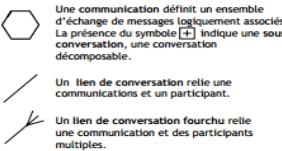
## Activités



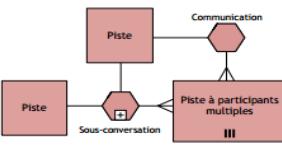
### ranchements



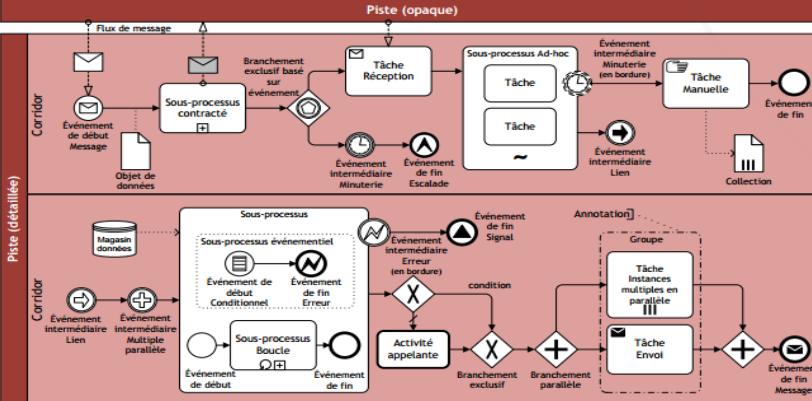
## Conversations



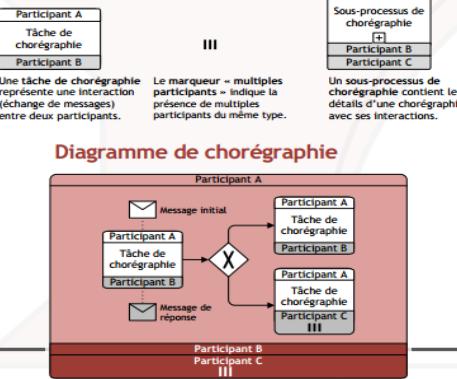
## Diagramme de conversation



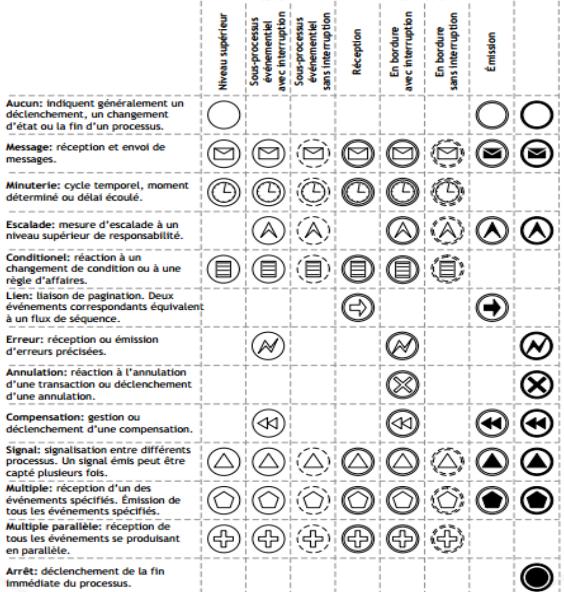
## Diagramme de collaboration



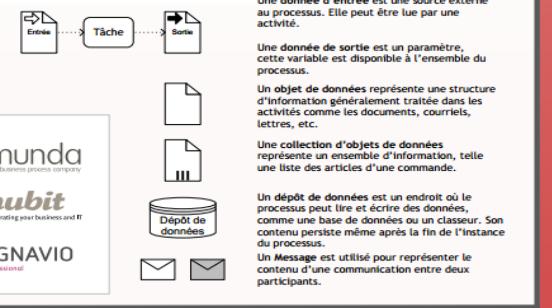
## Chorégraphies



## Événements

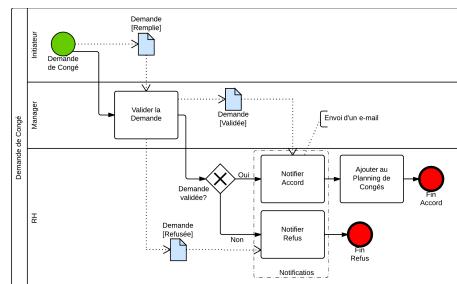
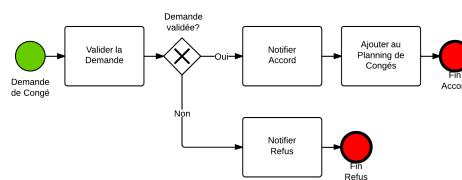


## Données



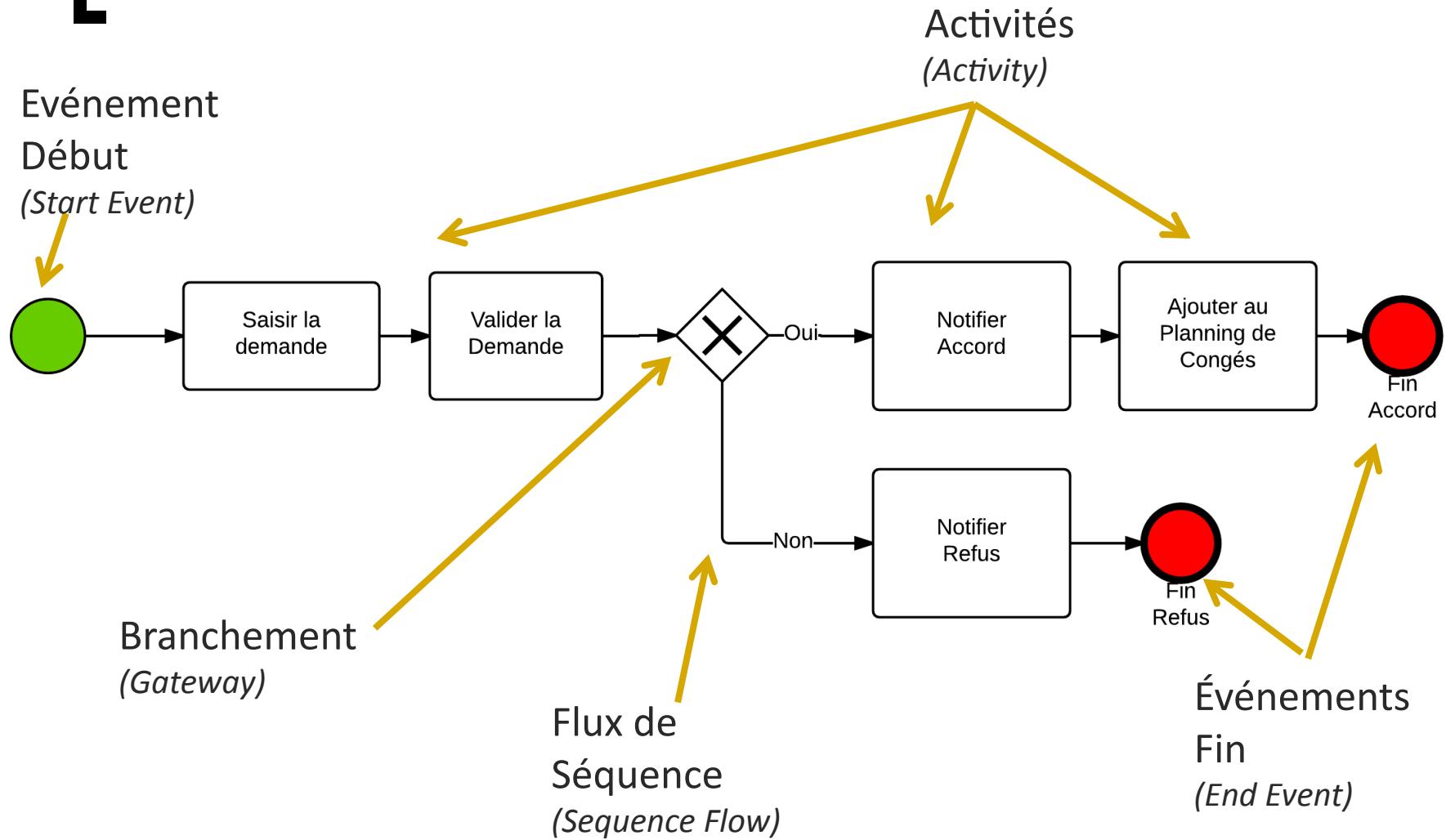
# Un des premiers objectifs de BPMN

- **Faciliter la communication** entre les différents acteurs engagés dans le développement et la maintenance des processus de l'entreprise

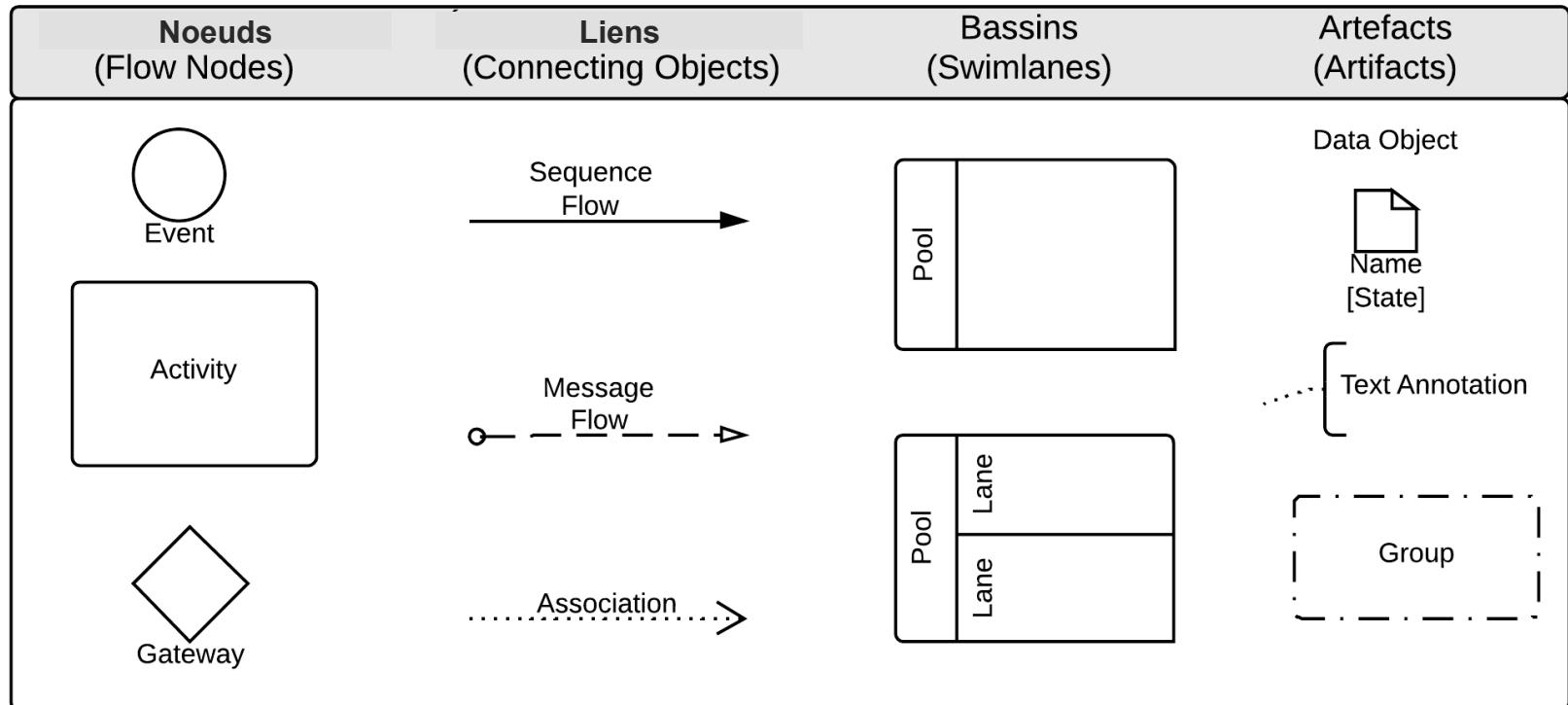


```
<model:process id="_zWy" name="Demande de Congé">
<model:ioSpecification id="_33H">...</model:ioSpecification>
<model:dataObject id="DataObject_33H" name="nom" isCollection="false" itemSubjectRef="_mxL"/>
<model:dataObject id="DataObject_33M" name="prenom" isCollection="false" itemSubjectRef="_oGP"/>
...
<model:userTask id="_z7Z" name="Valider la Demande"/>
<model:startEvent id="_0SZ" name="Start1"/>
<model:serviceTask id="_MFV" name="Notifier Accord"/>
...
</model:process>
<di:BPMNDiagram name="MyDiagram">...</di:BPMNDiagram>
```

# BPMN 2.0 – Exemple

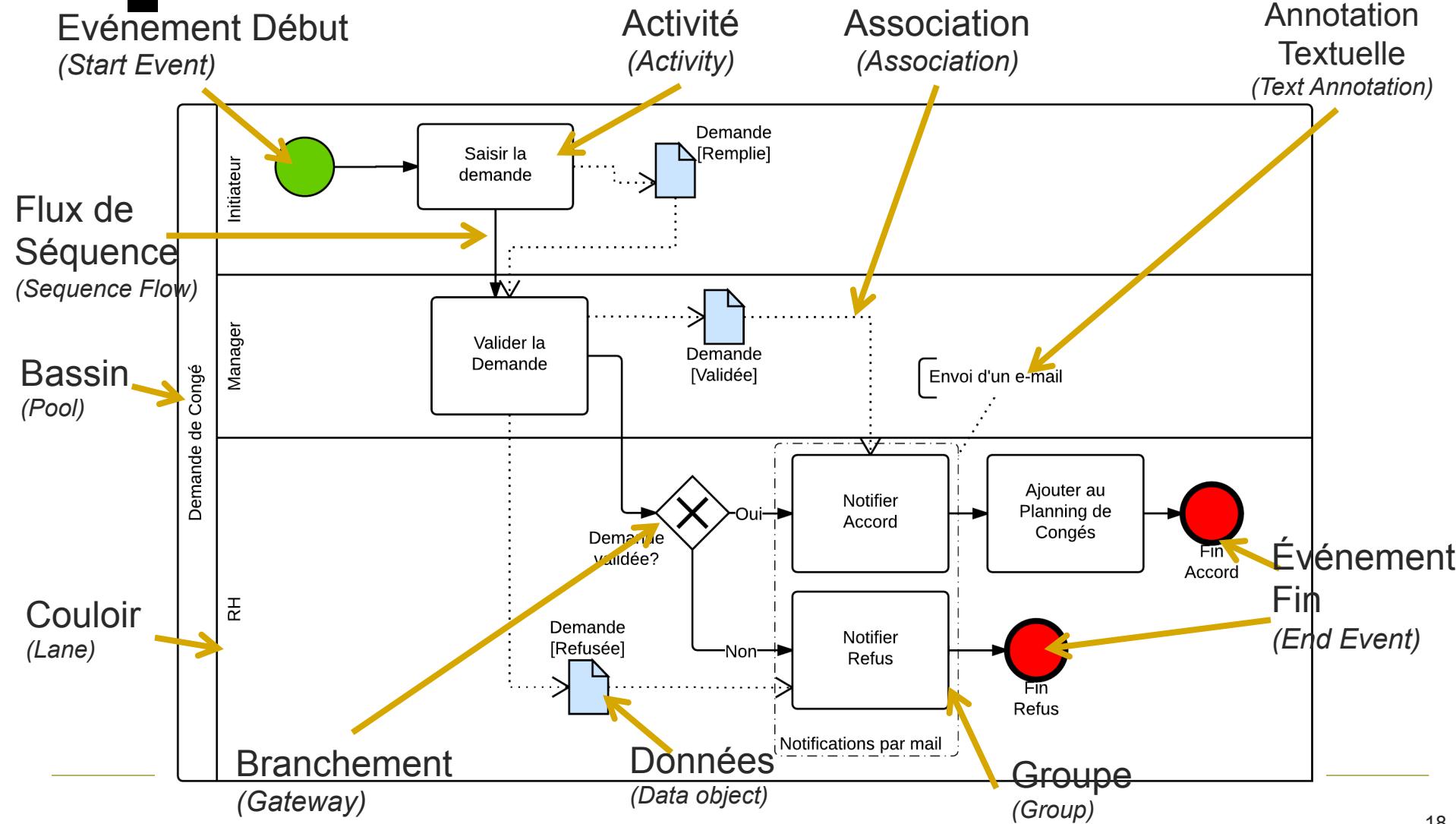


# Eléments principaux de BPMN (I)

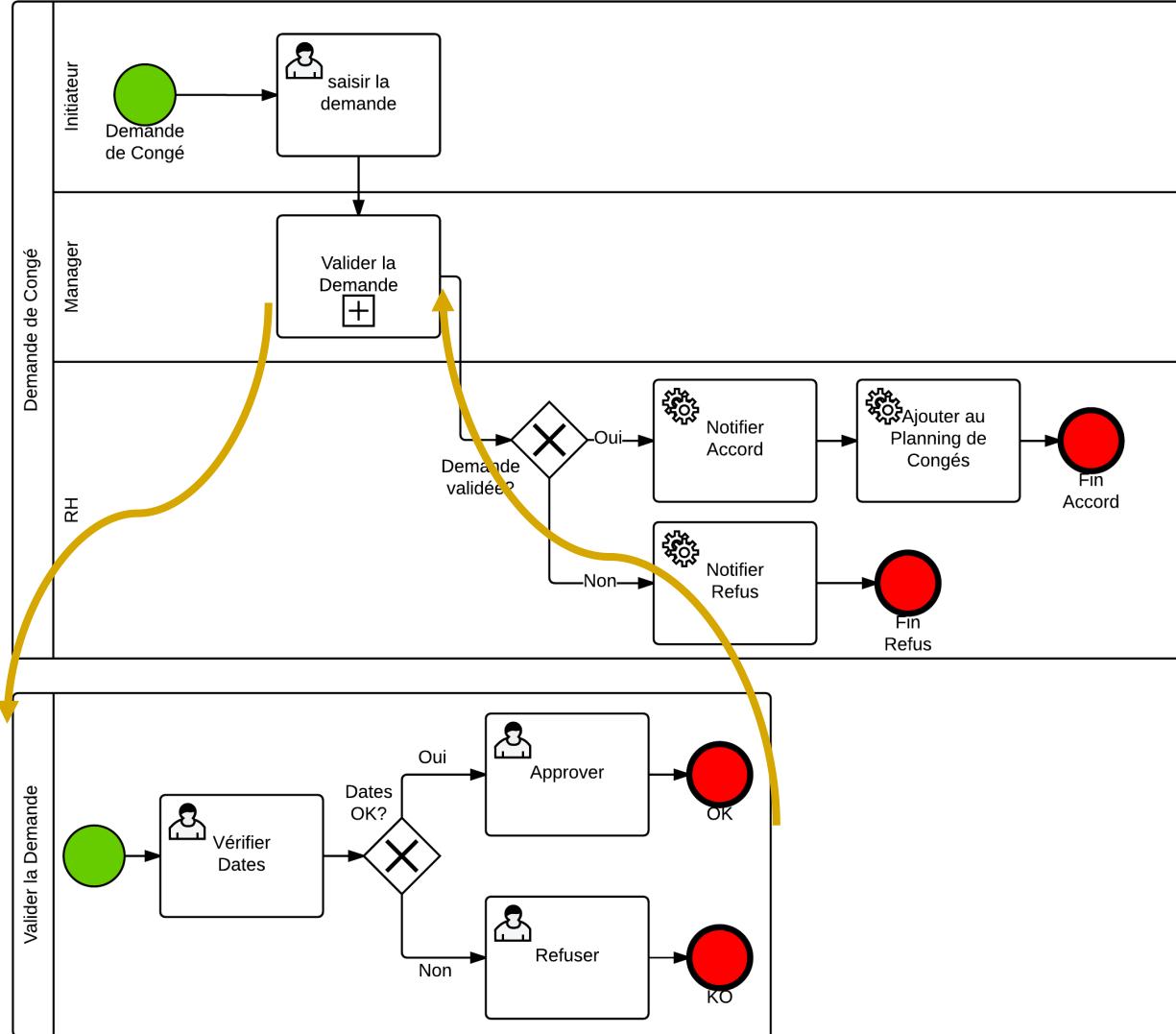


Source: OMG (2011). *Business Process Model and Notation* (BPMN) <http://www.omg.org/spec/BPMN/2.0>

