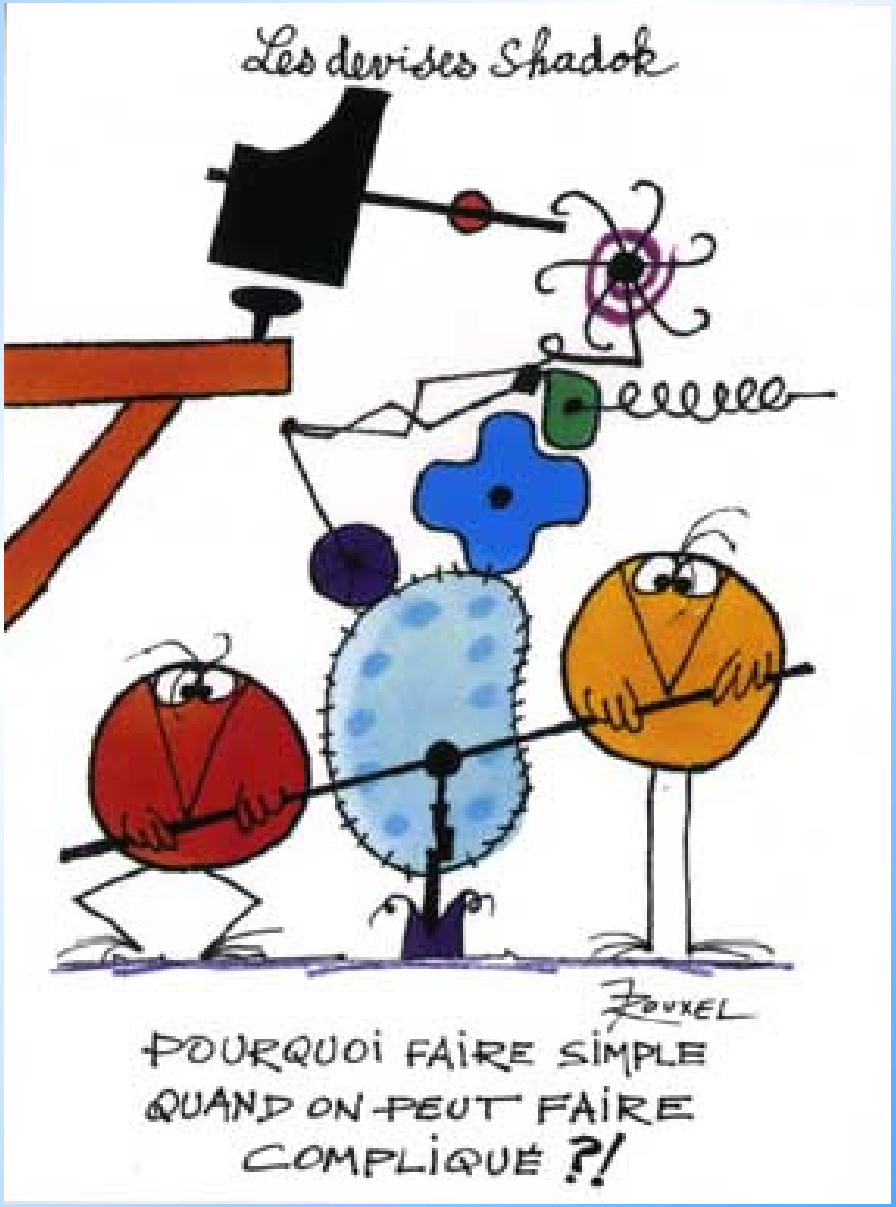
DIAGRAMME DE CAS D'UTILISATION

Description de haut niveau des fonctionnalités du système



L'expression des besoins

L'utilisateur au moment de l'expression des besoins

L'utilisateur lors de la revue d'analyse

L'utilisateur après lecture des documents de conception

L'utilisateur le jour de la livraison et installation

L'expression des besoins

L'utilisateur au moment de l'expression des besoins

L'utilisateur lors de la revue d'analyse

L'utilisateur après lecture des documents de conception

L'utilisateur le jour de la livraison et installation

SOMMAIRE

- **Introduction**
- Identification des acteurs
- Identification des cas d'utilisation
- Relations entre cas d'utilisation
- Structuration en paquetages

PROCESSUS UNIFIÉ : LES ACTIVITÉS

SPECIFICATION DES EXIGENCES

Qu'est-ce qu'ILS veulent faire avec ce logiciel ?

J'installe et J'explique

ANALYSE du domaine

Avec quoi ILS travaillent ?

DEPLOIEMENT

Comment JE vais faire ce logiciel ?

CONCEPTION de l'architecture

TESTS

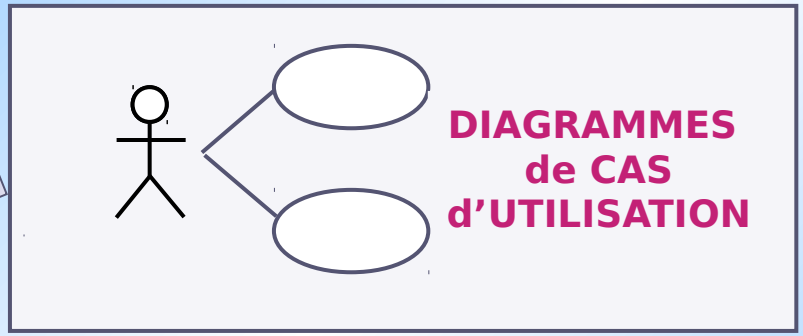
Super, ça marche !

Chic, JE programme !