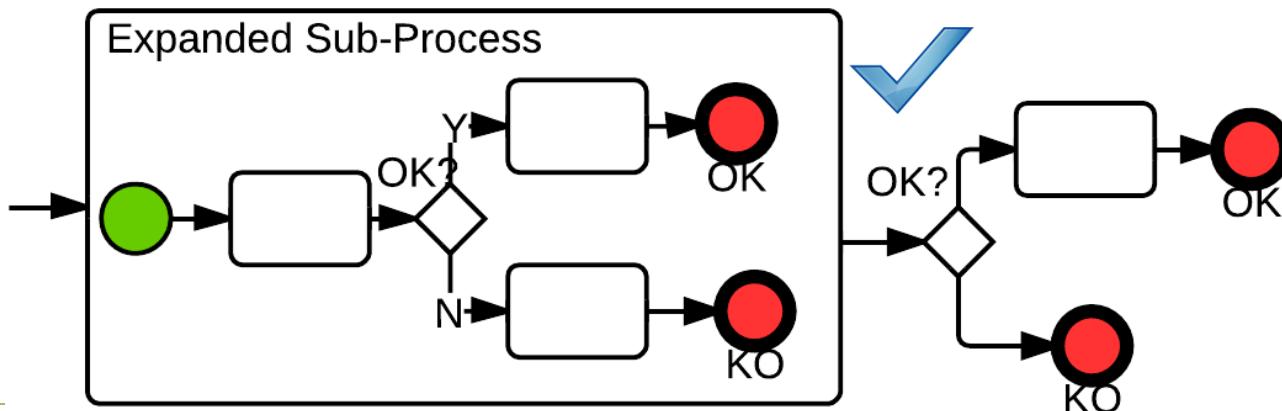
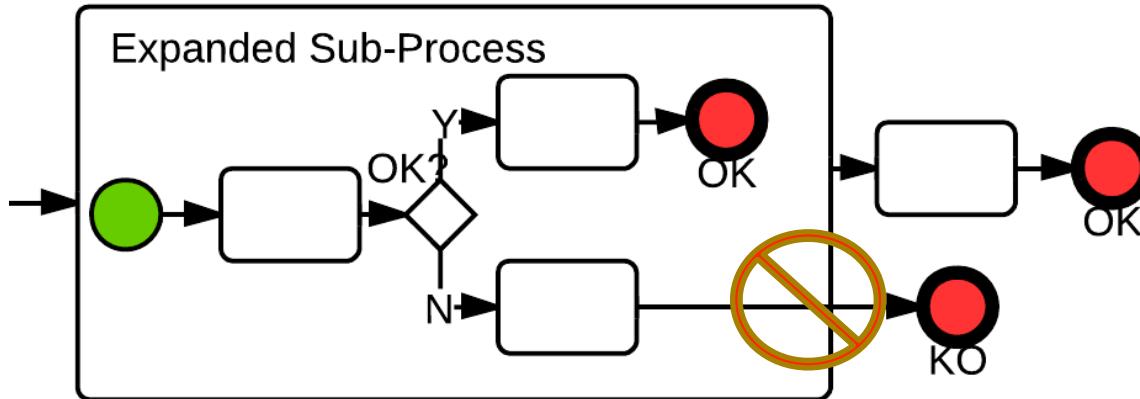
# Eléments principaux de BPMN (II)



# Sous-Processus



# Règles d'Utilisation Sous-Processus



# Activités

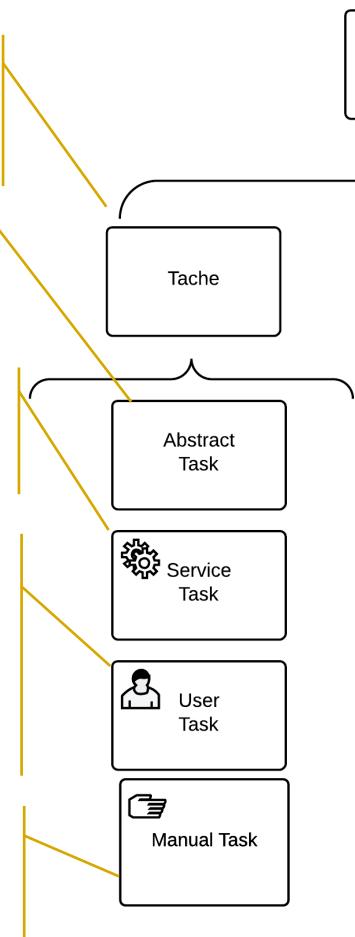
Une **tâche** est une activité atomique.

*Abstract Task* désigne une activité sans type.

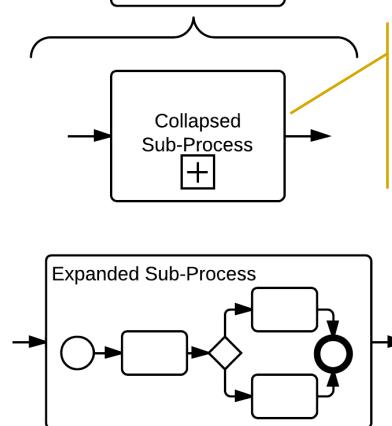
*Service Task* désigne une activité complètement automatisée.

*User Task* désigne une activité informatisée nécessitant une interaction homme-machine

*Manual Task* désigne une activité manuelle



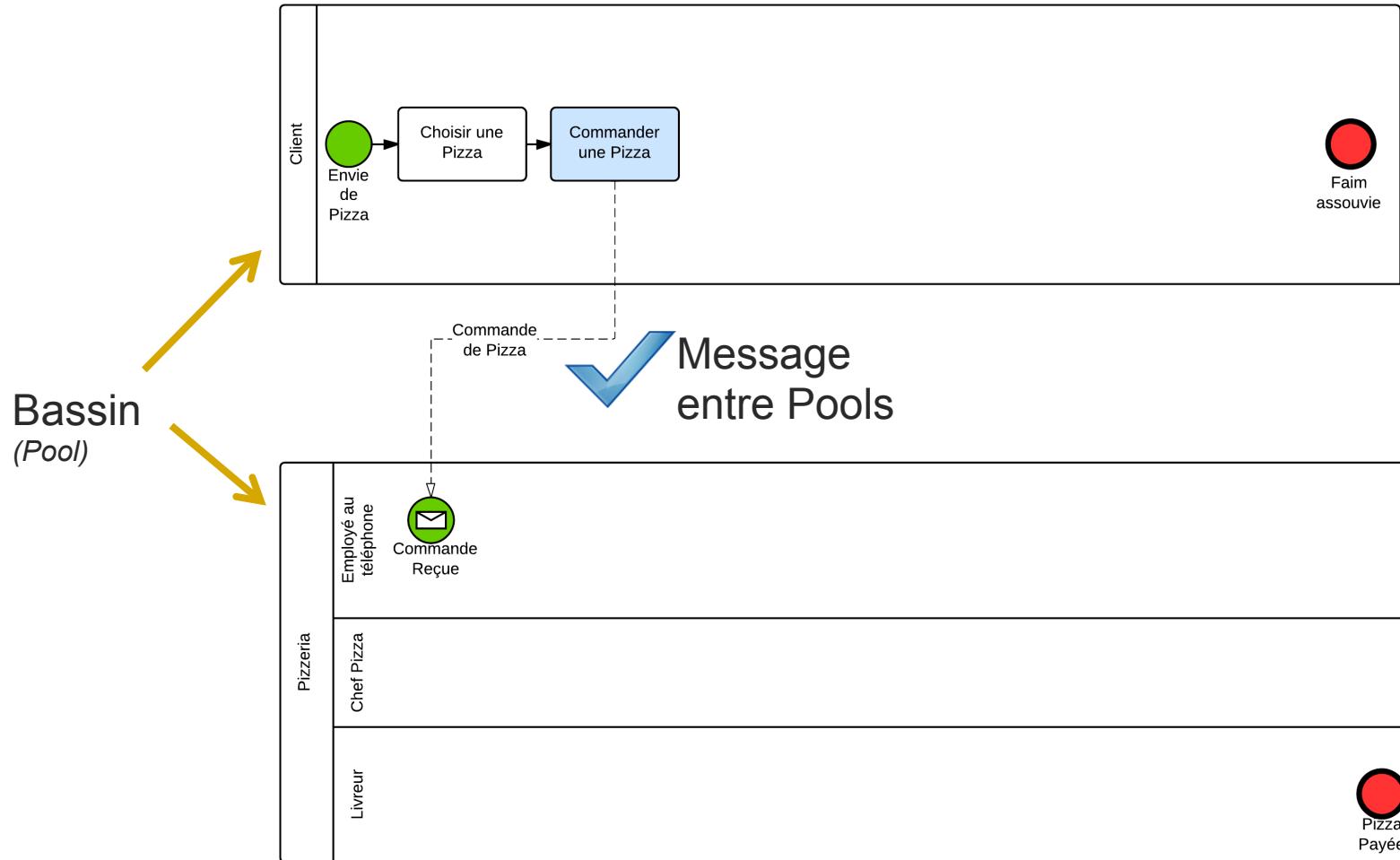
Un **sous-processus** est une activité complexe et donc décomposable



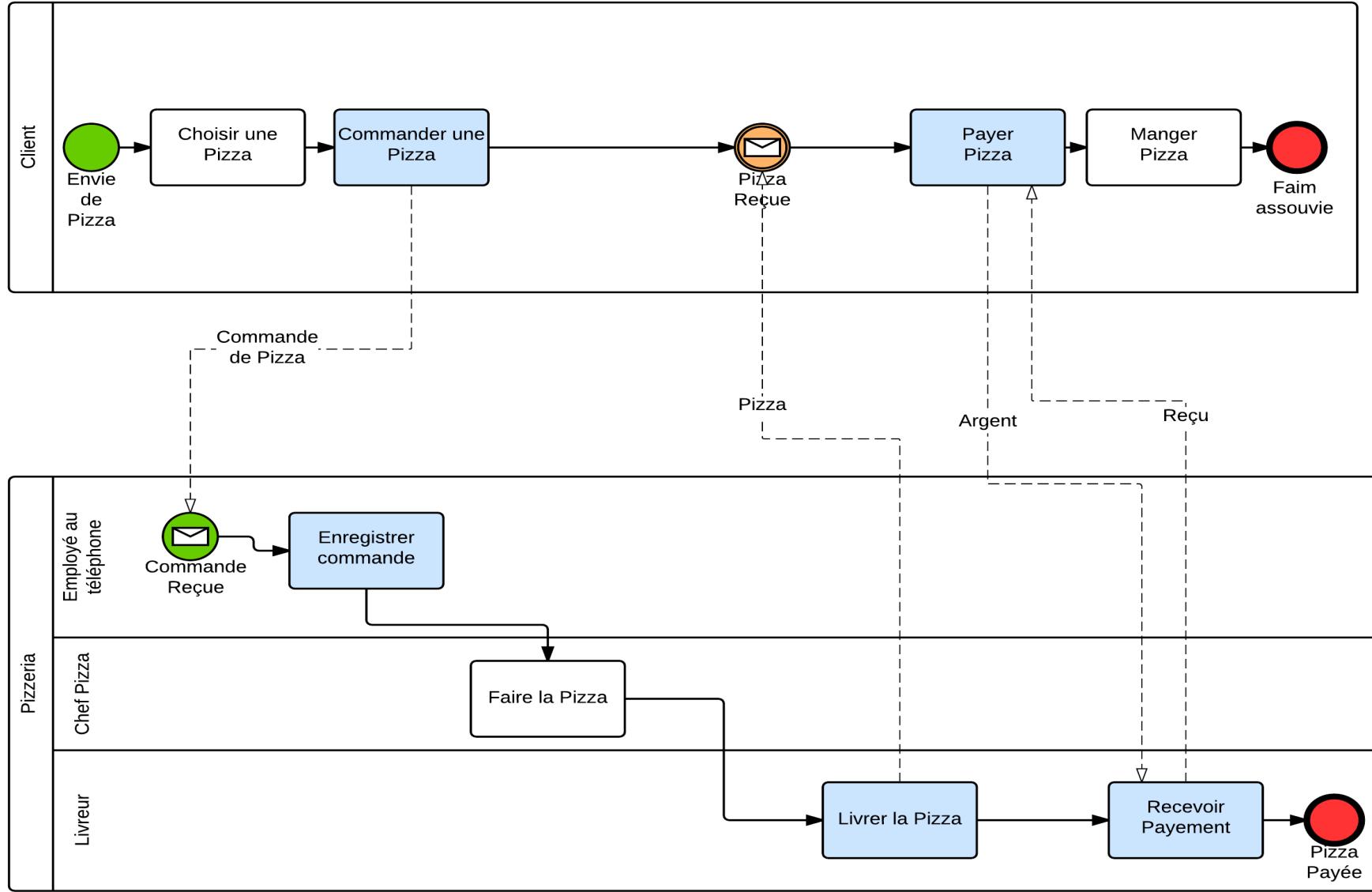
Dans un *Collapsed Sub-Process* les activités à l'intérieur ne sont pas visibles.

Dans un *Expanded Sub-Process* les activités à l'intérieur sont visibles.

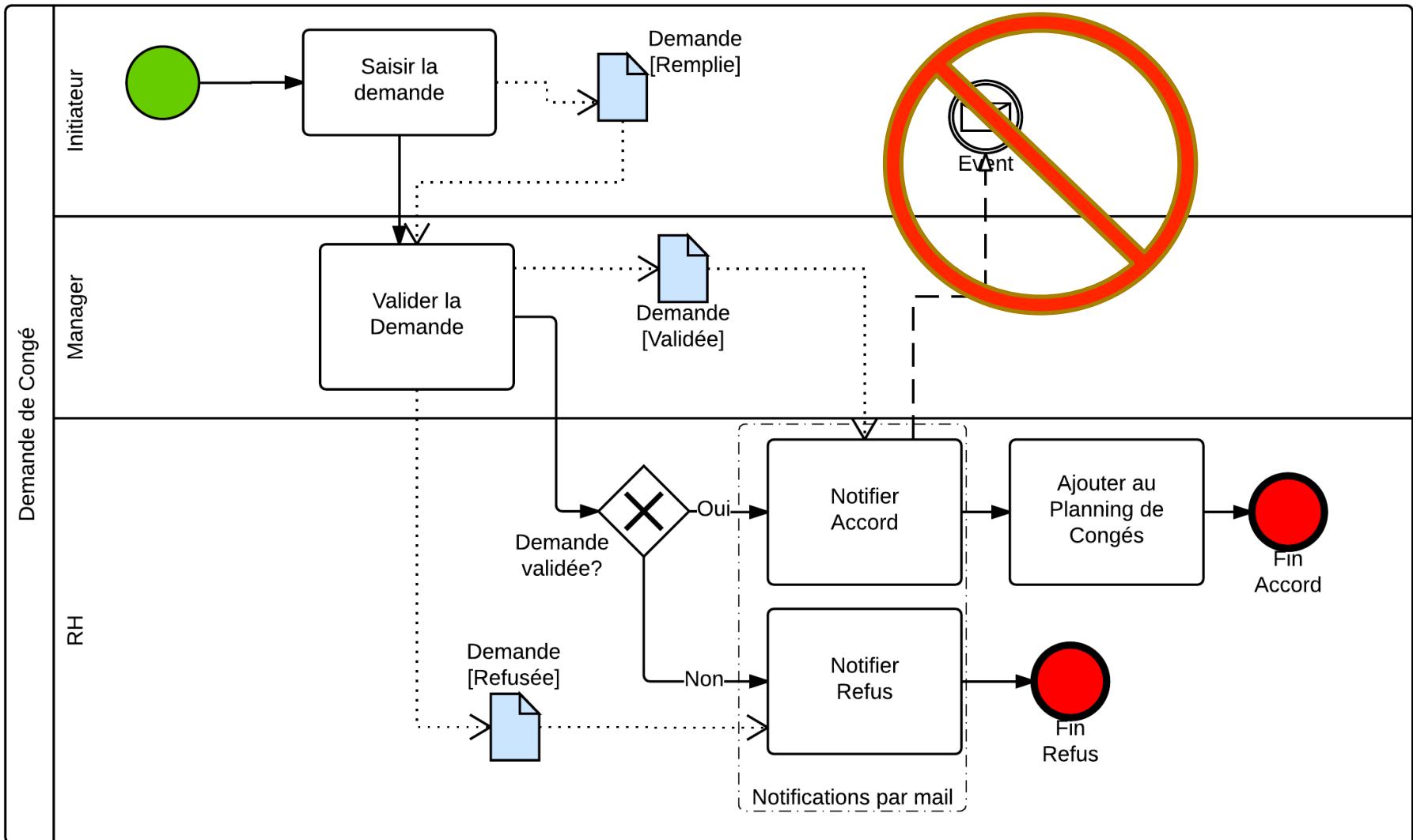
# Collaboration – plusieurs pools



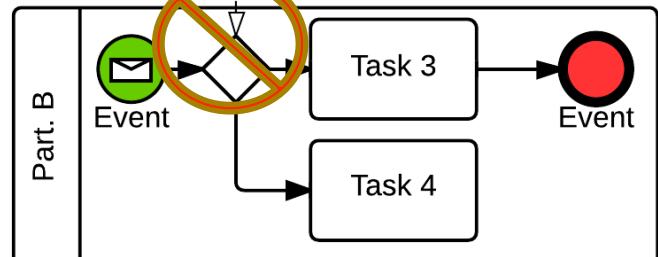
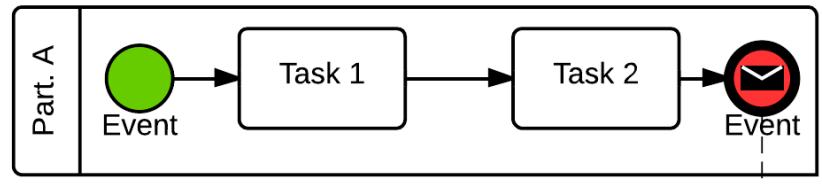
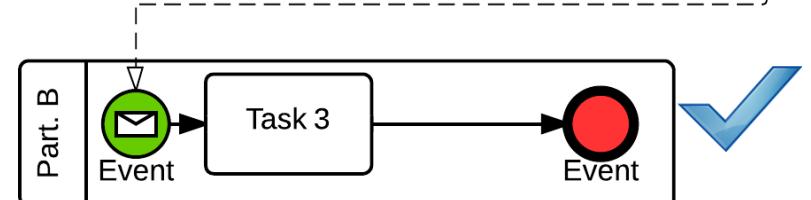
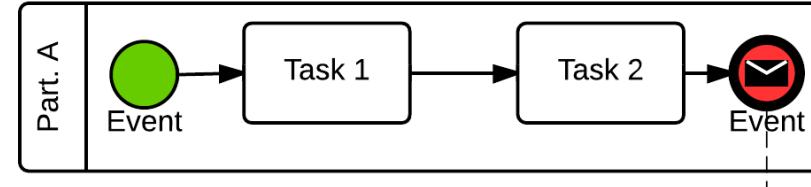
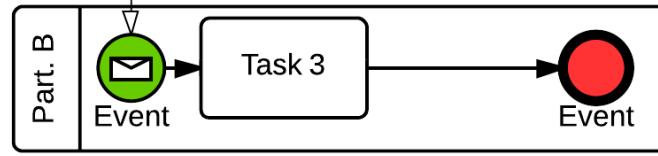
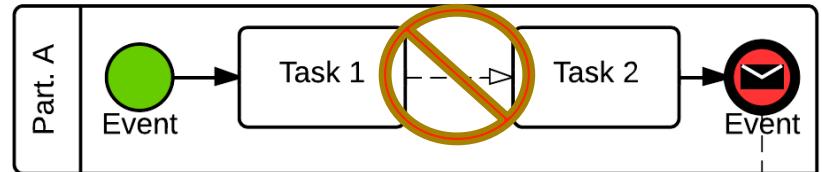
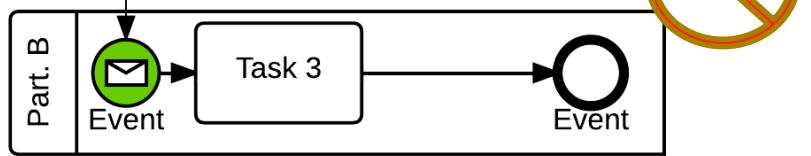
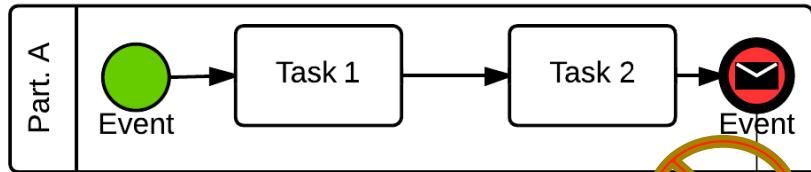
# Collaboration – plusieurs pools



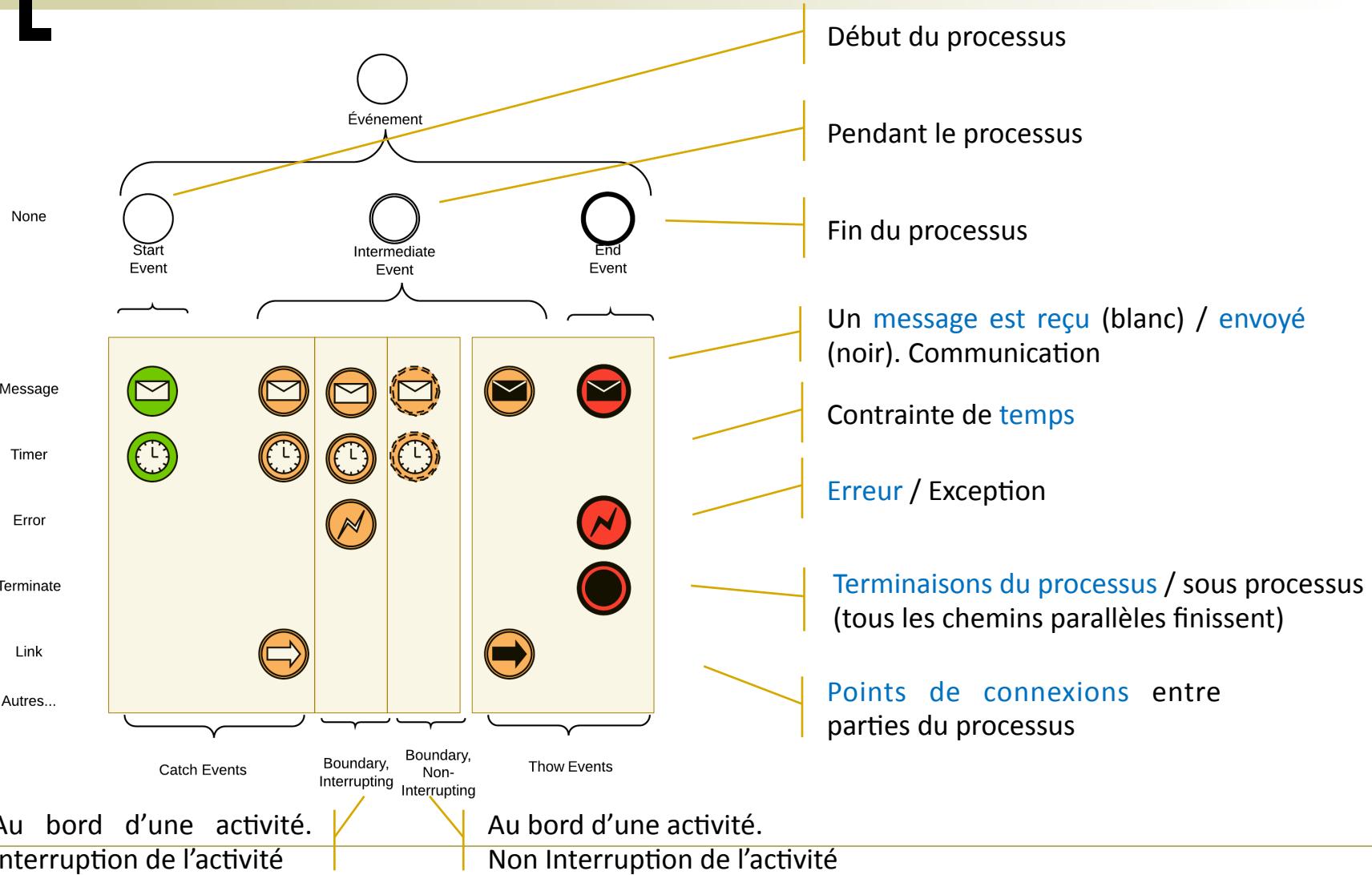
# Erreur classique Message Flow à l'intérieur d'un Bassin (Pool)



# Règles d'Utilisation des liens



# Evénements

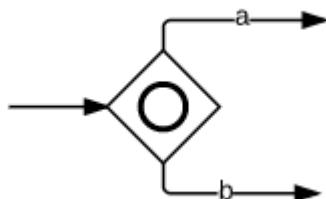


# Branchements (Portes Logiques)

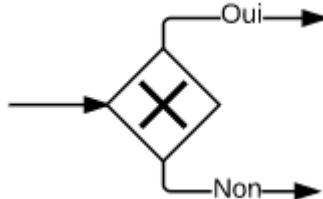


Un **Gateway** représente un point de contrôle sur le flux du processus

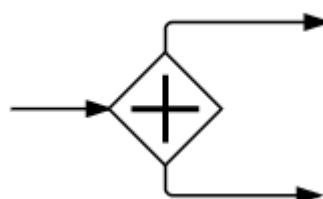
Inclusive



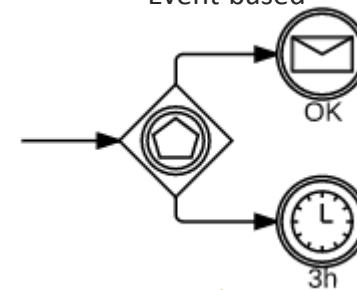
Exclusive



Parallel



Event-based



Autres ...

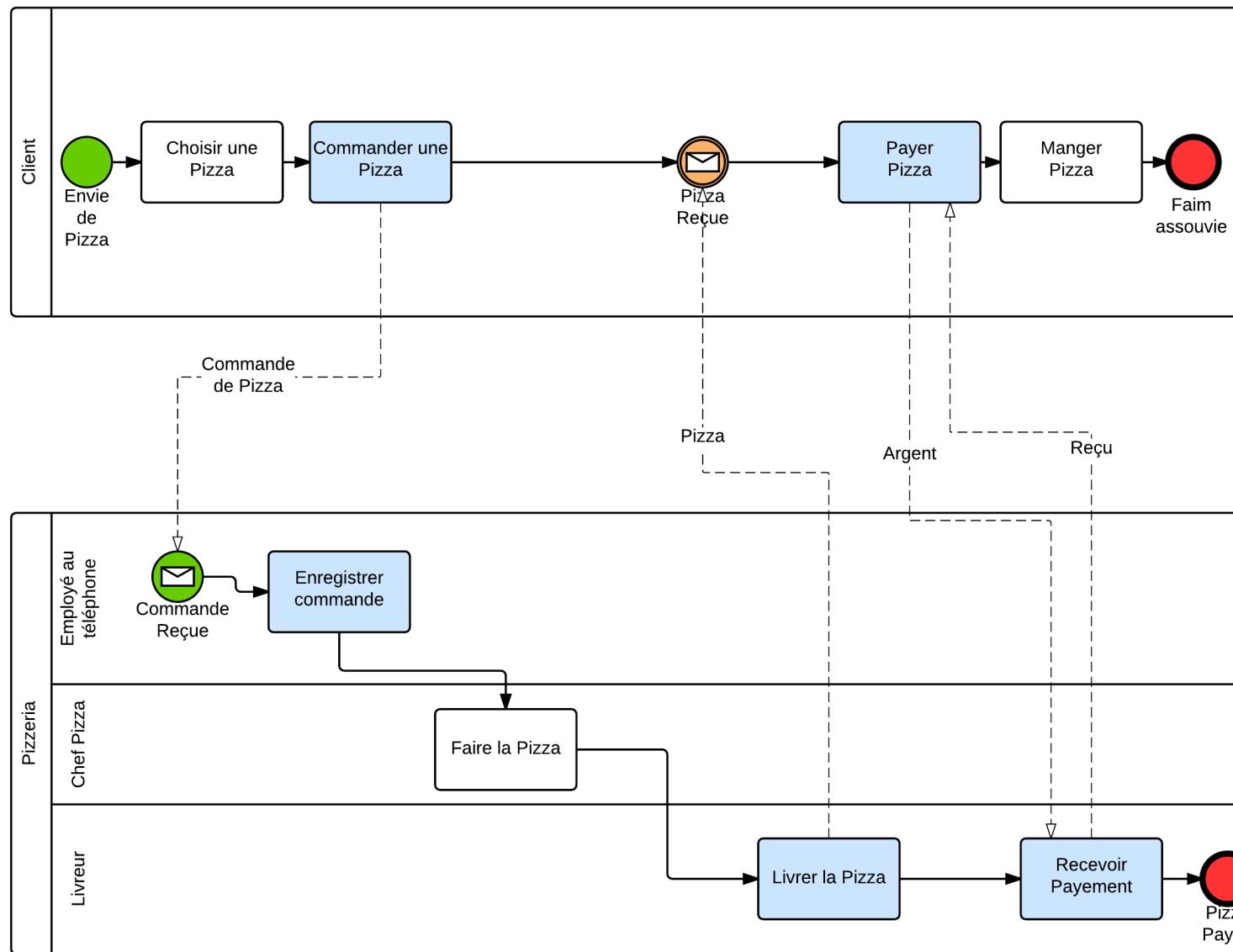
Inclusive (OR)

Exclusive (XOR)

Parallèle (AND)

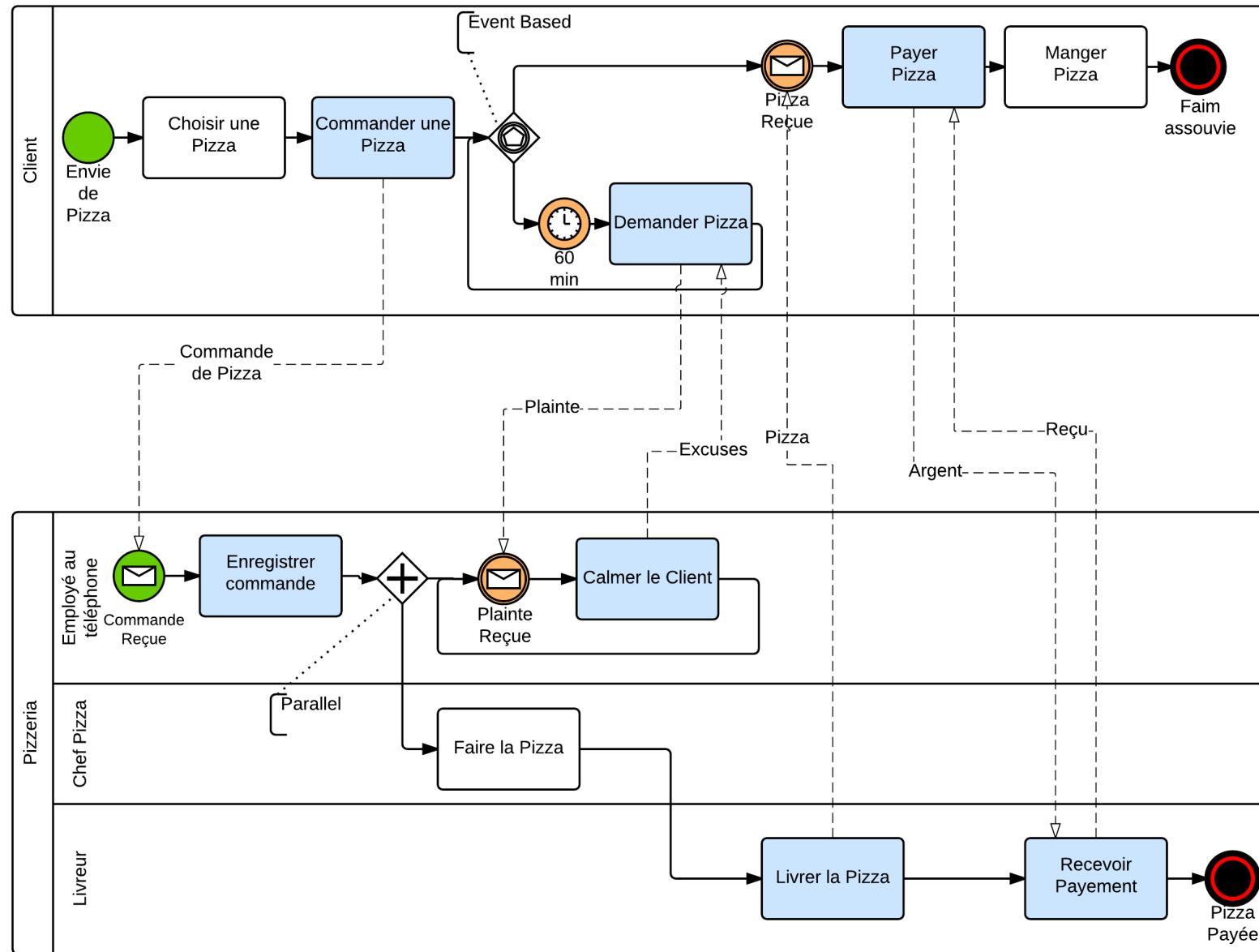
Décision basée sur des événements

# Evénements/Branchements

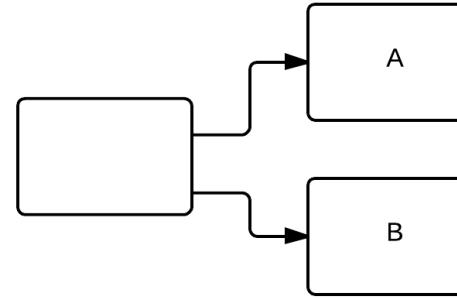
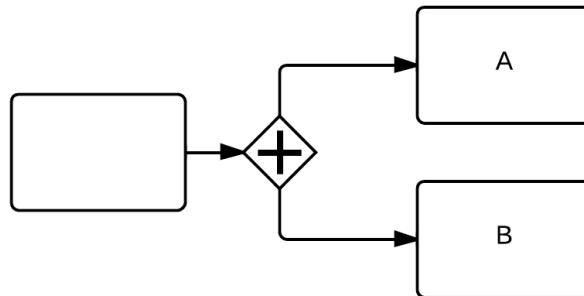


Si 60 minutes se sont écoulées sans recevoir la pizza, le client appelle pour se plaindre. L'employé doit calmer le client.

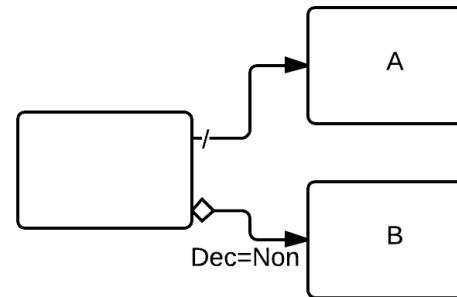
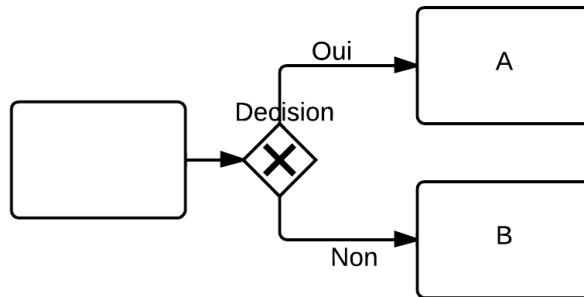
# Evénements/Branchements



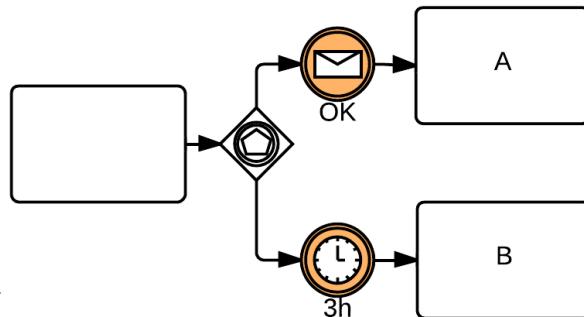
# Règles d'utilisation – Branchements



A **ET** B

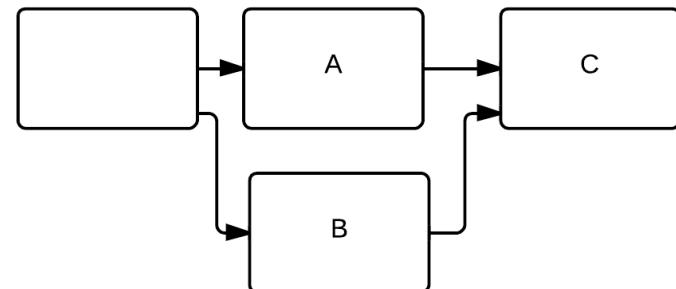
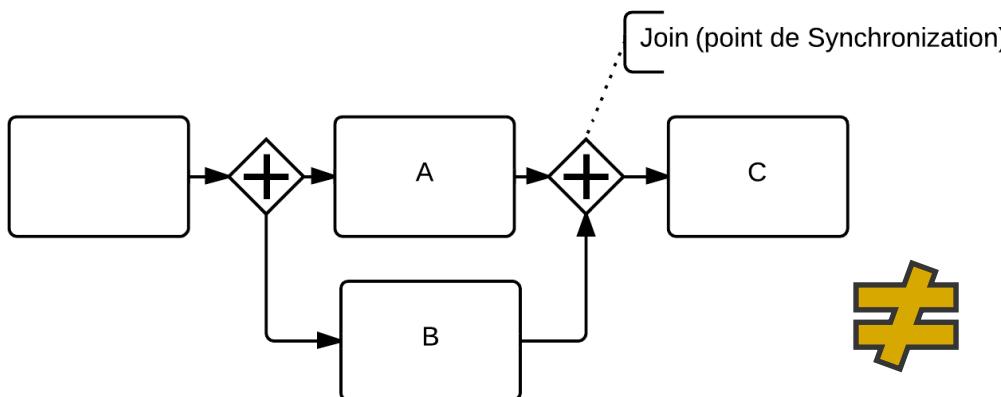
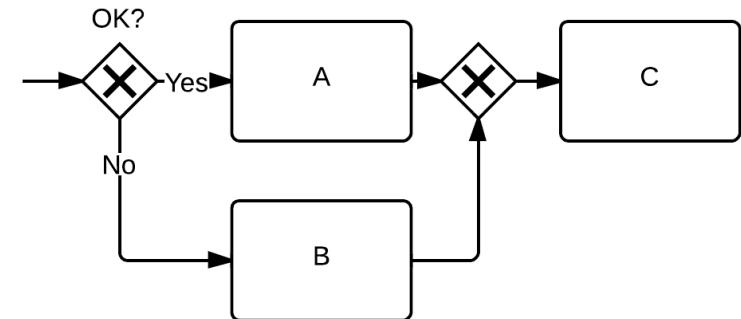
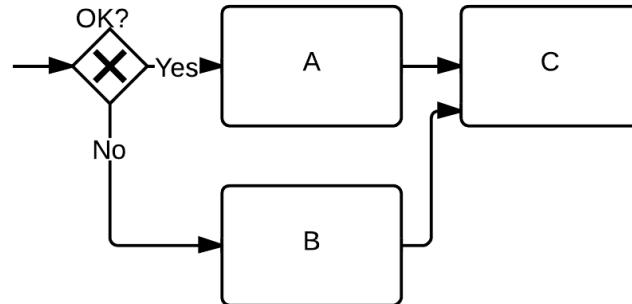
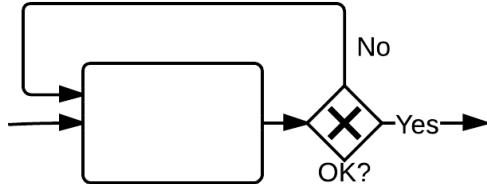