IMPLEMENTATION

Diagramme de cas d'utilisation

SPECIFICATION DES BESOINS



DIAGRAMMES de CAS d'UTILISATION

ANALYSE

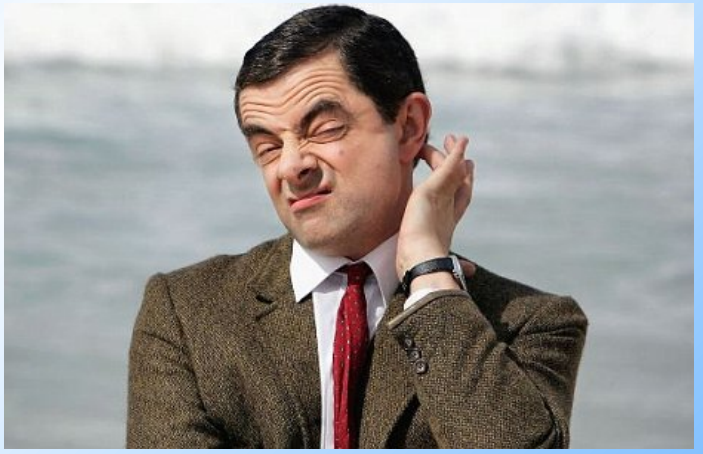
CONCEPTION de l'architecture

TESTS

IMPLEMENTATION

Première étape :

S'assurer que le logiciel va faire ce que le client souhaite



Comment comprendre ce qu'un client veut vraiment ?

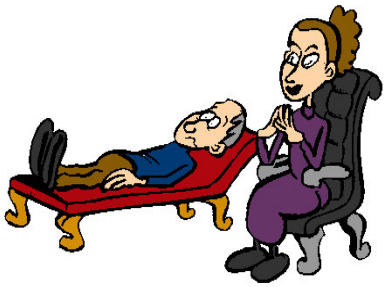
Comment s'assurer qu'un client sait vraiment ce qu'il veut ?

recueil des exigences

recueil des exigences

Satisfaire votre client en livrant avec certitude ce qu'il a demandé

Écouter le client



Laisser parler le client

Se concentrer sur ce que le système doit faire

~~Comment le système pourra le faire~~

NON !!

Le client s'attend à ce que les choses fonctionnent même si des problèmes surviennent

Il faut donc anticiper ce qui pourrait poser problème et ajouter des exigences pour régler ces éventuels problèmes

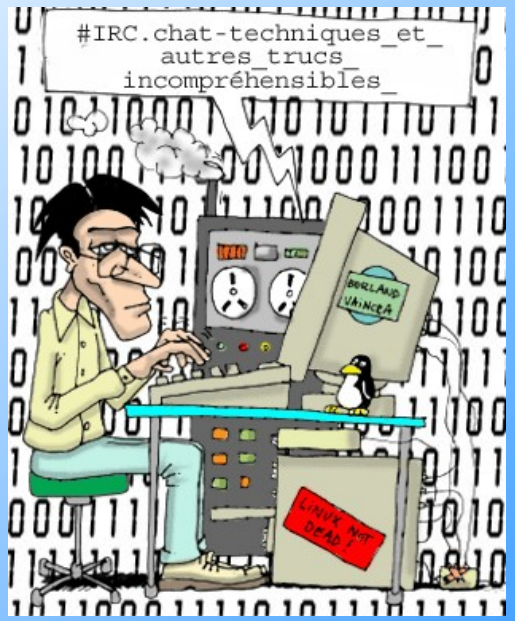
Bon ensemble d'exigences = *au-delà de ce que votre client vous dit.*

Il faut s'assurer que le système fonctionne même dans des circonstances inhabituelles ou inattendues

Un problème de communication

- Expertise, jargon du domaine
- Indécision, opinion changeant selon l'offre
- Besoins ambigus, éléments manquants

- Schémas souvent incompréhensibles pour les non initiés.
- Langages formels
- Spécifications longues et fastidieuses



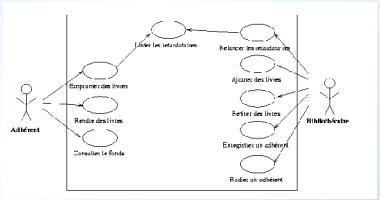
Difficultés

- Définir de QUOI les utilisateurs ont vraiment besoin :
  - Poser le bon problème,
  - Ne pas laisser les utilisateurs se mêler de réalisation,
  - Ne pas inventer des fonctions pour le plaisir.
- Bannir toute considération de réalisation lors des premières rencontres,
- Comprendre le besoin de manière globale :
  - Flot incommensurable d'informations (il faut que…)
  - Contradictions entre les utilisateurs.
- Besoins mouvants.

INTRODUCTION

- Expression préliminaire des besoins :
  - ✓modélisation par les cas d'utilisation
  - ✓ maquette d'interface homme-machine (*IHM*)

Besoins utilisateurs



Cas d'utilisation

Maquette

- Maquette graphique : montre rapidement l'aspect visuel de l'application.

INTRODUCTION

- Dans le processus de standardisation des méthodes objet ayant abouti au formalisme UML, le concept des cas d'utilisation a été **repris de la méthode OOSE** de **Ivar Jacobson** (*un des pères de UML*).