A **XOR** B



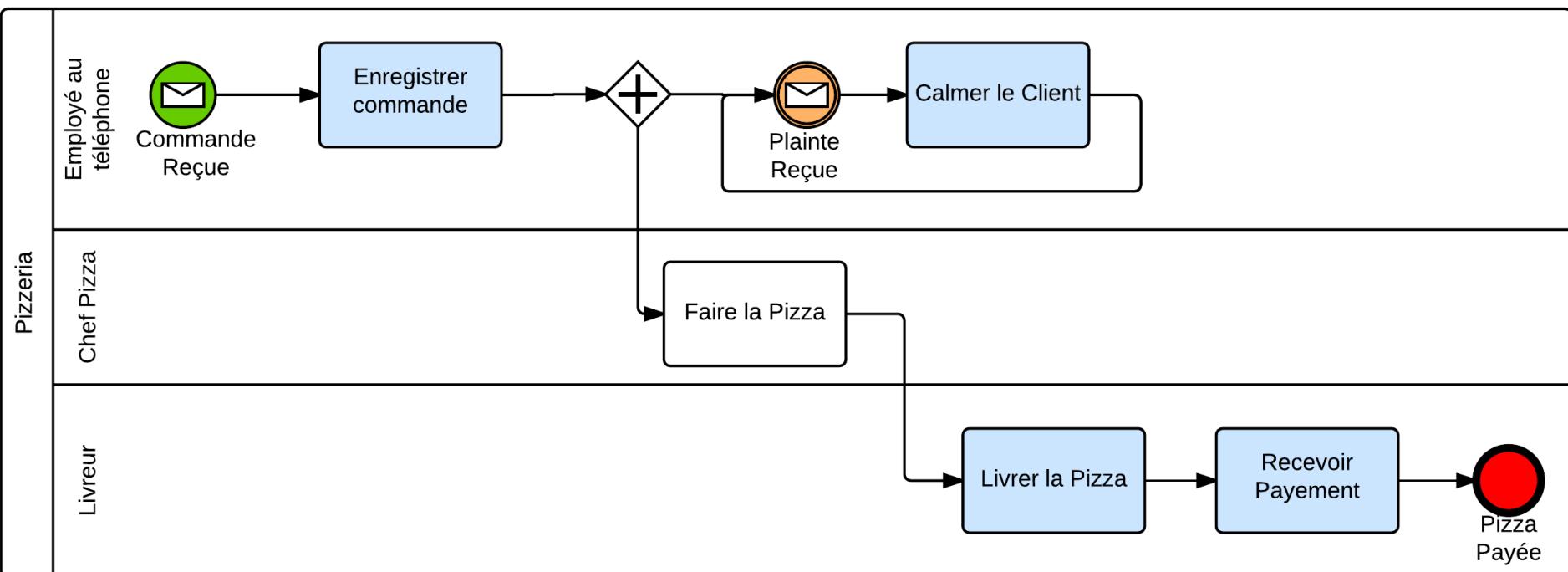
A **XOR** B

# Règles d'utilisation – Branchements



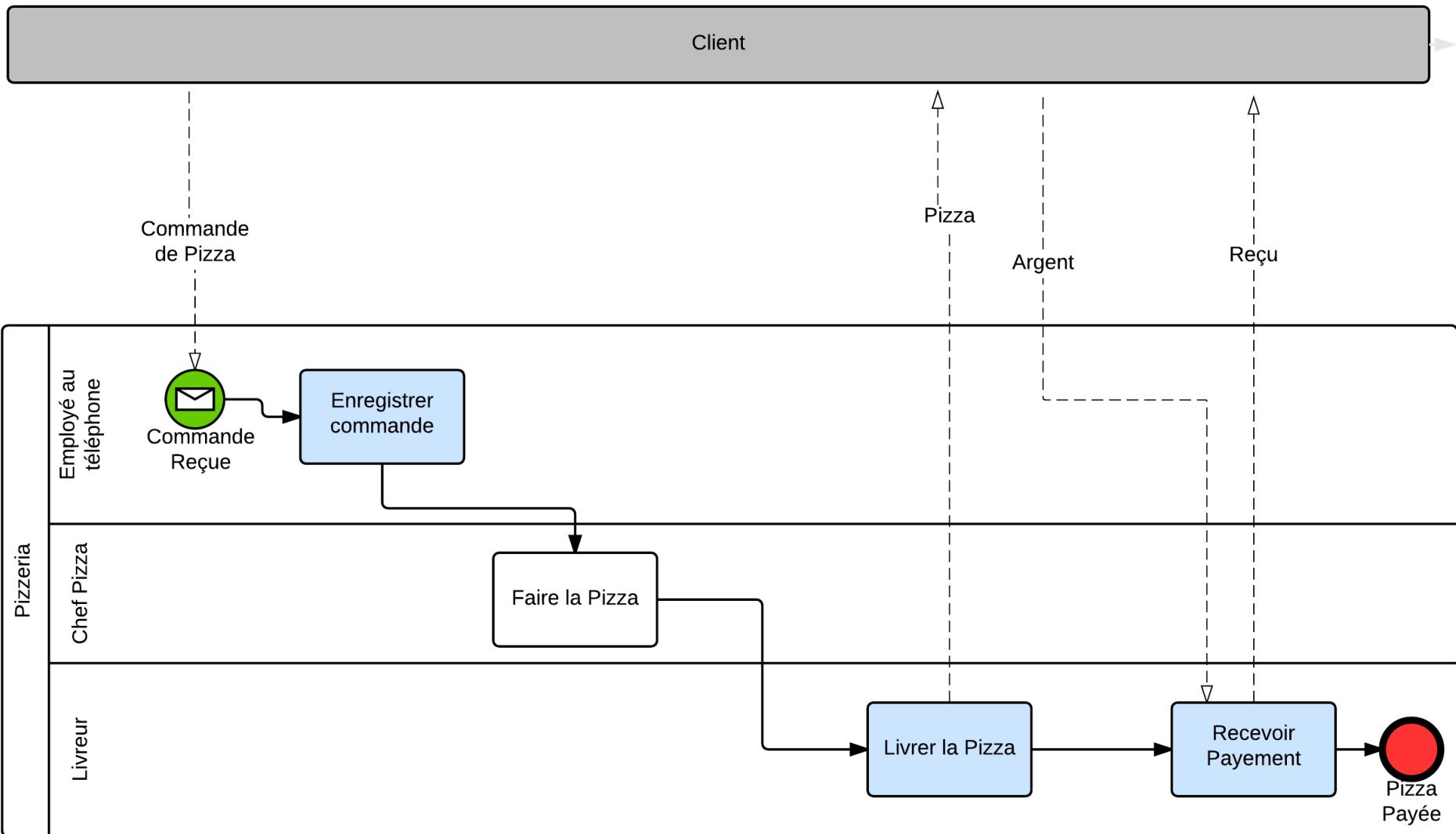
# Acteur externe

Le client est externe – Ses activités ne concernent pas le processus d'entreprise  
⇒ le processus démarre à la réception de la commande



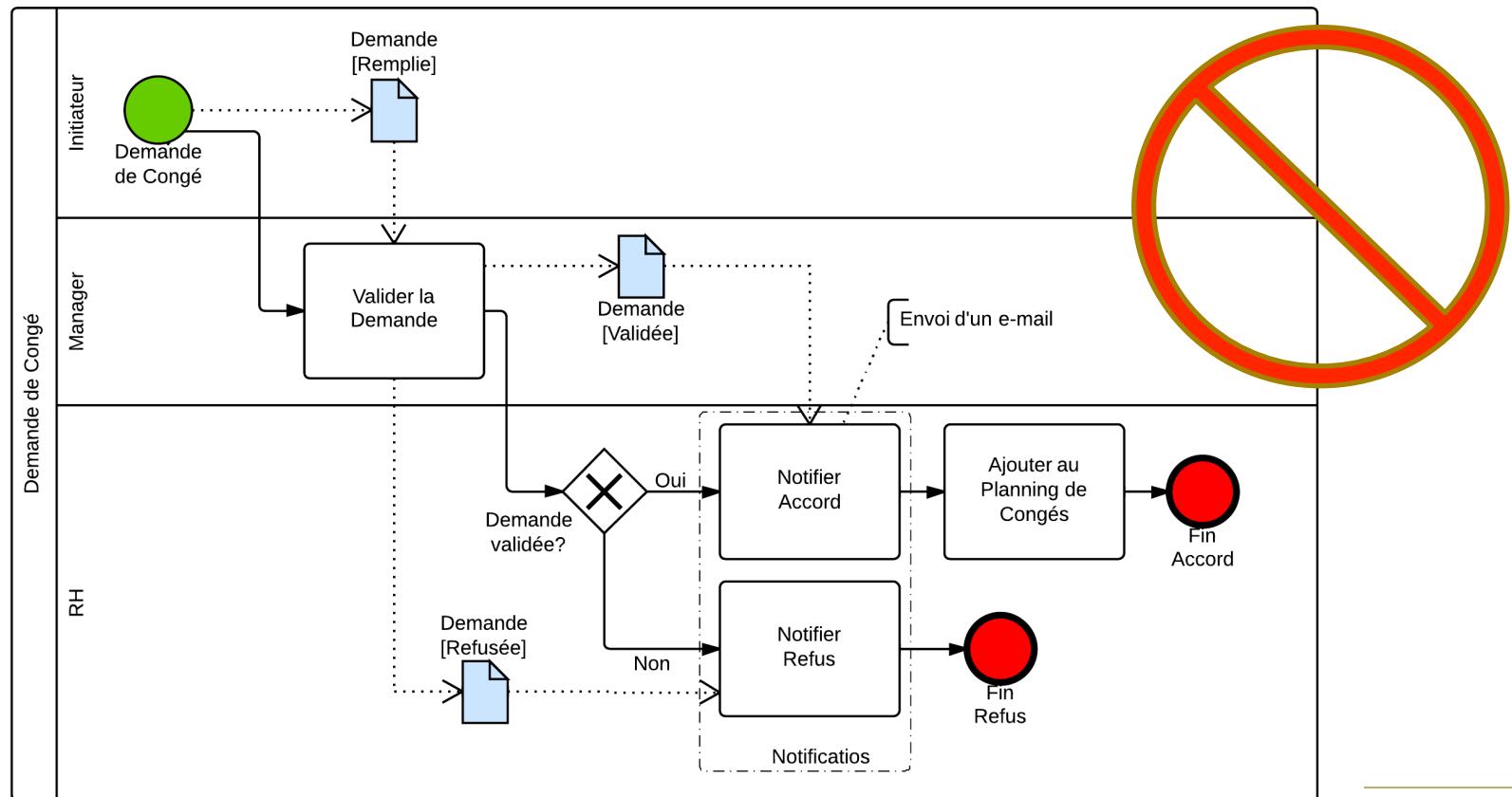
Processus privé ⇒ les acteurs externes ne sont pas modélisés

# Acteur externe dans des pools opaques

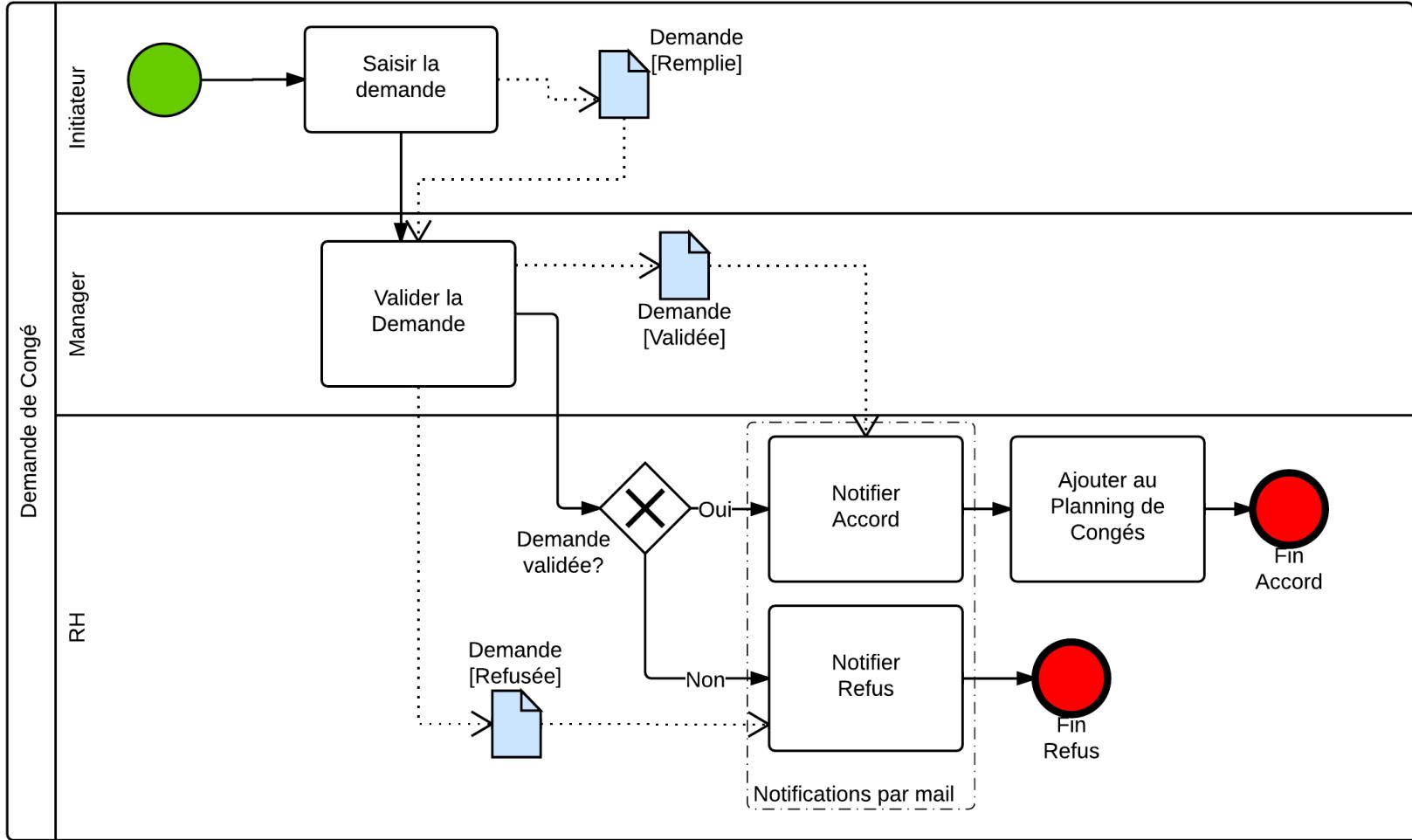


# Acteur interne ou externe ?

Et l'initiateur de la demande de congé ? – Tout dépend de ce que l'on veut contrôler. Est-ce que le processus démarre (est instancié) à l'arrivée de la demande de congé ou est-il instancié par le fait que l'initiateur souhaite saisir sa demande ?



# Acteur interne



# Principes d'usage/ Bonnes Pratiques

1. Modéliser les **acteurs internes comme des Lanes** à l'intérieur d'un *Pool* et pas comme des *Pools* indépendants.
2. Utiliser des « **pools noirs (opaques)** » pour représenter des acteurs externes (nom de l'organisation ou du participant)
3. Nommer les « **pools blancs** » avec le nom du processus
4. Faire des **modèles hiérarchiques**, chaque processus est sur une seule page
5. Nommer les **activités verbe-nom**
6. Si possible, nommer les **branchements exclusifs avec une question oui/non** et nommer les flux sortants avec oui et non
7. Nommer les **événements de message** avec la formule « **Recevoir X** » ou « **X Reçu** » où X correspond à l'objet envoyé.
8. Nommer les **flux de message avec le nom du message**
9. Utiliser **différents événements de fin**

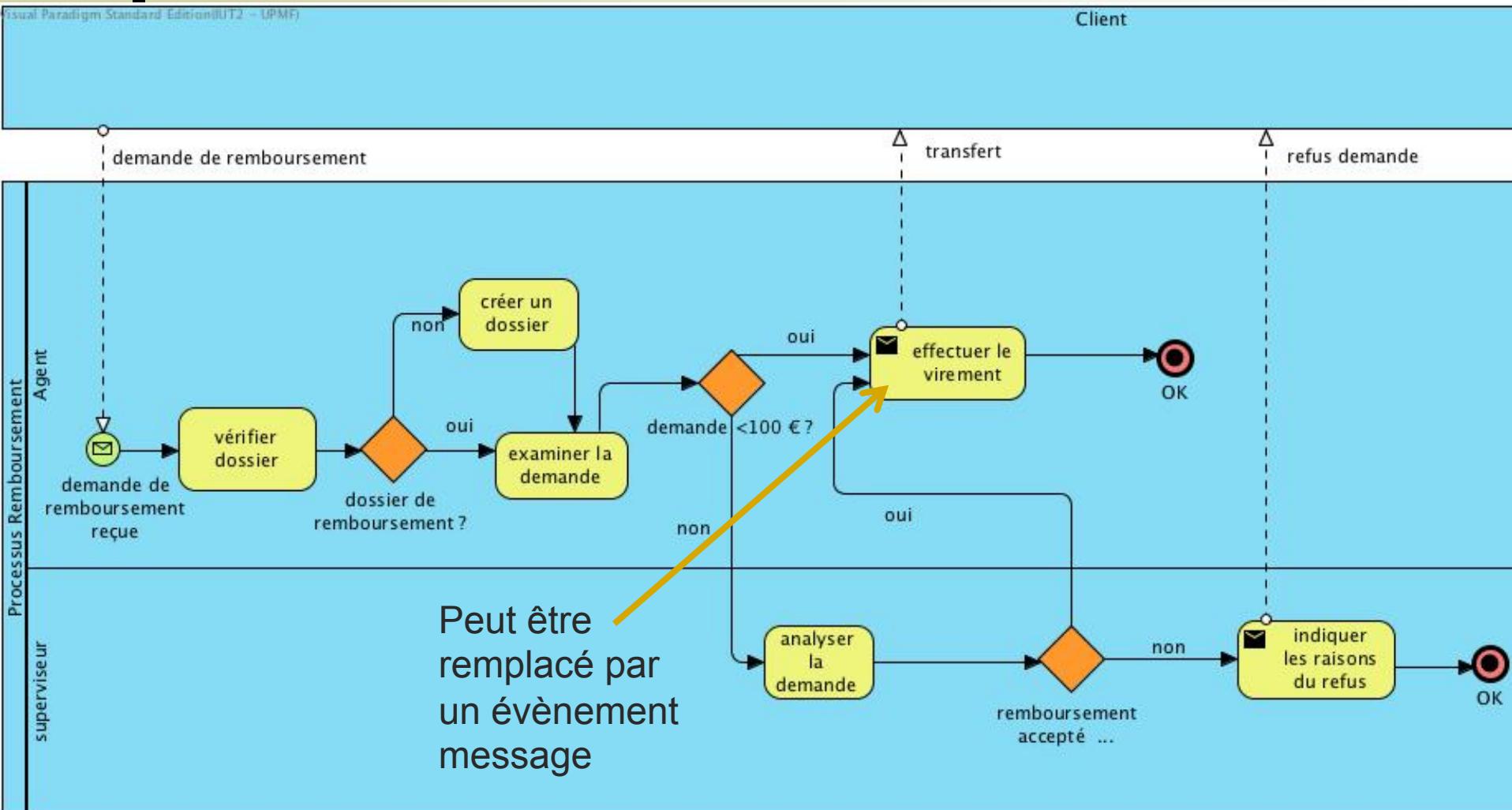
Source: Silver, B.: *BPMN Method and Style: A levels-based methodology for BPM process modeling and improvement using BPMN 2.0*

# Exercice (1)

La société Truc souhaite modéliser son processus de remboursement qui se déroule de cette façon :

- Le client envoie une lettre de remboursement.
- Après la réception de la demande, par un **agent** du service à la clientèle, un dossier de remboursement doit être créé si le client n'en possède pas.
- L'agent examine la demande. Une demande inférieure à 100 euros est automatiquement créditée au compte du client.
- Une demande de plus de 100 euros doit recevoir l'approbation du **superviseur**.
- Un remboursement s'effectue automatiquement sur le compte du client.
- S'il y a un refus, le client doit en être informé par courriel.

# Correction exercice (1)

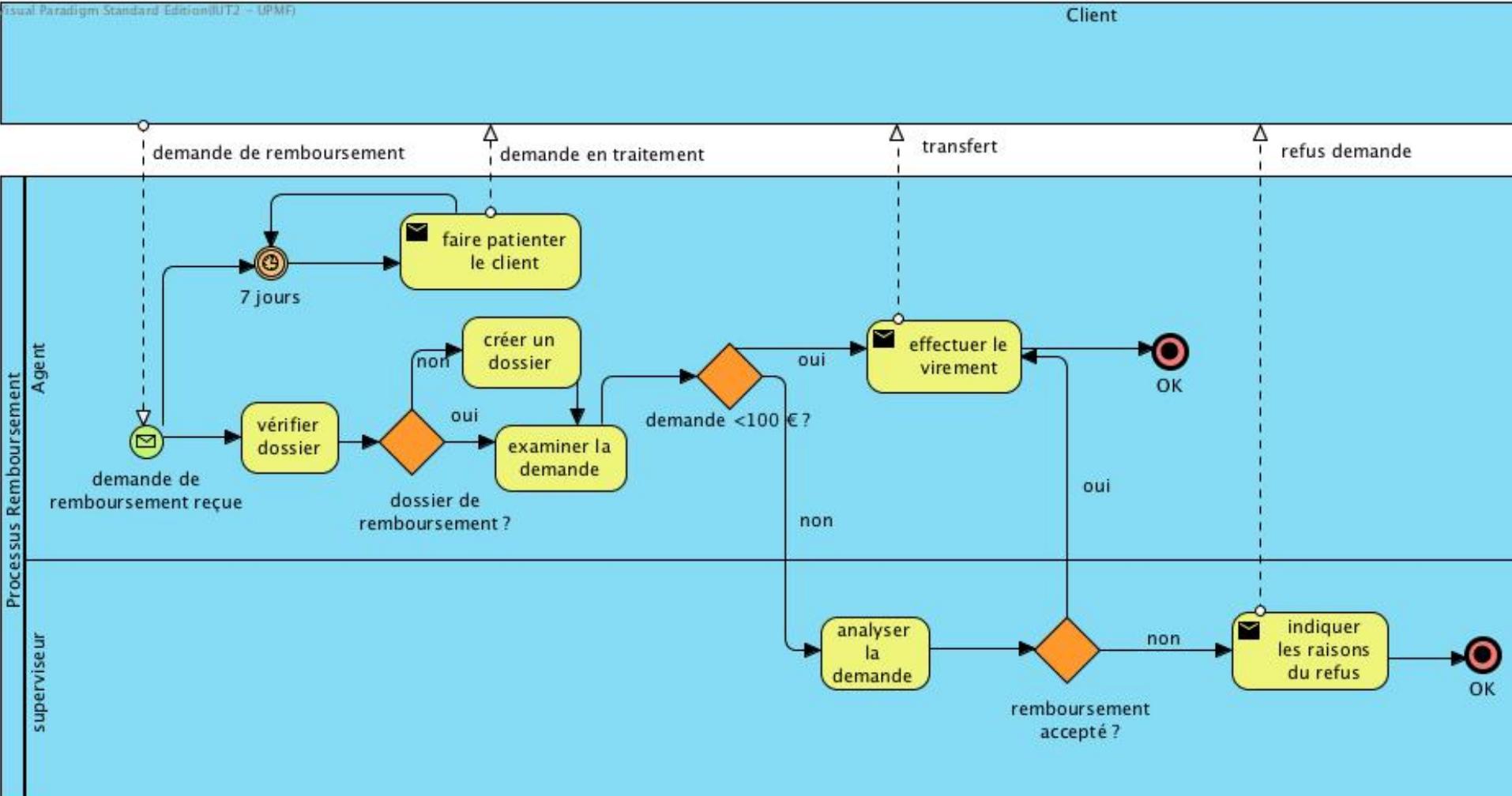


## Exercice (2)

La société Truc est satisfaite de votre modélisation mais réalise que ce processus n'est pas satisfaisant pour ses clients. Pour que les clients prennent patience on nous demande de prendre en compte la règle suivante :

- Si le processus n'est pas terminé au bout de 7 jours, l'agent doit envoyer un courriel au client indiquant que la demande est en traitement. Ce courriel sera envoyé tous les 7 jours jusqu'à ce que la demande soit satisfaite.

# Correction exercice (2)



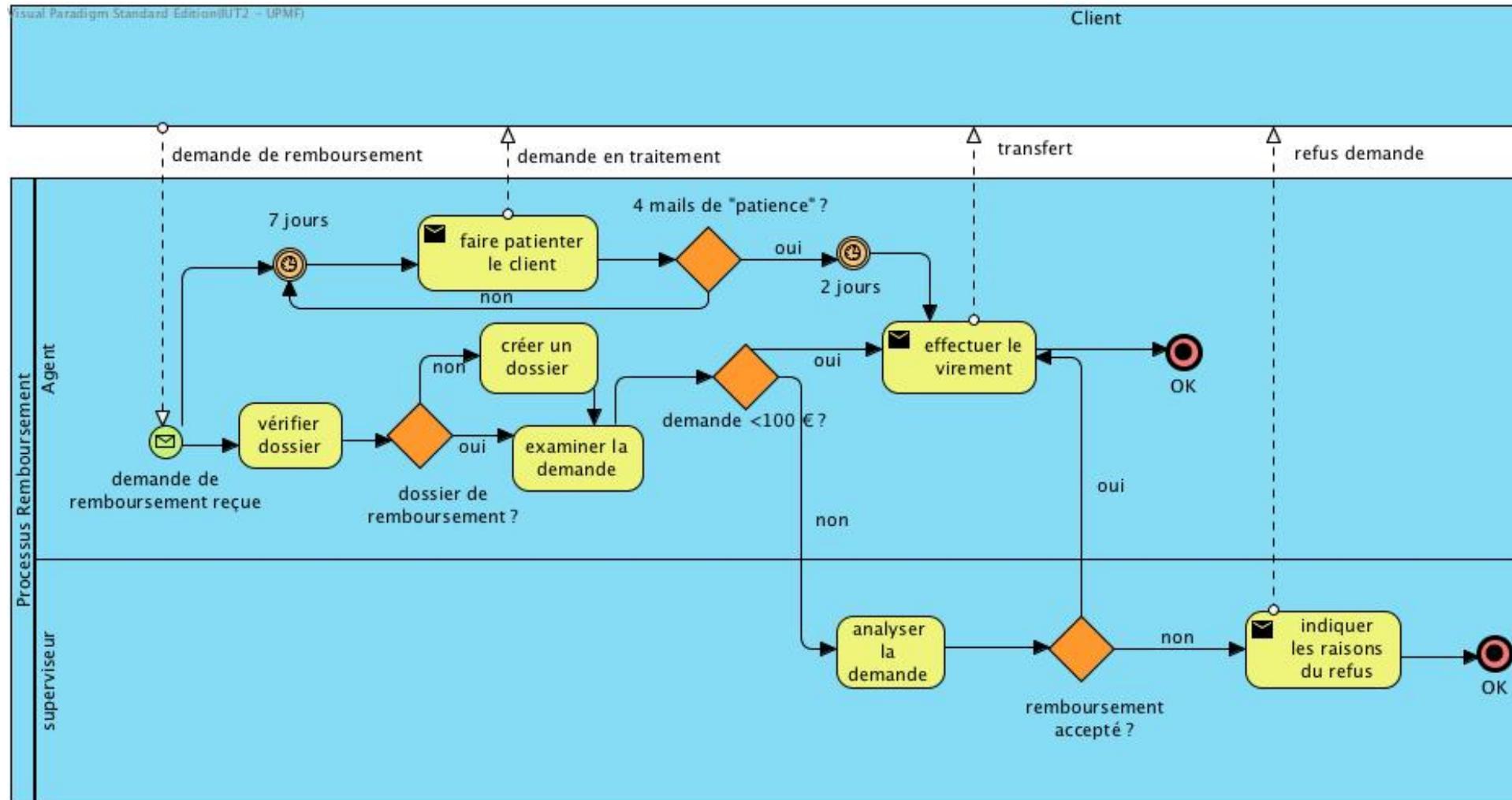
# Exercice (3)

Suite à de nombreuses plaintes des clients, on nous demande de prendre en compte une nouvelle règle :

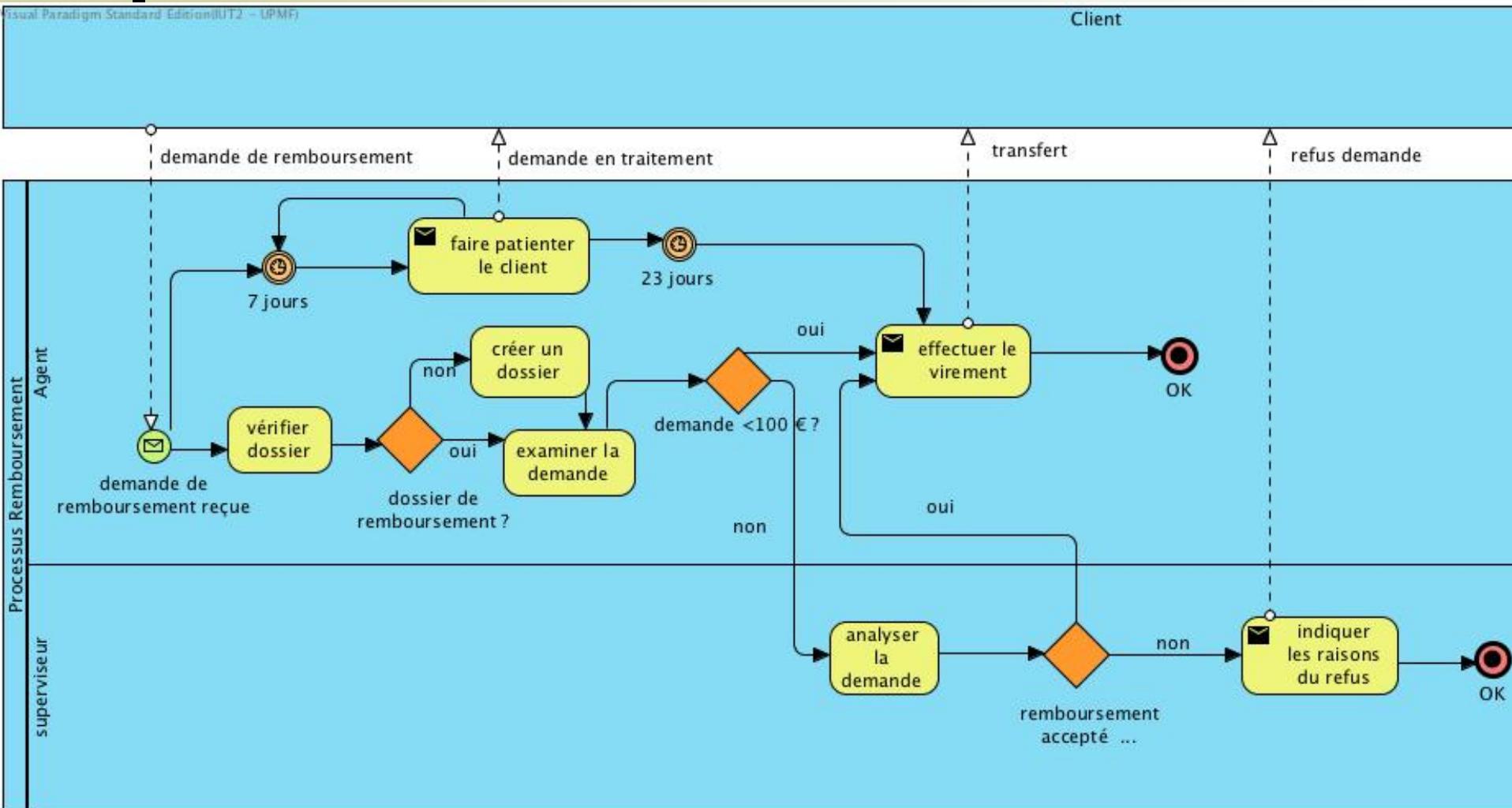
- Après 30 jours, la demande est automatiquement approuvée.

# Correction exercice (3)

Solution juste mais « lourde »



# Correction exercice (3)

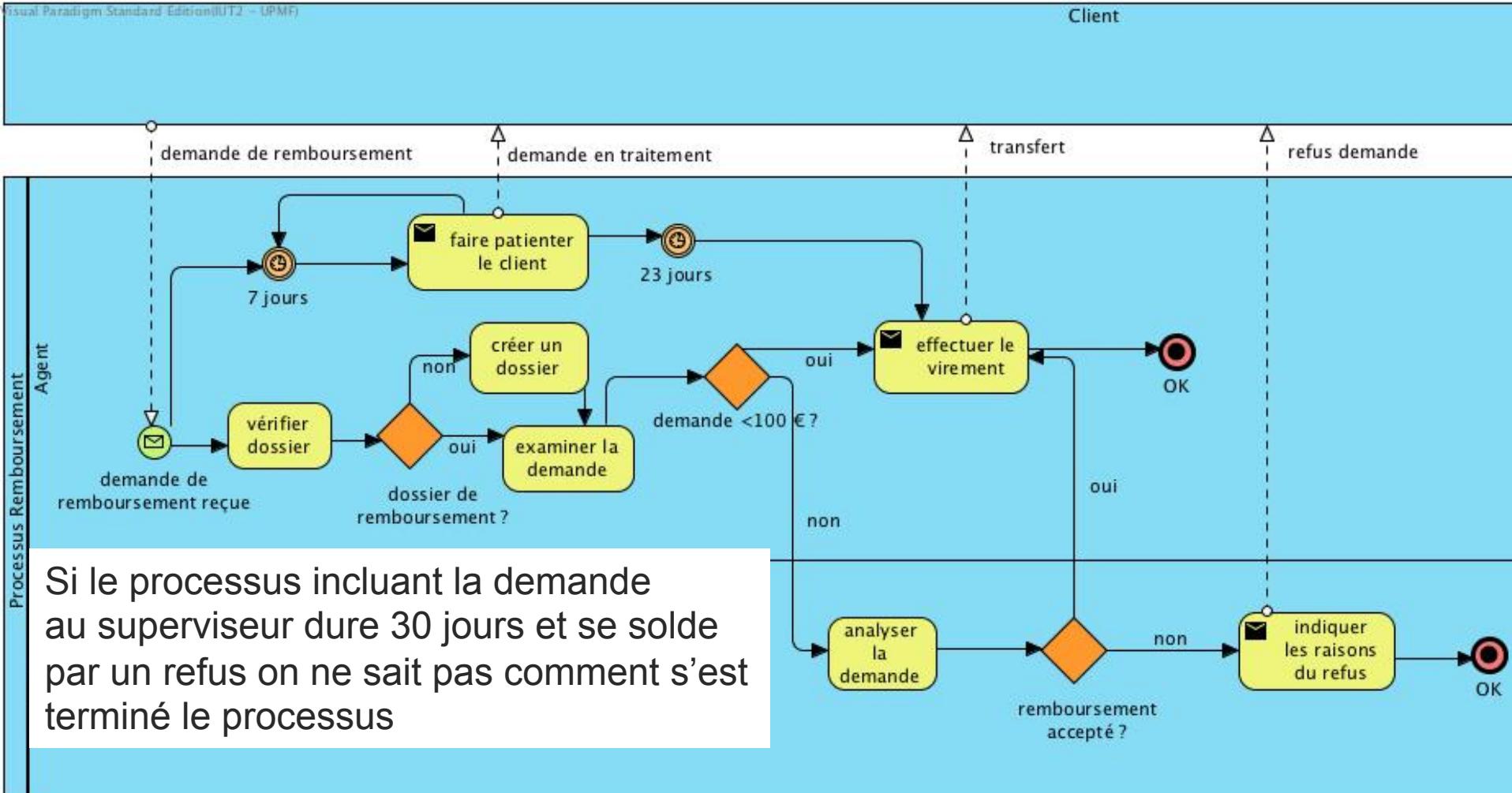


# [ Exercice (4) ]

Nous sommes contents de nous... mais notre processus comporte un risque. Lequel ?

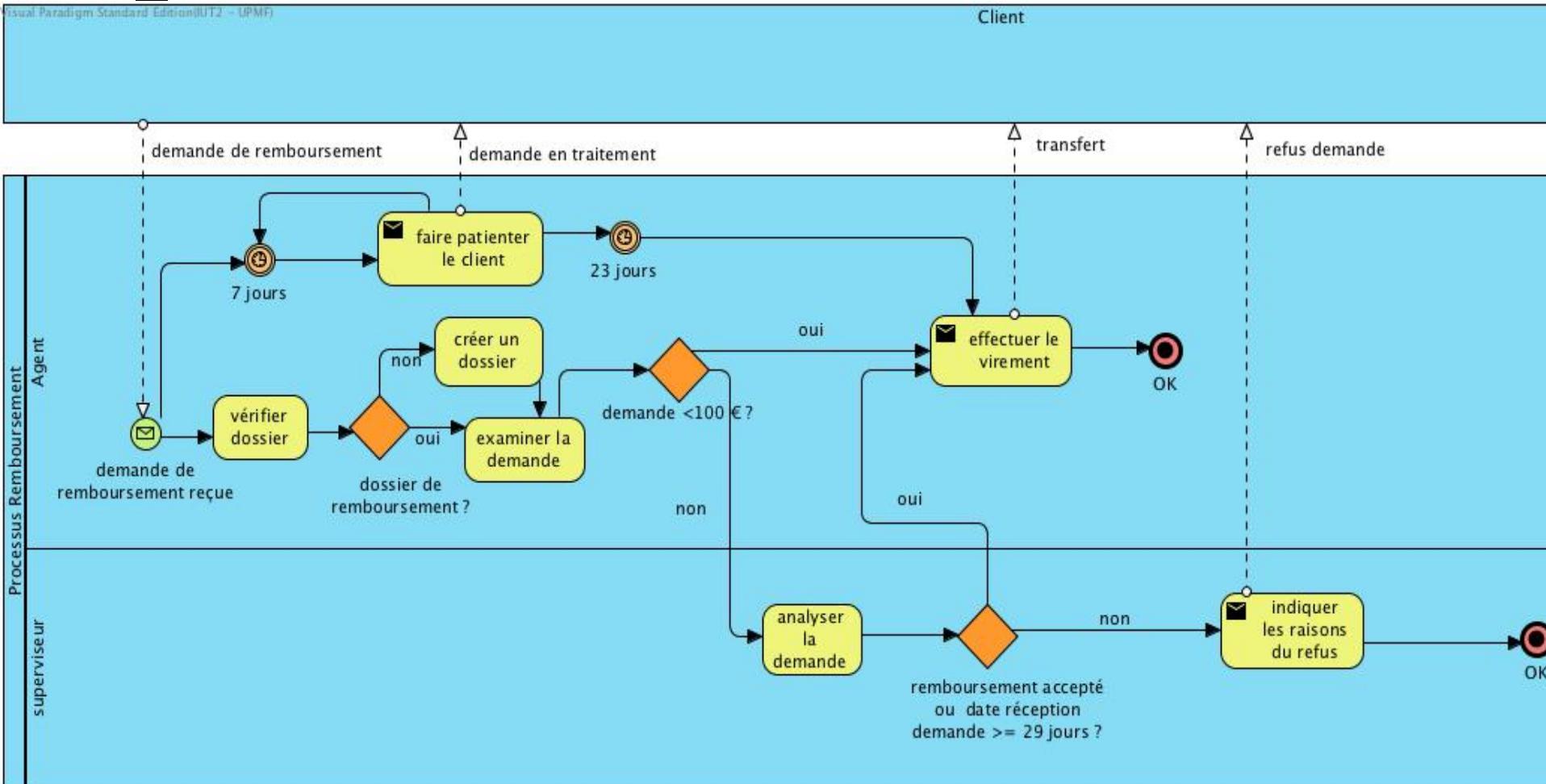
... un indice : le problème pourra se poser 30 jours après réception d'une demande de remboursement

# Correction exercice (4)



Si le processus incluant la demande au superviseur dure 30 jours et se solde par un refus on ne sait pas comment s'est terminé le processus

# Correction exercice (4)



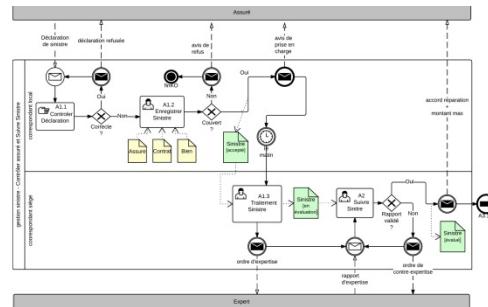
# Retour sur l'énoncé

La société Truc souhaite modéliser son processus de remboursement qui se déroule de cette façon :

- Le client envoie une demande de remboursement .
- Après la réception de la demande, par un **agent** du service à la clientèle, un dossier de remboursement doit être créé si le client n'en possède pas.
- L'agent examine la demande. Une demande inférieure à 100 euros est automatiquement créditée au compte du client.
- Une demande de plus de 100 euros doit recevoir l'approbation du **superviseur**. **Le superviseur doit prendre une décision au plus tard le 29ième jour. Si ce n'est pas le cas la demande est automatiquement approuvée le 30ième jour.**
- Un remboursement s'effectue automatiquement au compte client.
- S'il y a un refus, le client doit en être informé par courriel.
- ....

# Attention aux termes !

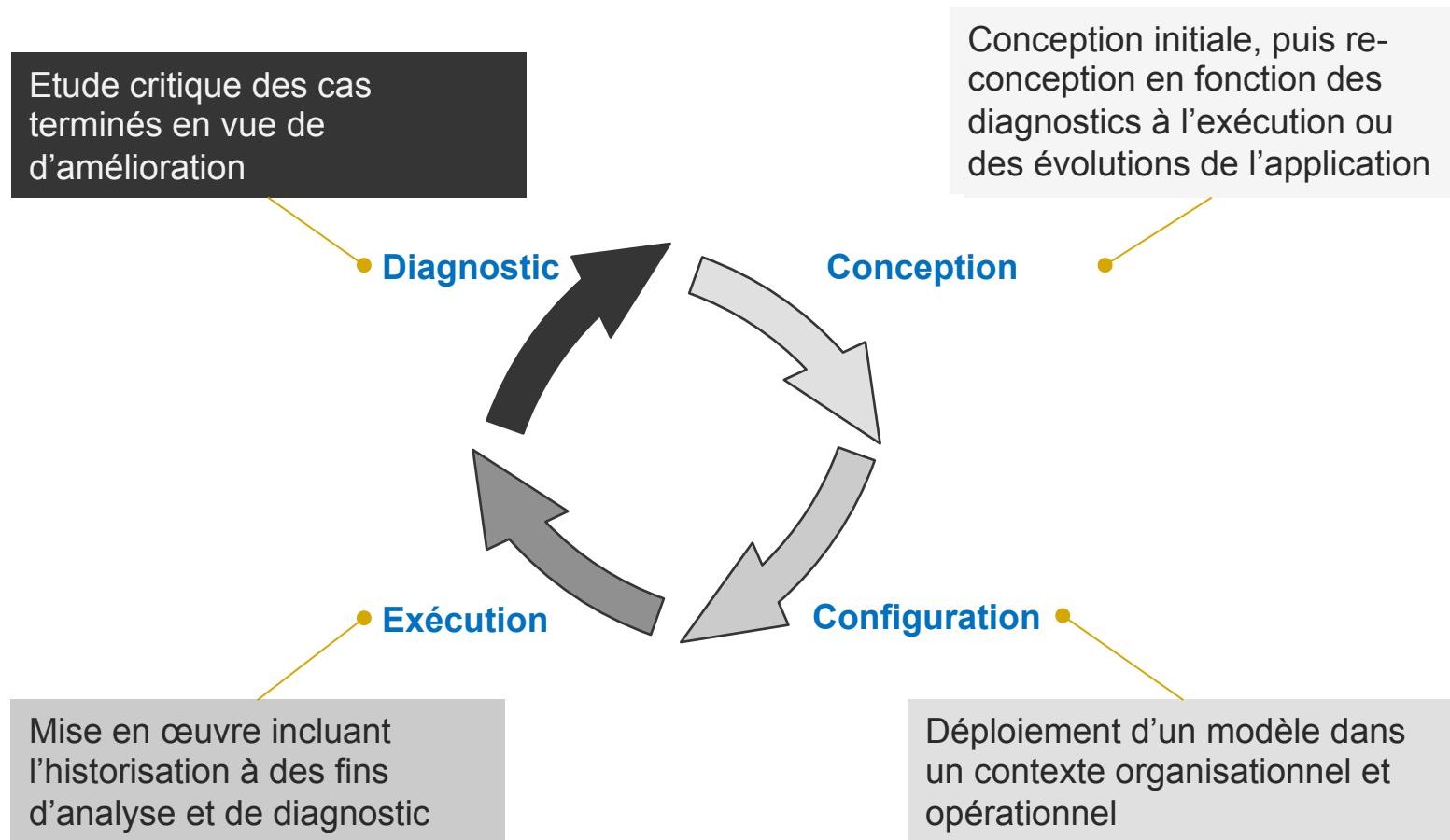
- **BPMN : Business Process Model and Notation**
  - Notation pour modéliser les processus métier



- **BPM : Business Process Management**
  - Gestion des processus métier
- **BPMS : BPM Systems**
  - Logiciels qui respectent la démarche BPM

# BPM - Cycle de vie de la Gestion des Processus Métier

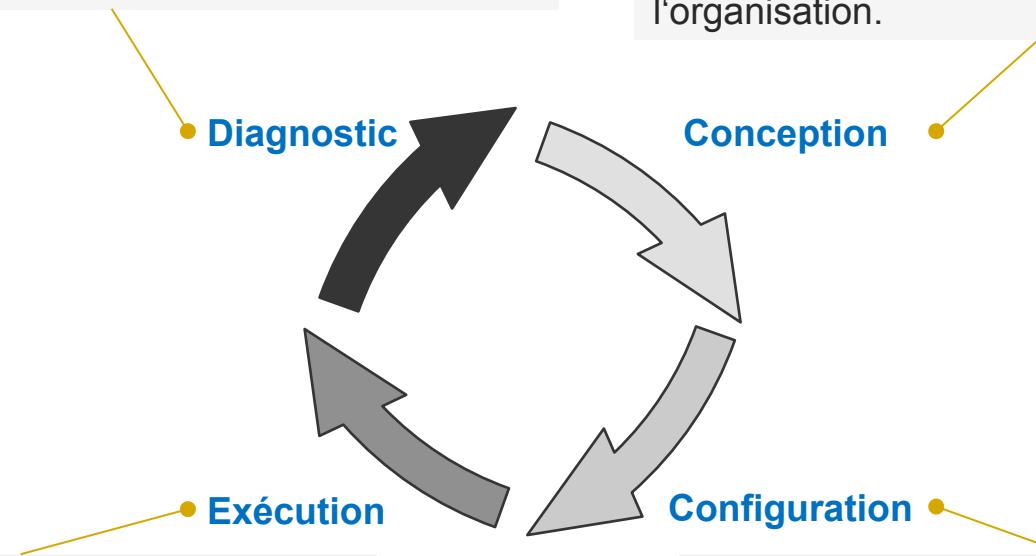
## Application du PDCA (*Plan Do Check Act*) de Deming



# BPMS - Outils pour la Gestion des Processus Métier

Des **outils de pilotage** et de *reporting* basés sur des indicateurs afin de disposer de tableaux de bord permettant de prendre rapidement les bonnes décisions.