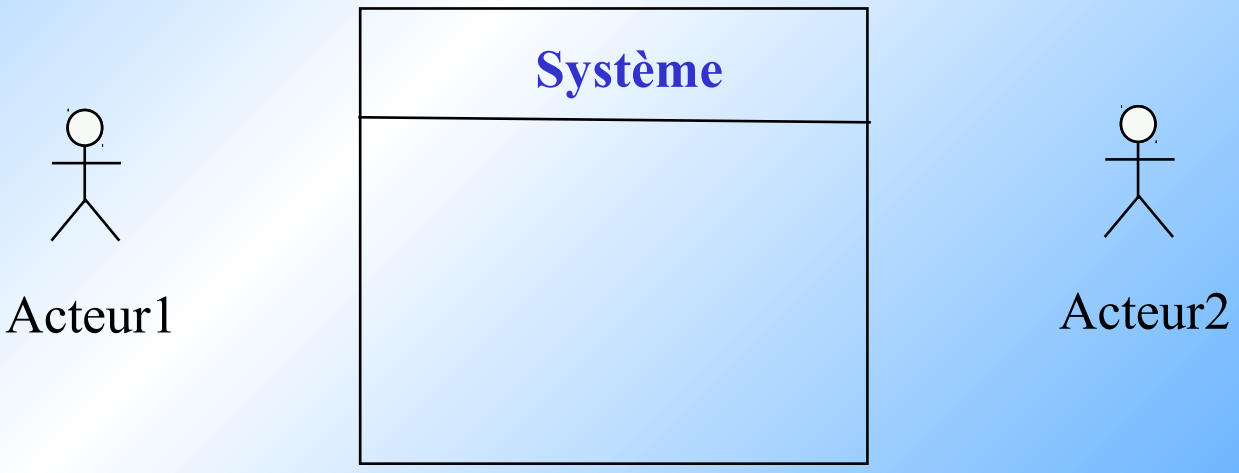
- **But des cas d'utilisation** :

  - effectuer une bonne **délimitation du système**
  - améliorer la **compréhension de son fonctionnement**.

INTRODUCTION

- Pour permettre une **bonne délimitation** du système :
  - bien séparer les éléments constitutifs de l'application,
  - des éléments extérieurs à l'application.

Les acteurs représentent cette frontière.



INTRODUCTION

Objectif des cas d'utilisation

- Définir un **comportement** d'une partie du système **sans révéler la structure interne** du système.

- Un cas d'utilisation représente du point de vue fonctionnel une **fonction essentielle** du système.

INTRODUCTION

Qu'est ce qu'un cas d'utilisation ?

- Représente une interaction typique entre un utilisateur et un système informatique
- Identifie une fonctionnalité visible de l'utilisateur
- S'occupe d'un objectif élémentaire de l'utilisateur

EXEMPLE

Machine distributeur boissons

Se servir une boisson

Gérer la machine

Consommateur

Employé

EXEMPLE

Les cas d'utilisation de la machine à distribuer des boissons :

- **Cas 1 : Se servir une boisson**
  - Le consommateur introduit des pièces
  - Le consommateur effectue une sélection de boisson
  - Le consommateur retire le gobelet contenant la boisson

- **Cas 2 : Gérer la machine**
  - L'employé alimente la machine en ingrédients de base et en gobelets
  - L'employé entretient la machine (*nettoyage des buses d'alimentation*)
  - L'employé récupère l'argent et met de la monnaie

Démarche

- Identifier les acteurs
- Identifier les cas d'utilisation
- Structurer les cas d'utilisation en paquetages
- Ajouter les relations entre cas d'utilisation

SOMMAIRE

- Introduction
- **Identification des acteurs**
- Identification des cas d'utilisation
- Relations entre cas d'utilisation
- Structuration en paquetages

DÉFINITION

- Un **acteur** représente un **rôle** joué par une entité externe (*utilisateur humain, dispositif matériel ou autre système*) qui interagit **directement** avec le système étudié.

Consommateur

messages

Système

Machine distributeur boisson

- Un acteur peut **consulter** et/ou **modifier** directement l'état du système, en émettant et/ou recevant des messages éventuellement porteurs de données.

TYPES DES ACTEURS

Les acteurs peuvent être de *3* **types** :

- **humains** : ce sont des utilisateurs du logiciel à travers son interface graphique par exemple.

- **logiciels** : ce sont des logiciels déjà disponibles qui communiquent avec le système grâce à une interface logicielle (*une Application Programming Interface par exemple*).

- **matériels, robots et automates** qui exploitent les données du système ou qui sont pilotés par le système.

Acteurs à l'extérieur du système

Il faut vérifier que les acteurs se trouvent bien ***à l'extérieur*** du système !

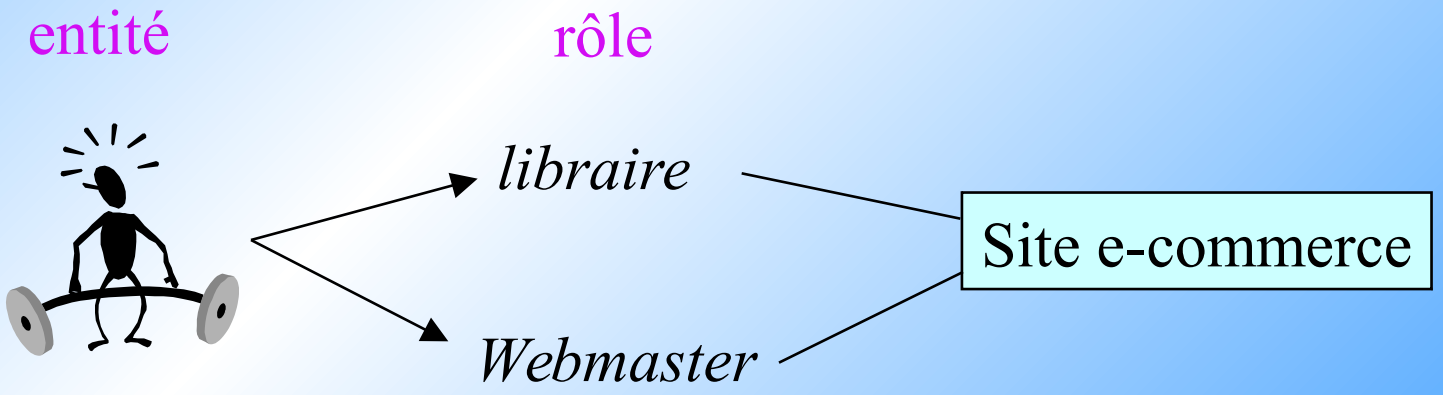
Une erreur fréquente consiste à répertorier des acteurs qui correspondent en fait à des composants du système étudié, voire même à de futures classes.

RÔLE ET ENTITÉ CONCRÈTE

Ne confondez pas rôle et entité concrète.

- Une **même entité concrète** peut jouer successivement **différents rôles** par rapport au système étudié, et être modélisée par plusieurs acteurs.

Exemple

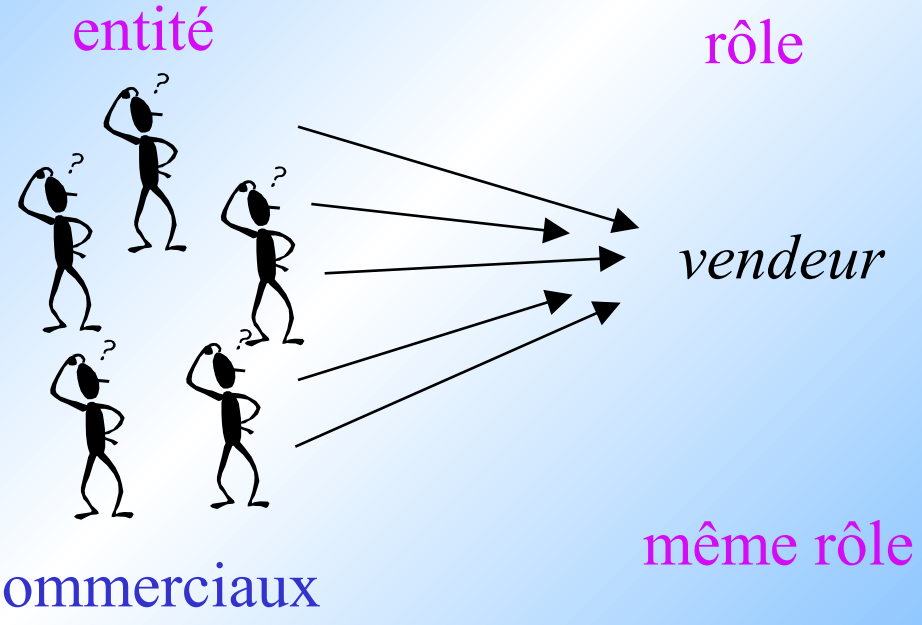


2 acteurs distincts (*2 profils différents*)

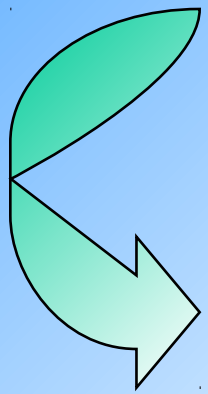
RÔLE ET ENTITÉ CONCRÈTE

Ne confondez pas rôle et entité concrète.

- Réciproquement, le **même rôle** peut être tenu simultanément par **plusieurs entités concrètes**, qui seront alors modélisées par le même acteur.



Rôle et entité concrète



On doit penser en termes de rôles et non de personnes ou d'intitulés de fonctions.

Acteurs physiques / logiques

Éliminez les acteurs physiques au profit des logiques

L'acteur est celui qui bénéficie de l'utilisation du système. Il a une autonomie de décision. Cela ne doit pas se réduire à un simple dispositif mécanique passif.
Cette règle simple permet de s'affranchir dans un premier temps des technologies d'interface et de se concentrer sur les acteurs *"métier"*, nettement plus stables.

Exemple

- Les terminaux informatiques (*écrans, claviers, imprimantes,etc*.) des différents employés d'une entreprise, à la place des différents profils identifiés.

COMMENT LES REPRÉSENTER ?

1 La **représentation** graphique **standard** est :
- l'icône appelée *stick man*,
- avec le nom de l'acteur sous le dessin.

Client

2 On peut également le représenter :
- sous la forme rectangulaire d'une classe
- avec le mot-clé <<actor>>.

<<actor>>

SI banque

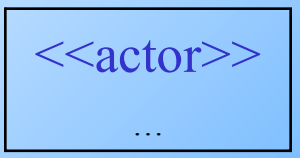
COMMENT LES REPRÉSENTER ?

3 Une troisième représentation :
- intermédiaire entre les deux premières.

Bonne recommandation

Acteurs humains →

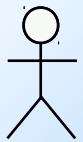
Acteurs non-humains →



NOTATION DES ACTEURS

Icônes proposées par Ivar Jacobson (*un des pères d'UML*) :

Acteur externe à l'entreprise →



Acteur métier

Acteur interne à l'entreprise →

Travailleur métier

Intérêt : améliorer la lisibilité des diagrammes de cas d'utilisation.

TYPOLOGIE DES ACTEURS

- **Acteur principal** : celui pour qui le cas d'utilisation produit un résultat observable.

- **Acteur secondaire** : les autres participants du cas d'utilisation.

TYPOLOGIE DES ACTEURS

Acteurs principaux

- répondent à la question :
  - A qui va servir le système qu'on va développer ?
  - Qui le système doit-il aider ?

Ils sont la raison même de l'existence de ce système.

Celui pour qui le cas d'utilisation produit la **plus-value métier**. L'acteur principal est la plupart du temps le déclencheur du cas d'utilisation.