Un **outil de modélisation** de processus, permettant de modéliser à l'aide d'une interface graphique les processus métiers de l'organisation.



Un **moteur d'exécution** (moteur de workflow) chargé d'instancier les processus et de stocker le contexte et leur état dans une base de données;

Des **outils d'aide à l'implémentation**, c'est-à-dire des interfaces (API) et des connecteurs permettant d'intégrer la solution de BPM au système d'information.

# Des outils très complets

Bonita Portal

BAM

Moteur d'exécution Bonita

Outil de modélisation

Bonita Studio

Conception

Diagnostic

Conception

Diagnostic

Bonita Portal

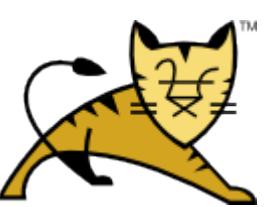
Moteur d'exécution

Moteur d'exécution Bonita

Connecteurs

Bonita Studio

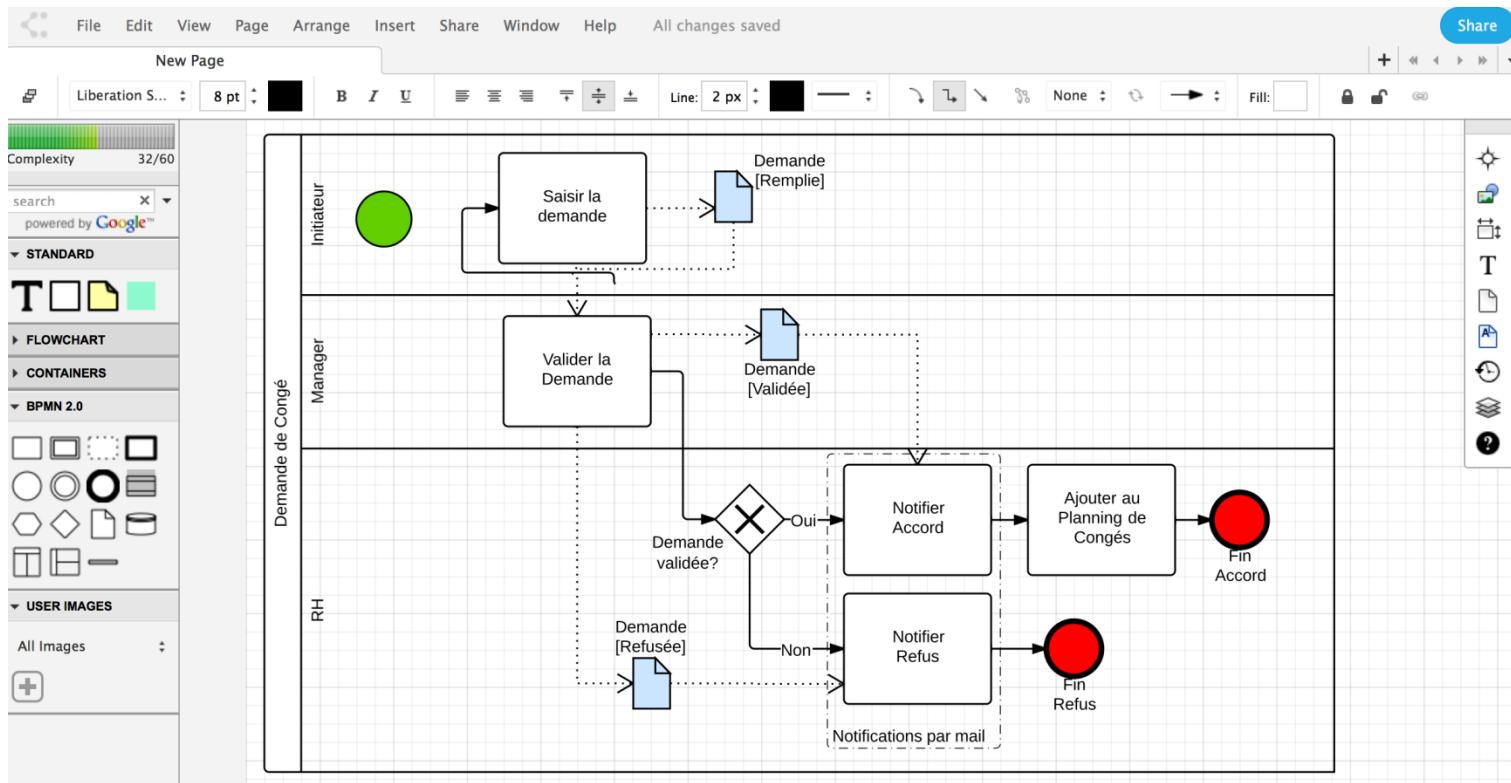
Configuration



Serveur Tomcat

# Mais aussi des outils « très légers »

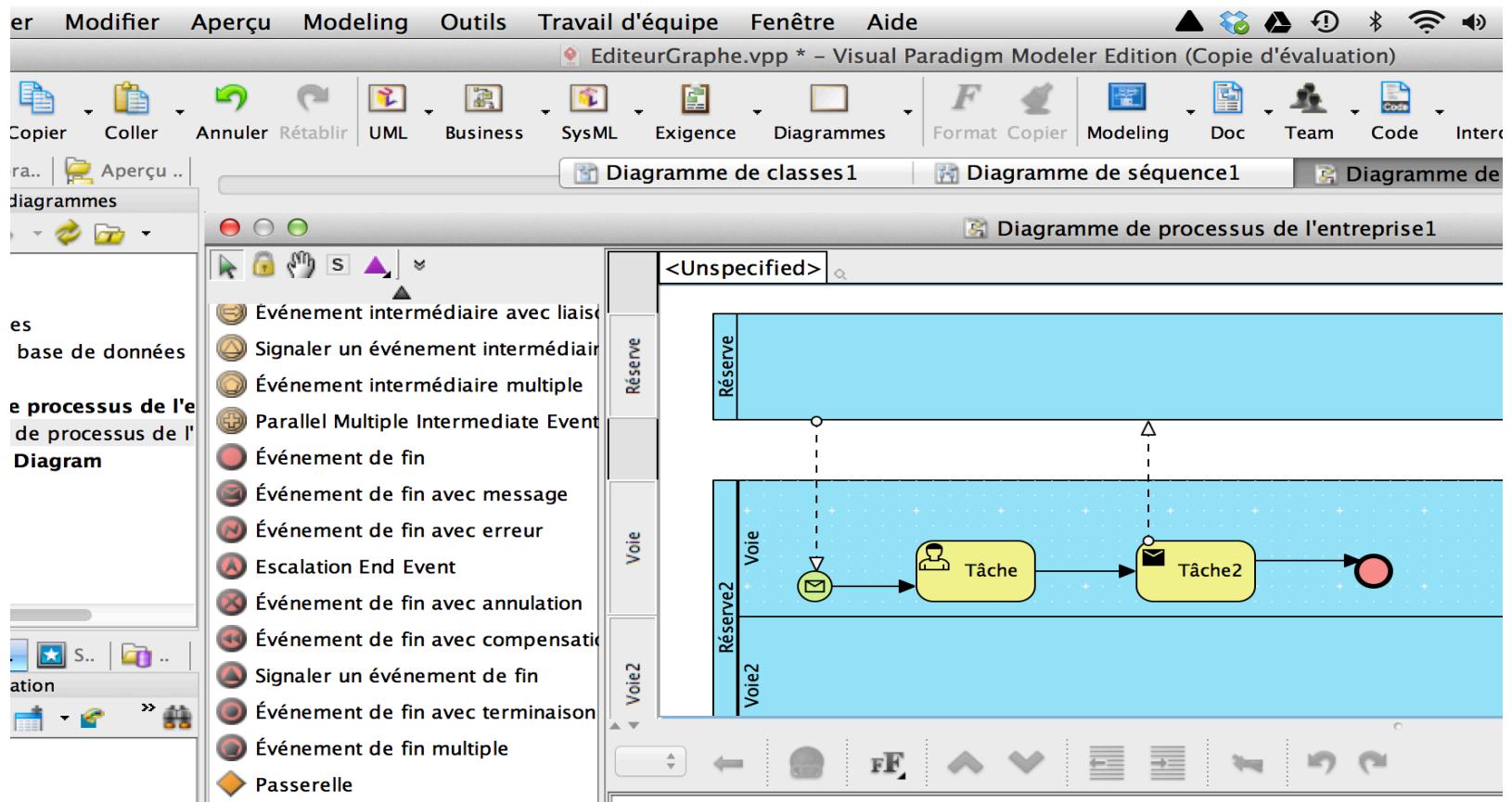
Exemple : lucidchart (un éditeur de dessin adapté)



*Tous les exemples  
de ce cours ont été  
réalisés avec ce  
logiciel*

<https://www.lucidchart.com/users/login>

# Des AGL tels que Visual Paradigm



# Visual Paradigm

- Visual Paradigm pour UML est installé sur les machines Linux et windows de l'IUT. Pour cette année :
  - La version la **13.1**
  - notre code d'activation est le **9U2S8-62437-MQW7N-L346A-G679J**
- Pour le lancer, tapez : **Visual\_Paradigm**
- Lors de la première utilisation :
  - Select **Window > License Manager** from the toolbar.
  - Click on **Change License**.
  - Click on **Subscription / Academic License**.
  - Expand **Academic Partner Program License**.
  - Enter the new activation code, name and email address.
  - Click **Activate**.
- Si vous voulez l'utiliser sur une autre machine que celles de l'IUT, il faut récupérer le logiciel Visual Paradigm pour UML. Attention de bien récupérer la **version 13.1**

Cette version est accessible sur <https://ap.visual-paradigm.com/iut2--upmf> (sur Chamilo voir fichier : VP sur vos machines dans le module Outils/Visual\_Paradigm)

# Bibliographie

- *Introduction au BPMN 2.0*, <http://introductionbpmn2.0.voila.net/co/IntroBPMN.html>
- Silver, B.: *BPMN Method and Style: A levels-based methodology for BPM process modeling and improvement using BPMN 2.0*. Cody-Cassidy Press, US (2009)
- OMG (2011). *Business Process Model and Notation (BPMN)* <http://www.omg.org/spec/BPMN/2.0>

# [ Démarche de modélisation ]

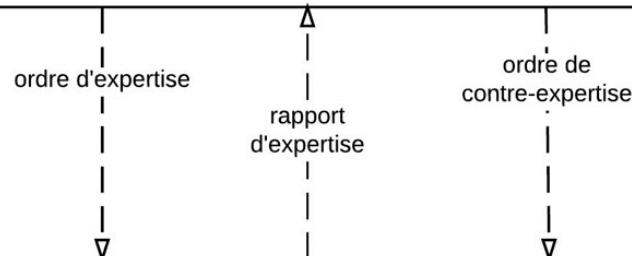
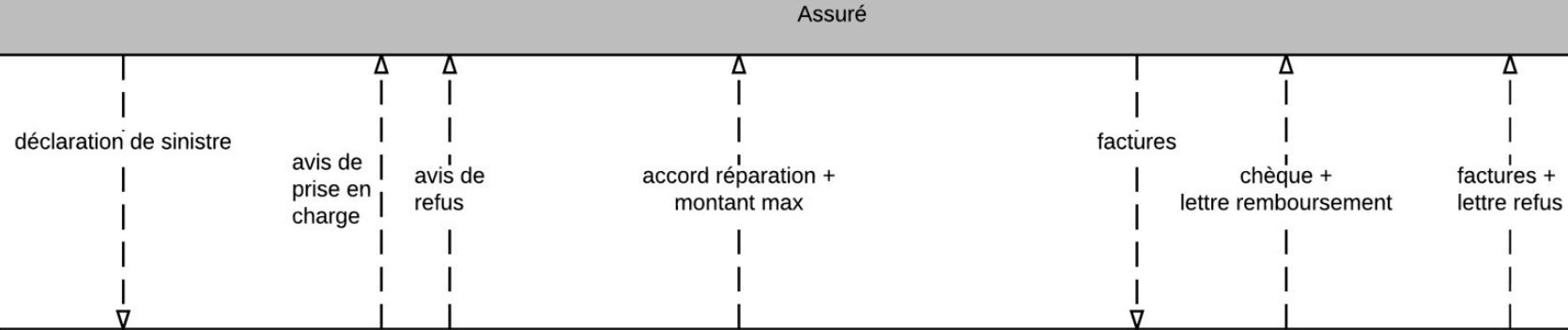
- Adopter une approche incrémentale !
- Niveau conceptuel (intentionnel) puis organisationnel
- Par l'exemple

# Niveau Conceptuel

- L'incontournable ! Indépendant de toute organisation et de tout choix technique
- Décrire le processus par des phrases simples.
- RG1 : Pour donner lieu au traitement d'un sinistre, l'**assuré** doit communiquer une déclaration de sinistre.....
- RG2 : Toute décision (de prise en charge ou de refus) suite à une déclaration de sinistre doit être communiquée par écrit à l'assuré.
- RG4 : Tout sinistre pris en charge donne lieu à la désignation d'un **expert**. Une demande d'expertise est communiquée à l'expert. Après expertise, l'expert doit renvoyer un rapport d'expertise. Les conclusions des rapports d'expertise doivent être enregistrés.....

## Niveau conceptuel

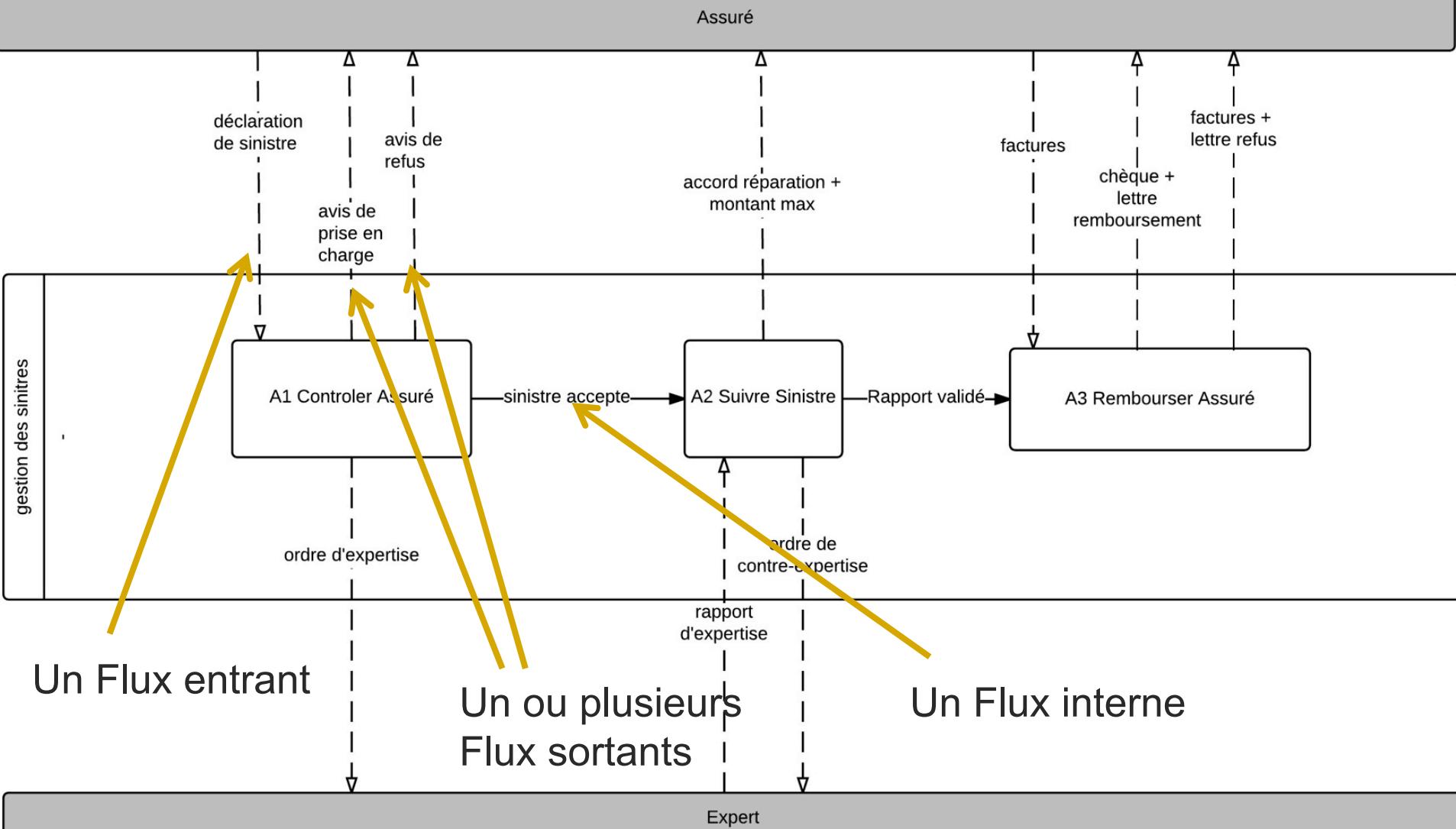
# Exemple : gestion des sinistres (1)