TYPOLOGIE DES ACTEURS

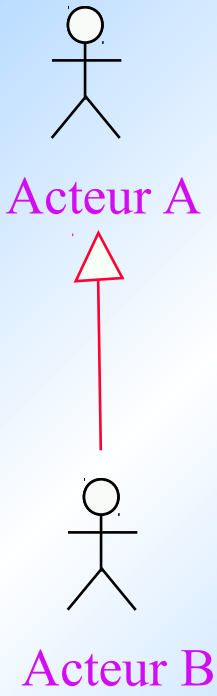
Acteurs secondaires

- Ils sont sollicités pour des **informations complémentaires**, ils peuvent uniquement consulter ou informer le système lors de l'exécution du cas d'utilisation.

Ce sont ceux qui paramètrent le système et qui lui fournissent toutes les informations nécessaires à son bon fonctionnement pour les acteurs principaux. Ils sont typiquement sollicités par le système pour obtenir des informations complémentaires.

Relations entre acteurs

- La seule relation possible entre deux acteurs est la généralisation.



L'acteur *B* est un acteur *A* avec un pouvoir supplémentaire.

ÉTUDE DE CAS

Étude d'un Guichet Automatique de Banque (GAB)

Le GAB offre les services suivants :

- Distribution d'argent à tout porteur de carte de crédit (*carte Visa ou carte de la banque*), via un lecteur de carte et un distributeur de billets.

- Consultation de solde de compte, dépôt en numéraire et dépôt de chèques pour les clients de la banque porteurs d'une carte de crédit de la banque.

ÉTUDE DE CAS

Étude d'un Guichet Automatique de Banque (GAB)

- Toutes les transactions sont sécurisées :

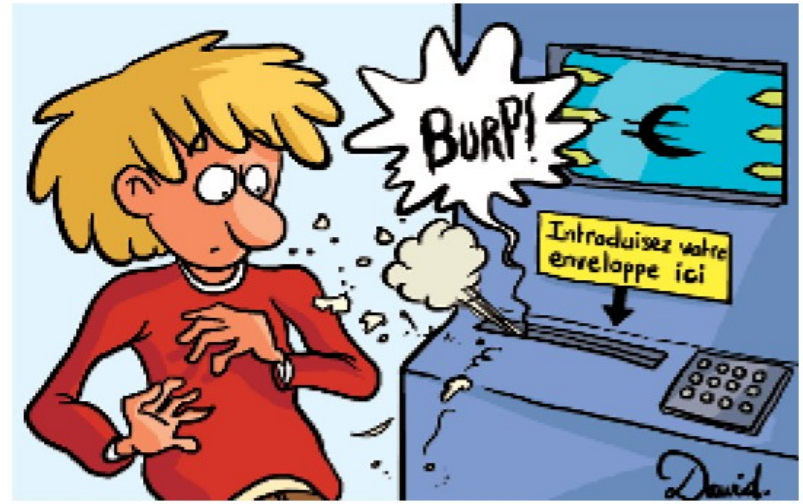
  ↳ Le système d'autorisation Visa, pour les transactions de retrait effectuées avec une carte Visa.

  ↳ Le système d'information de la banque, pour autoriser toutes les transactions effectuées par un client avec sa carte de la banque, mais également pour accéder au solde de ses comptes.

ÉTUDE DE CAS

Étude d'un Guichet Automatique de Banque (GAB)

- Il est nécessaire de recharger le distributeur, récupérer les cartes avalées, etc.



ÉTUDE DE CAS

Identifier les acteurs du GAB

SOLUTION

Phrase 1

Porteur de carte de crédit

Que pensez-vous des éléments suivants ?

- lecteur de carte
- distributeur de billets

SOLUTION

- Le lecteur de carte et le distributeur de billets font partie du GAB.

- L'identification des acteurs oblige à fixer précisément la frontière entre le système et son environnement.

- Si on restreint l'étude au système de contrôle-commande des éléments physiques du GAB, le lecteur de carte et le distributeur de billets deviennent alors des acteurs.

ÉTUDE DE CAS

La carte bancaire est-elle un acteur ?

SOLUTION

- La carte bancaire est bien externe au GAB et elle interagit avec lui.

- On applique le principe : **éliminer autant que possible les acteurs physiques au profit des acteurs logiques**.

- C'est le porteur de carte qui retire de l'argent pour le dépenser ensuite, pas la carte !

carte bancaire : pas un acteur

SOLUTION

Phrase 2

- Client de la banque

SOLUTION

Phrase 3

- Le système d'autorisation Visa.
- Le système d'information de la banque.