Expert

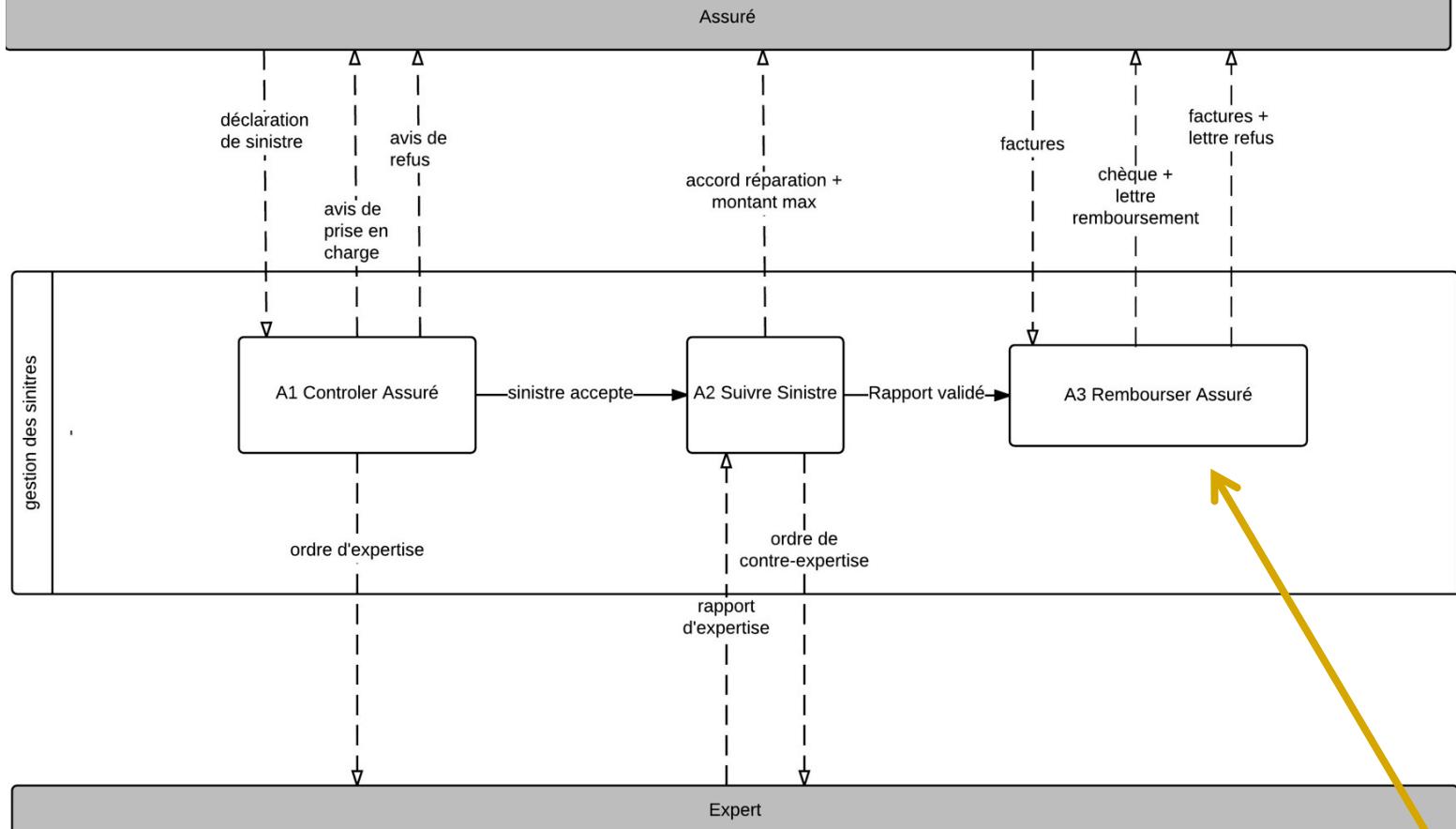
## Niveau conceptuel

# Exemple : gestion des sinistres (2)



## Niveau conceptuel

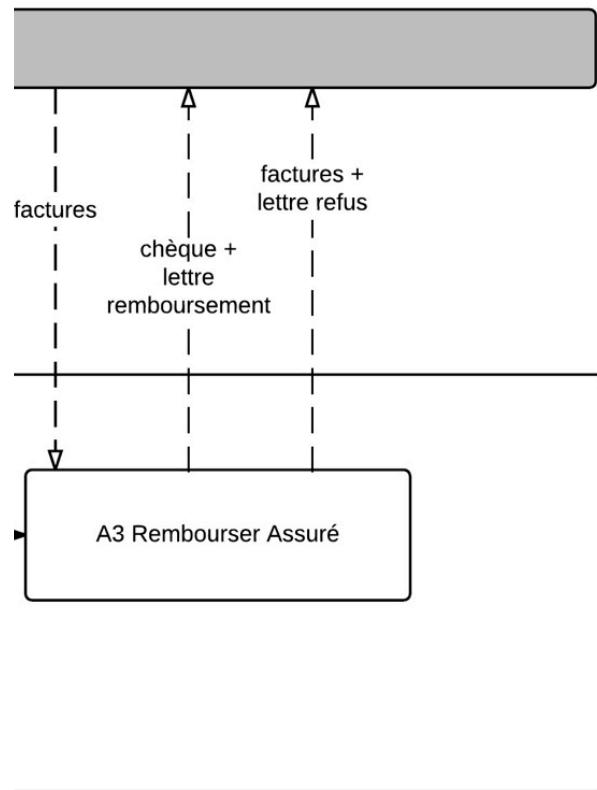
# Rajout des évènements temporels



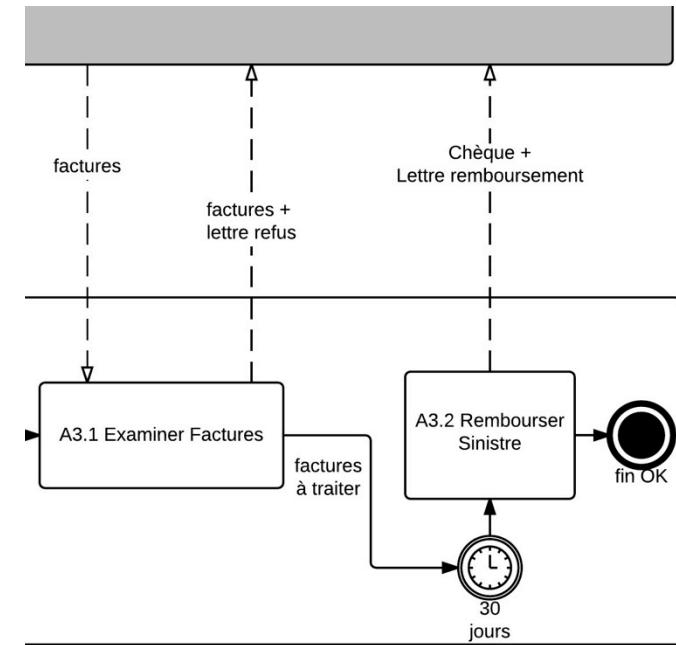
Les remboursements aux assurés s'effectuent par chèque au nom de l'assuré **30 jours après réception des factures**. Le remboursement est effectué en une seule fois.

*Niveau conceptuel*

# Rajout des évènements temporels

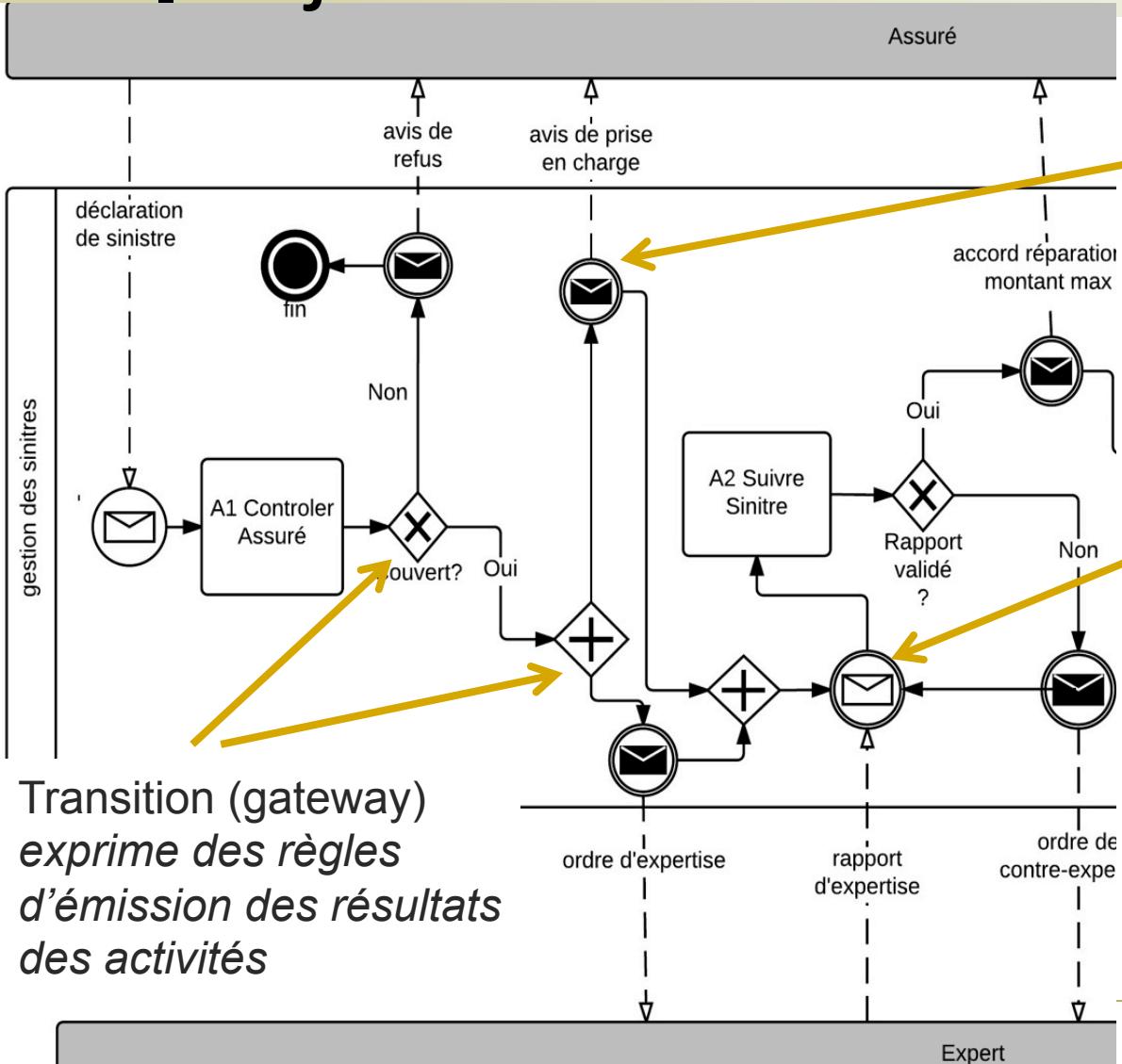


Événement temporel



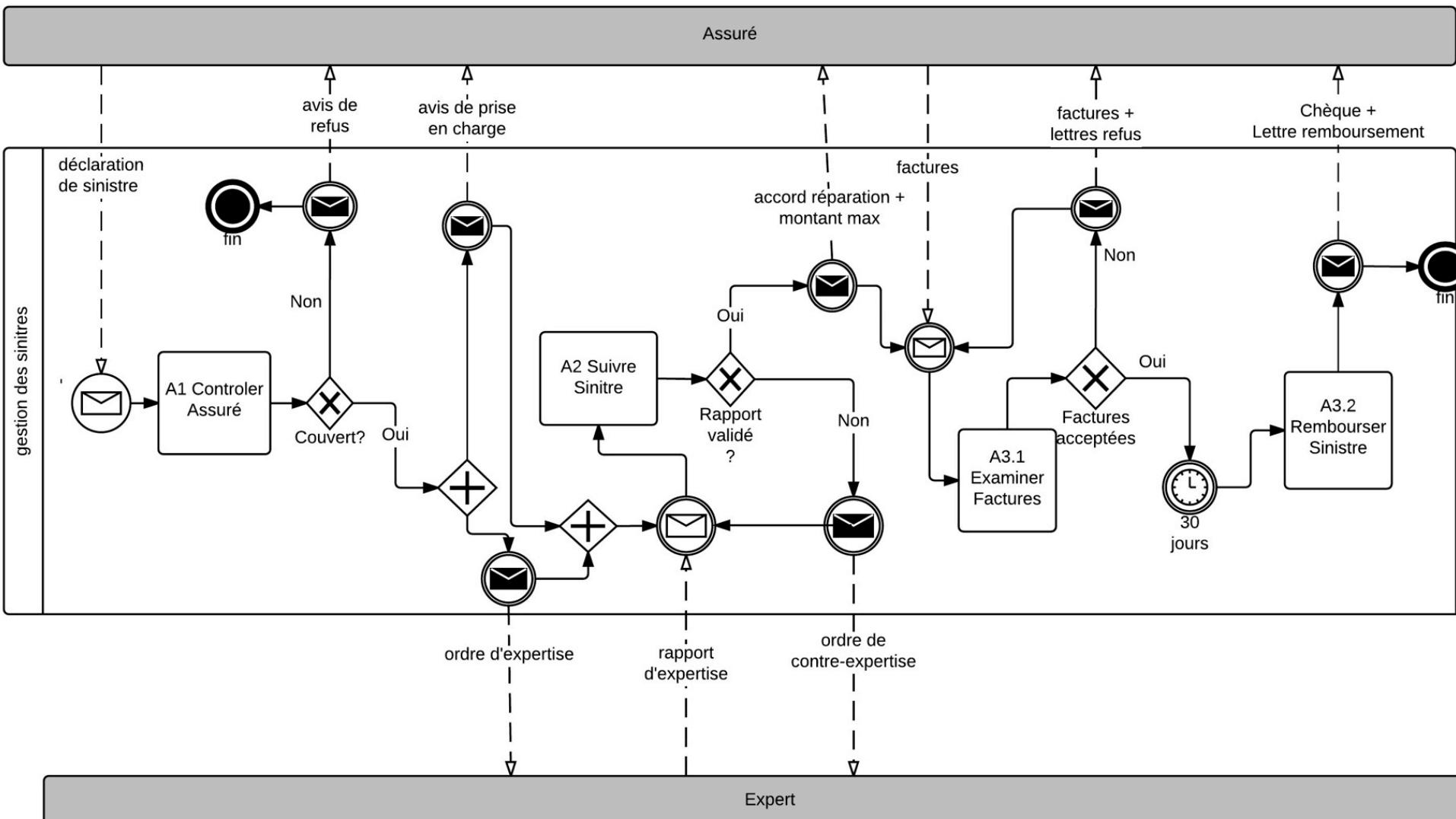
## Niveau conceptuel

# Rajout des évènements et transitions

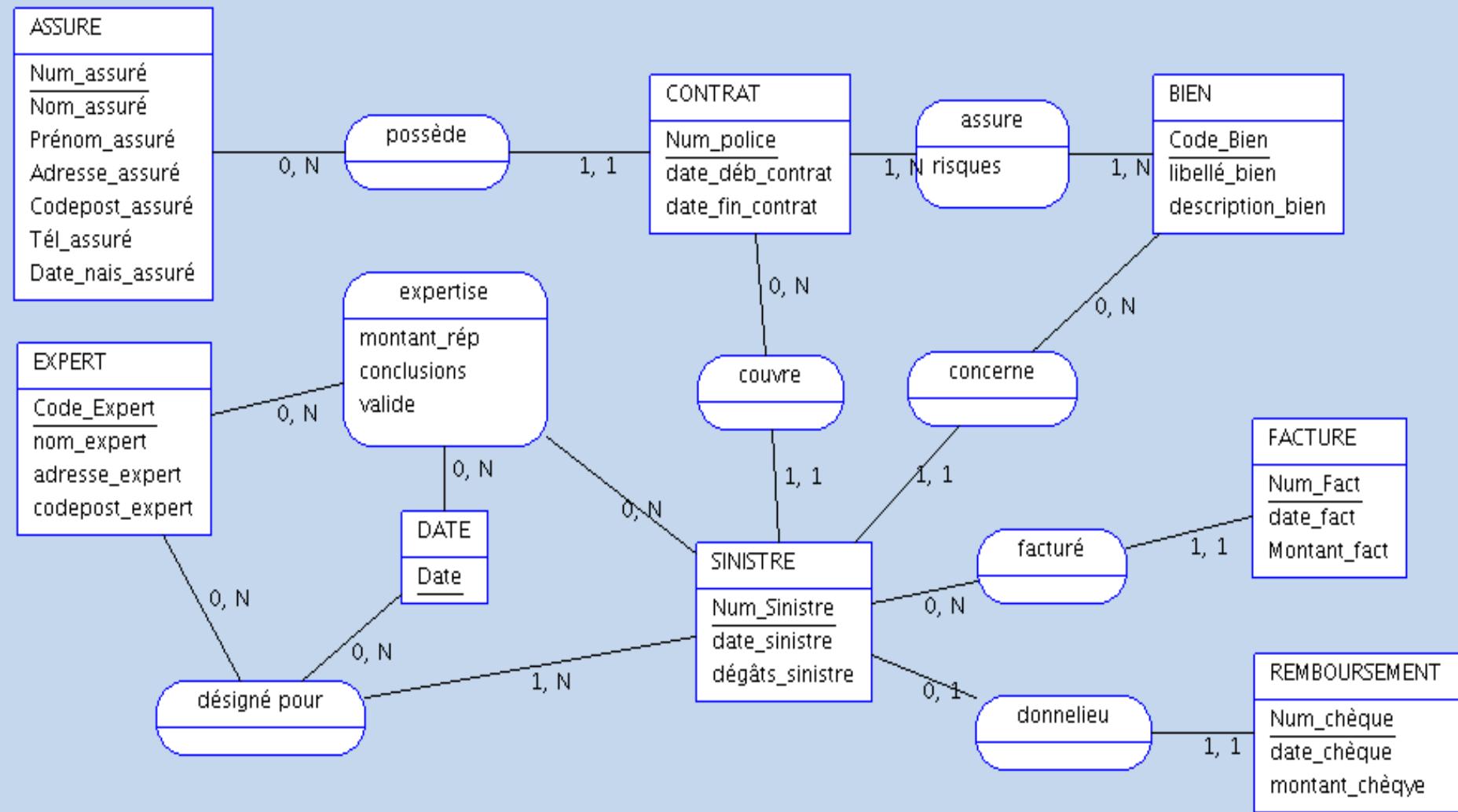


Niveau conceptuel

# Exemple : gestion des sinistres (3)

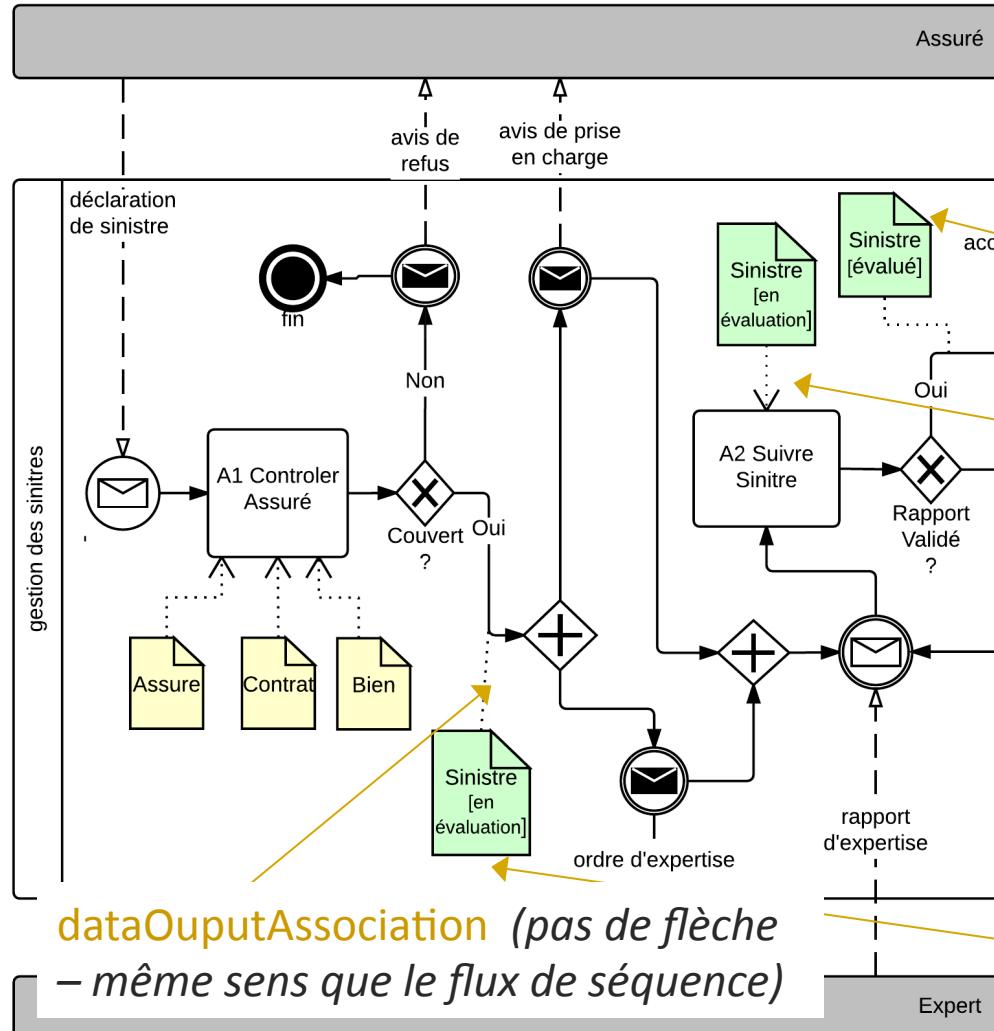


# Modèle de données



## Niveau conceptuel

# Couplage avec les données



Data object (une entité du MCD)