SOLUTION

Phrase 4

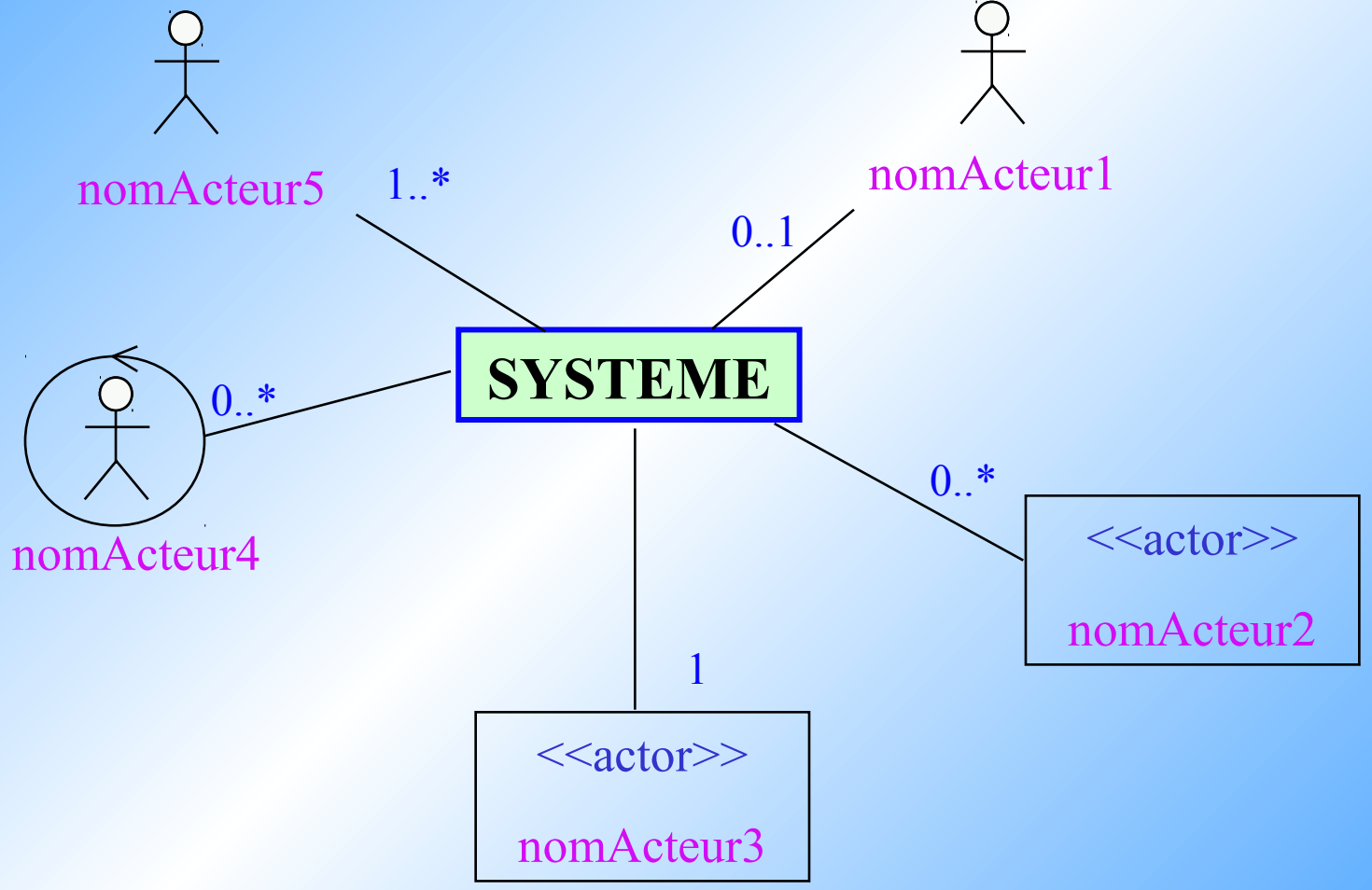
- Opérateur de maintenance.

ÉTUDE DE CAS

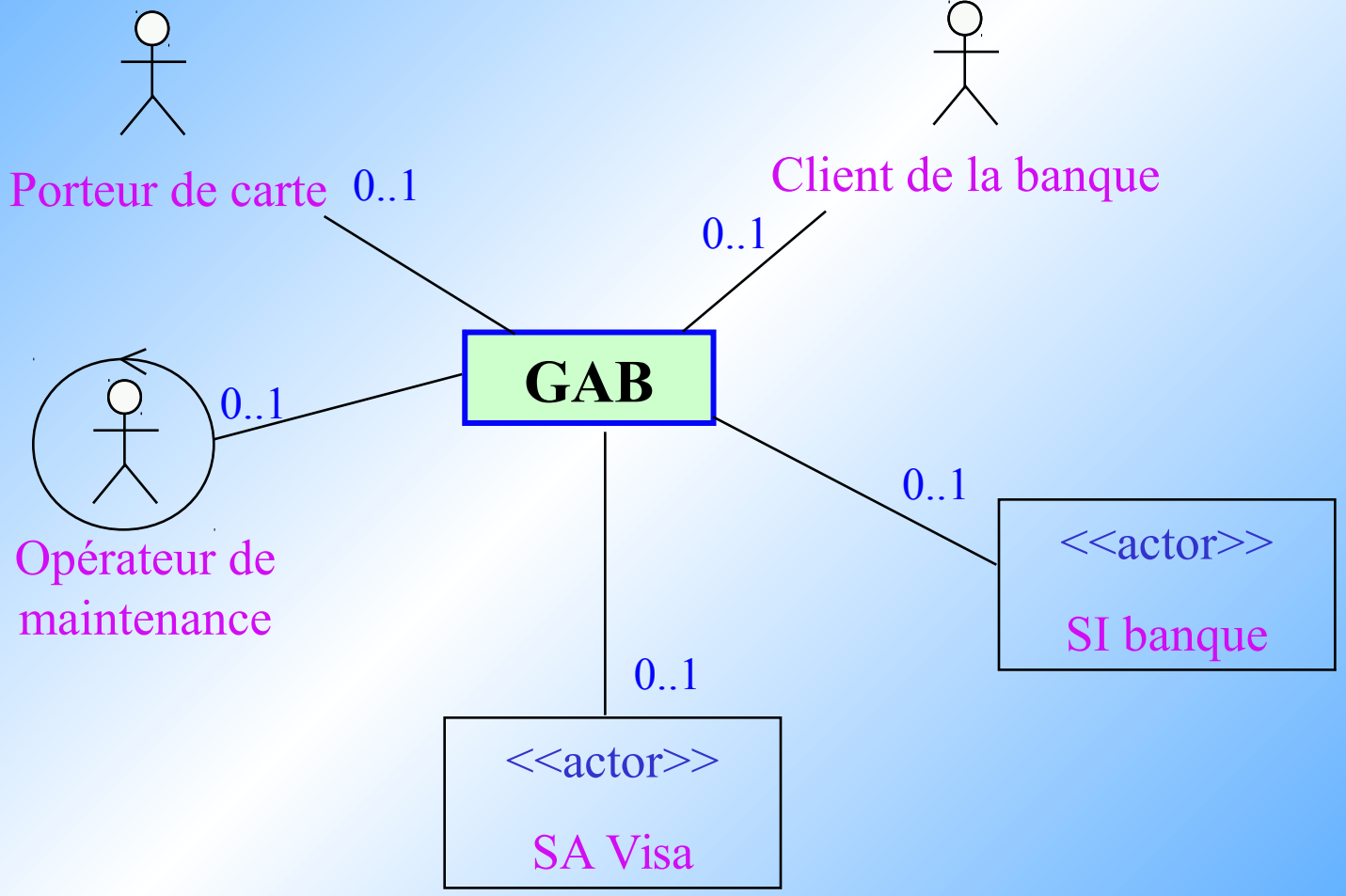
Réaliser un diagramme de contexte statique

- C'est un diagramme de classes dans lequel chaque acteur est relié à une classe centrale unique représentant le système, par une association (*ce qui permet de spécifier le nombre d'instances d'acteurs connectés au système à un moment donné*).

EXEMPLE


nomActeur5
1..*
nomActeur1
0..1
SYSTEME
0..*
nomActeur4
0..*
<<actor>>
nomActeur2
1
<<actor>>
nomActeur3

SOLUTION

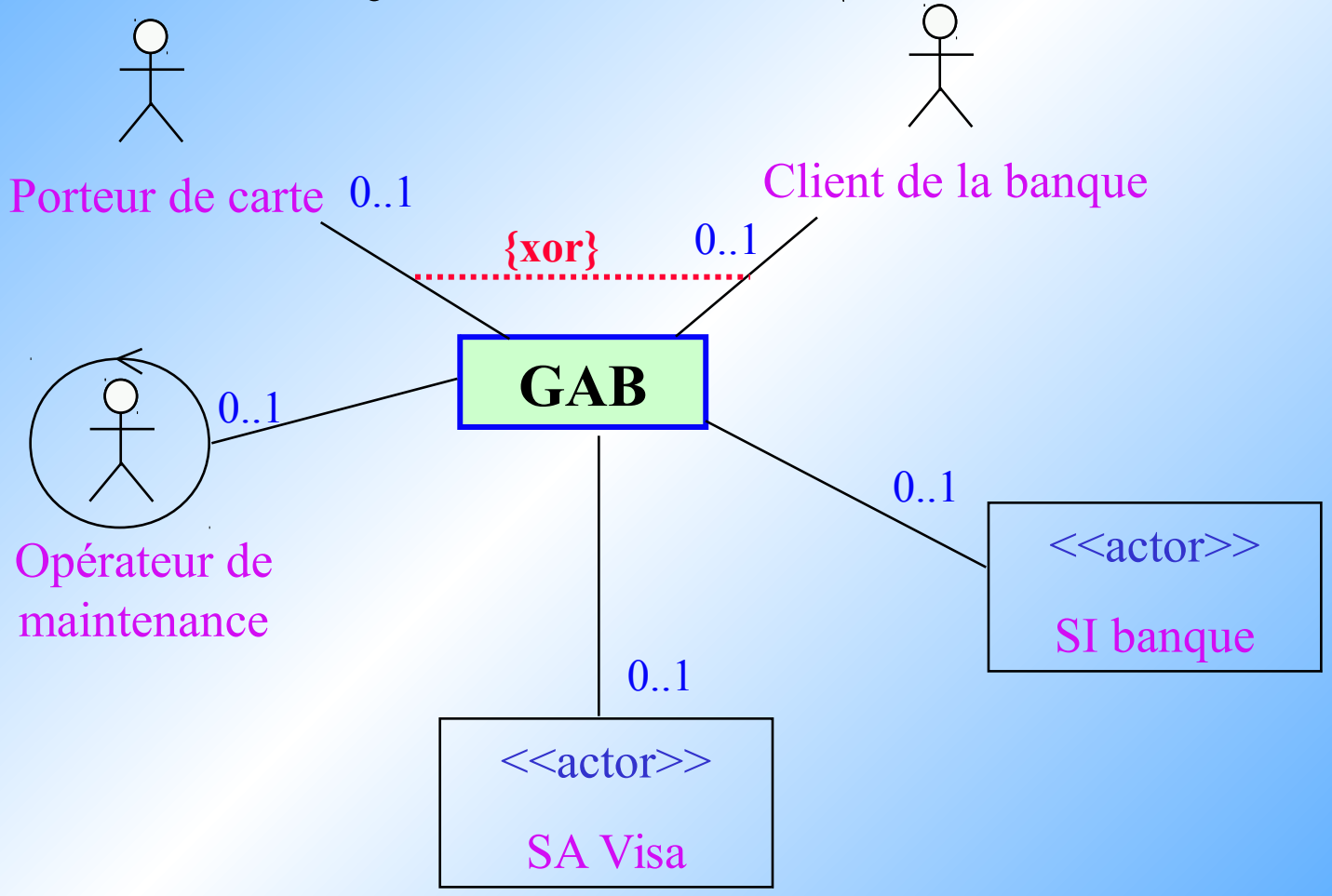

Porteur de carte
0..1
Client de la banque
0..1
GAB
0..1
Opérateur de maintenance
0..1
<<actor>>
SI banque
0..1
<<actor>>
SA Visa