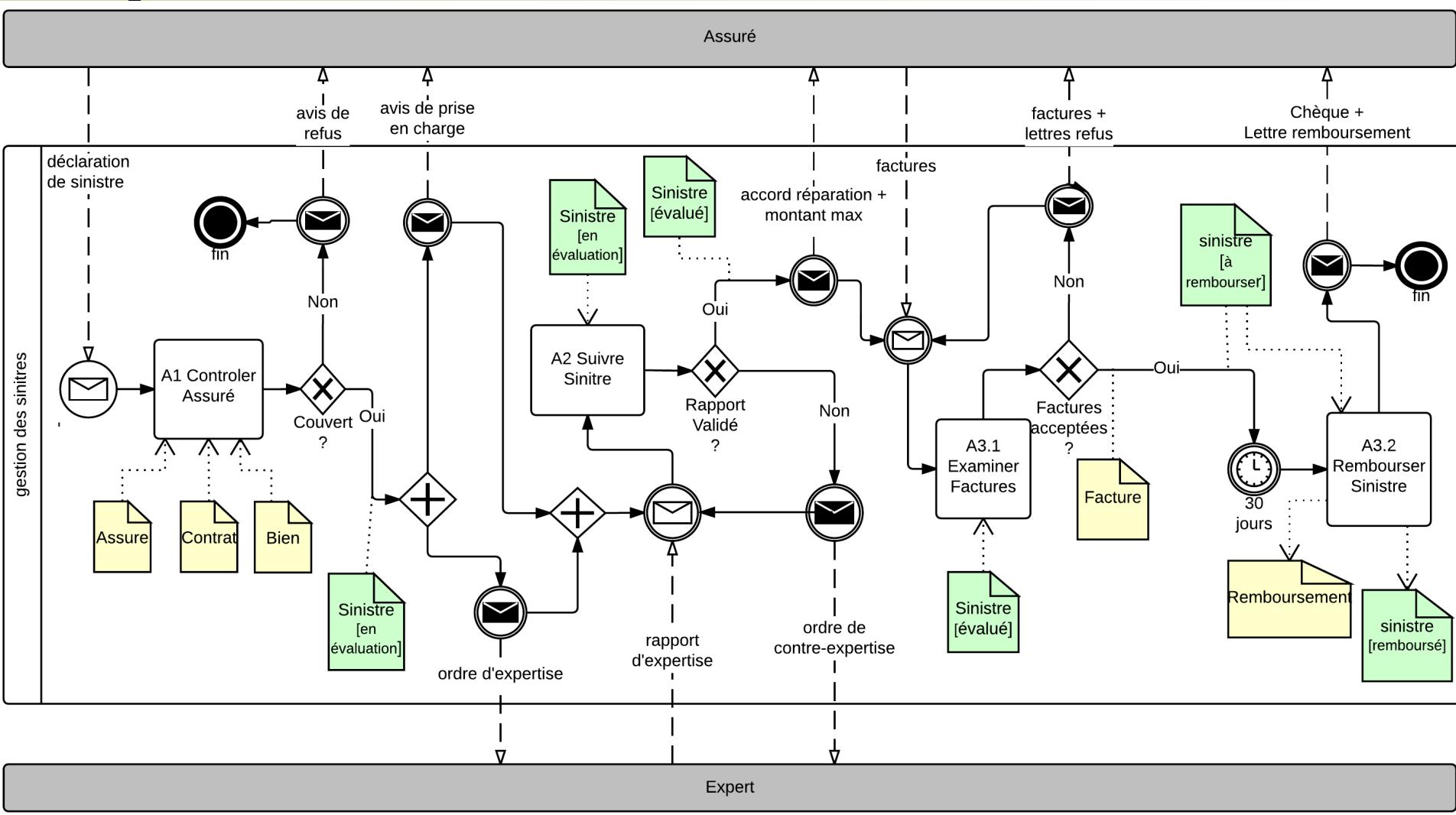
dataInputAssociation (l'entité est consultée, elle est en entrée de l'activité) – flèche de l'entité vers l'activité

dataOutputAssociation (l'entité est en sortie de l'activité – création ou modification) – flèche de l'activité vers l'entité

dataOutputAssociation (pas de flèche – même sens que le flux de séquence)

Possibilité de définir des états (state) d'entité du MCD

# Exemple : gestion des sinistres (4)

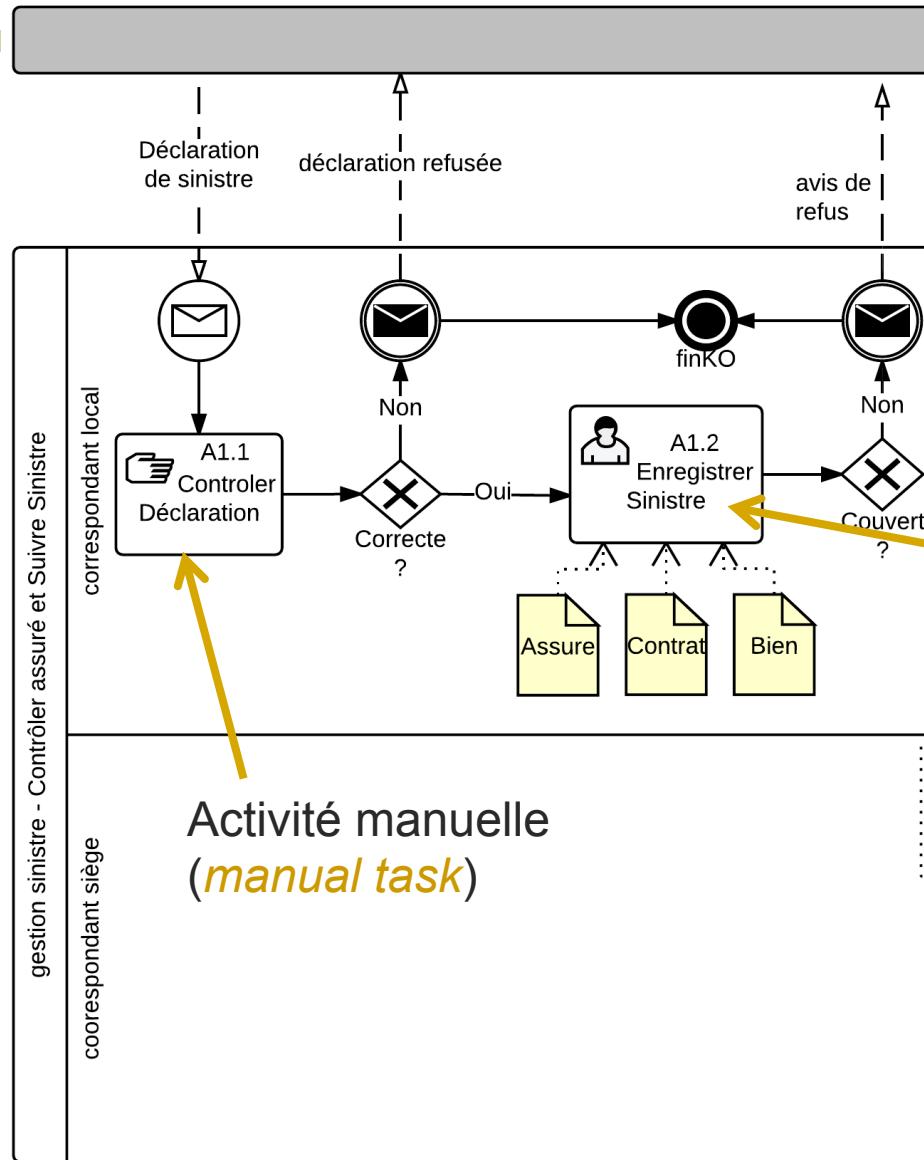


# Niveau Organisationnel

## ■ Le qui fait quoi comment....

- **RO2.** Les **agents locaux** ont pour rôle de **contrôler** que les déclarations de sinistre transmises par les assurés sont correctement remplies. En cas de déclaration de sinistre incomplète, l'agent local édite un courrier à l'assuré.
- **RO3.** Les agents locaux sont habilités à **consulter dans la base informatique** les caractéristiques des assurés et de leurs contrats pour vérifier la couverture des sinistres déclarés. Ils **saisissent** les caractéristiques du sinistre dans la base après vérification de la couverture. Ils éditent un courrier d'avis de prise en charge ou de refus du sinistre.
- **RO4.** Un dossier est suivi de bout en bout par le même **correspondant du siège**. Tous les matins, les correspondants du siège traitent les nouveaux dossiers :
  - Ils **saisissent** leur code de correspondant dans la base et font le choix d'un expert pour chaque sinistre.
  - Ils envoient un ordre d'expertise aux experts désignés.

# prise en compte de l'organisation



## *MOTA : raffinement du MCTA*

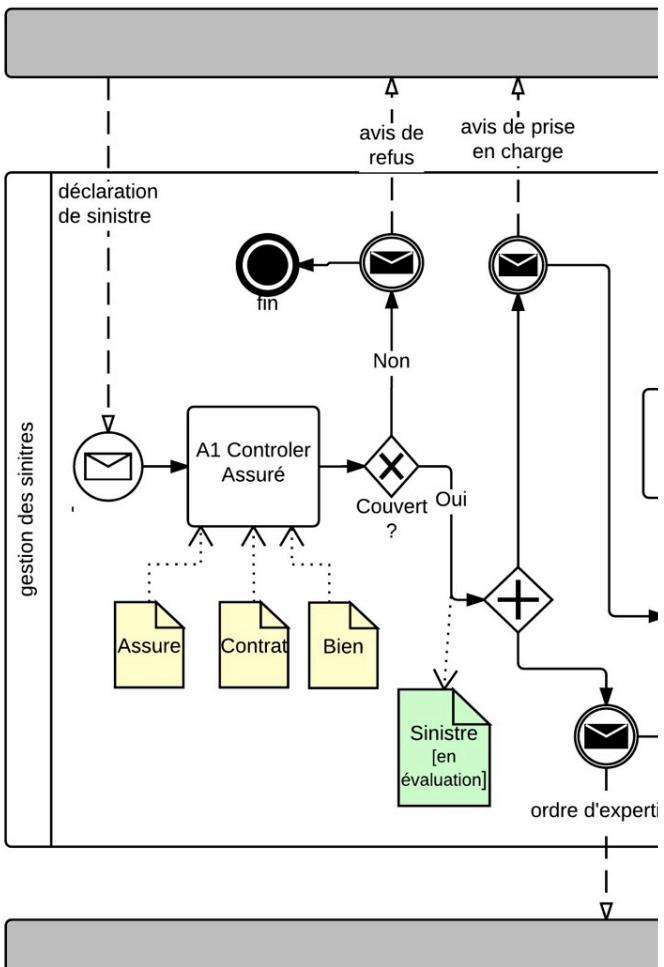
Couloir (*Lane*)  
*modélise les acteurs internes*

- Activité humaine informatisée (*user task*)  
Nécessite une interaction entre l'homme et la machine



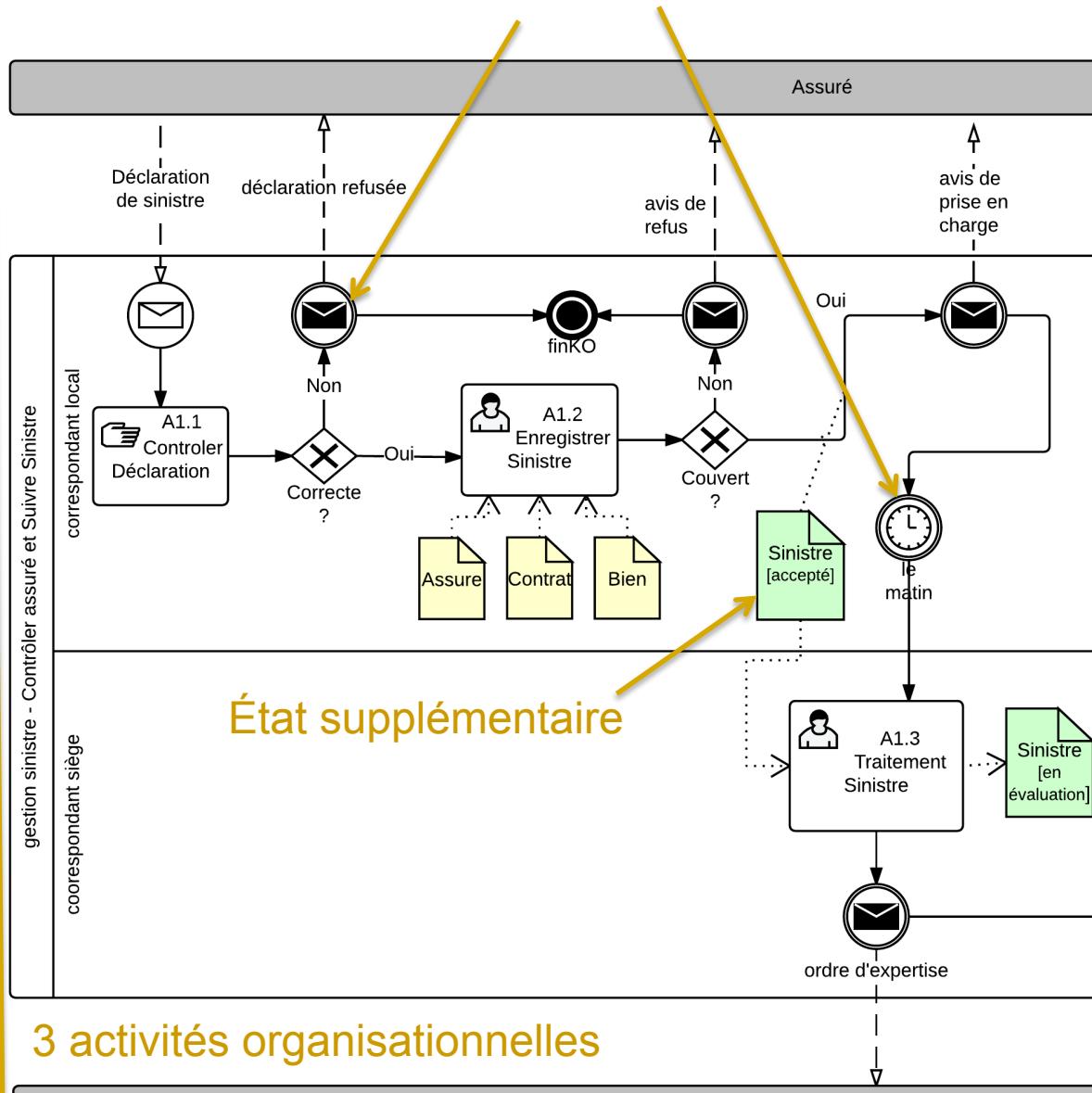
- Service (*service task*)  
Activité entièrement automatique

## conceptuel



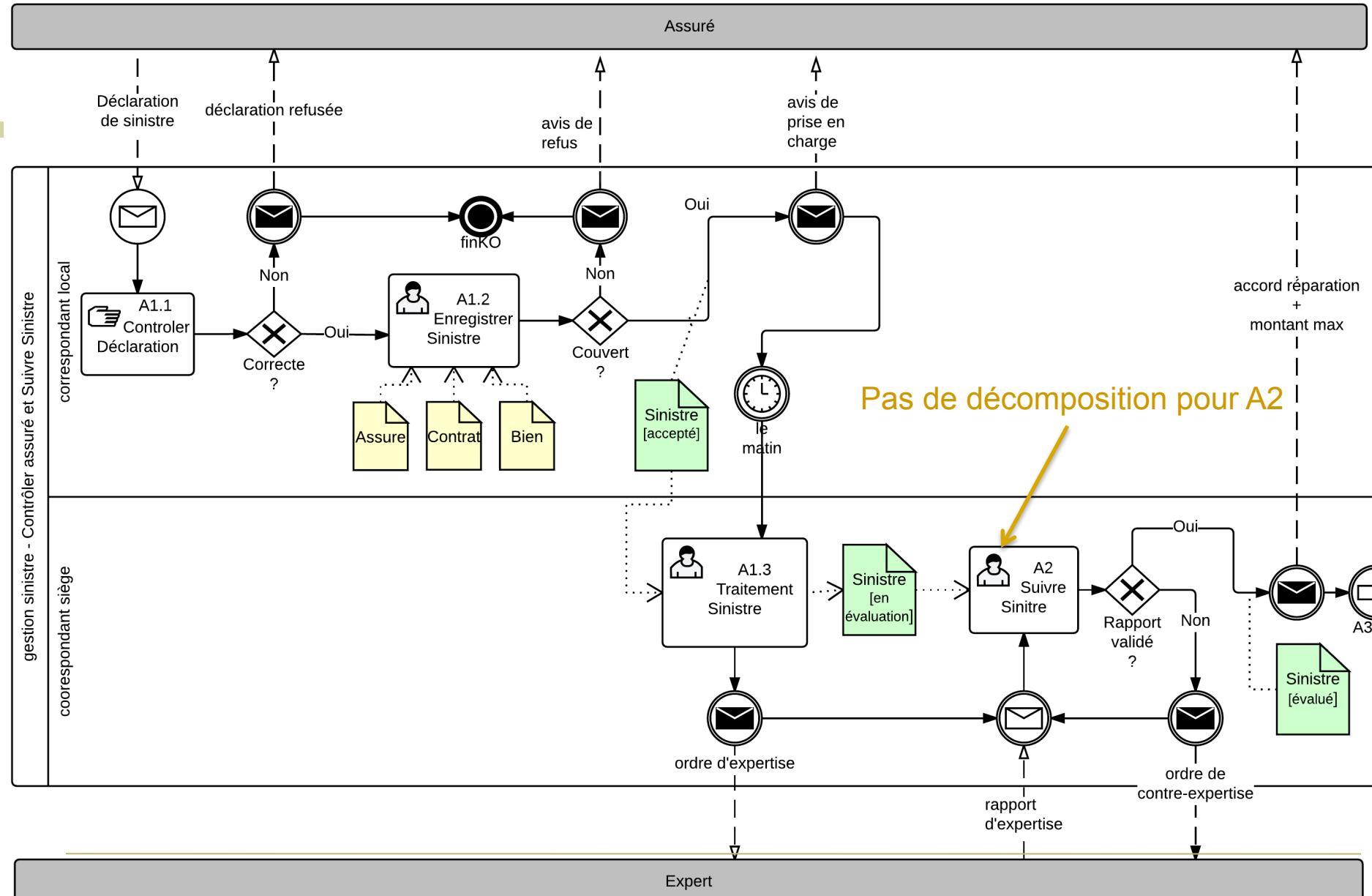
## Evènements supplémentaires

## organisationnel



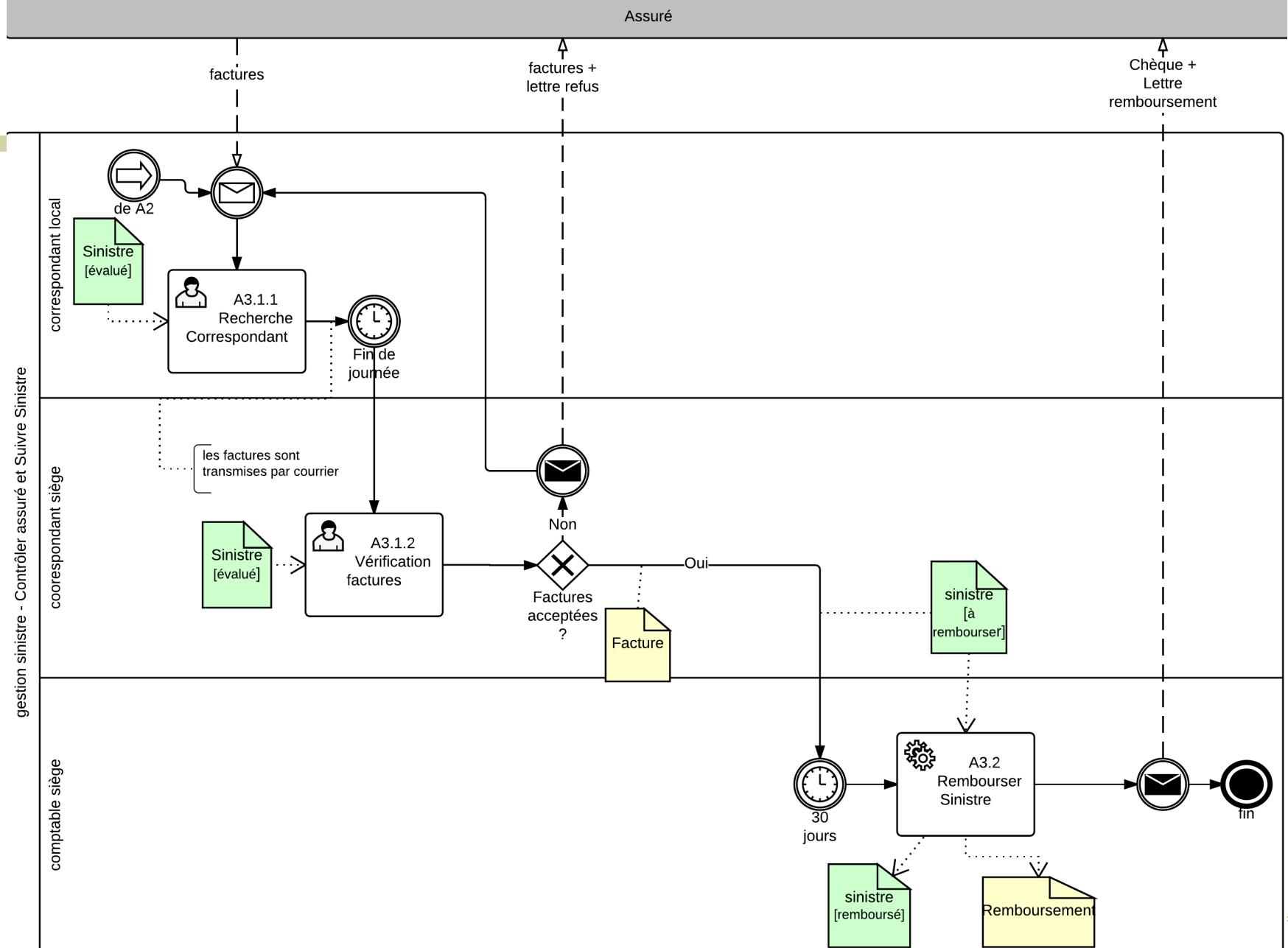
# Gestion des sinistres : A1 et A2

7



# MOTA correspondant à A3.1 et A3.2

7



## Niveau organisationnel

# Evolution des modèles de données