SOLUTION

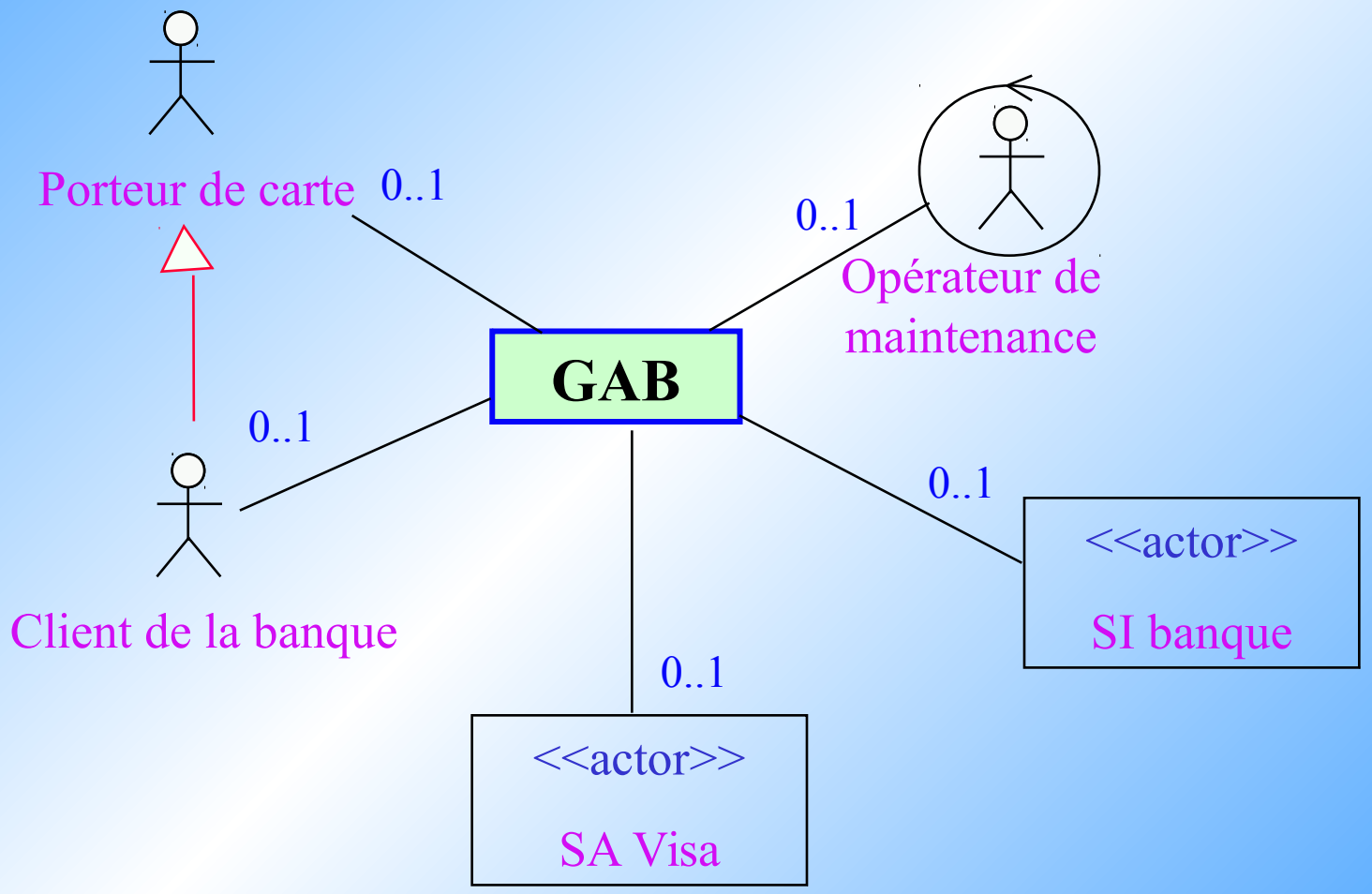
- Le GAB est un système mono-utilisateur : à tout instant, il n'y a qu'une instance au maximum de chaque acteur connectée au système.

Affiner les associations "*Porteur de carte* " et "*Client de la banque* ".

SOLUTION


Porteur de carte
0..1
Client de la banque
{xor}
0..1
GAB
0..1
Opérateur de maintenance
0..1
<<actor>>
SI banque
0..1
<<actor>>
SA Visa

SOLUTION


Porteur de carte
0..1
0..1
Opérateur de maintenance
GAB
0..1
0..1
Client de la banque
<<actor>>
SI banque
0..1
<<actor>>
SA Visa

Acteurs du site Web

Rechercher les acteurs humains pour le site Web *www.jeBouquine.com*.

SOLUTION

Les acteurs humains :

- L'**internaute** : personne qui visite le site pour rechercher des ouvrages et éventuellement passer des commandes. Il s'agit bien sûr de l'acteur le plus important, celui pour lequel le site existe !

- Le **Webmaster** : rôle des employés qui ont en charge le bon fonctionnement et la maintenance du site Web.

- Le **Service client** : rôle des employés qui s'occupent du suivi des commandes clients.

- Le **Libraire** : rôle des employés qui sont responsables du contenu rédactionnel du site.

Acteurs du site Web

Rechercher les autres acteurs pour le site Web *www.jeBouquine.com*.

SOLUTION

On va prendre en compte les systèmes informatiques de la société ***jeBouquine*** connectés au site Web :

- Le service **Nouveautés** (*alimente la base avec tous les nouveaux ouvrages*),

- Le service **Gestion des stocks** (*met à jour les données concernant le prix et l'état du stock des livres du catalogue*).

- Le service **Paiement sécurisé**

Les deux premières sources externes seront automatiquement chargées dans la base de données de façon périodique.

Acteurs du site Web

Représenter les acteurs avec le formalisme adéquat

SOLUTION

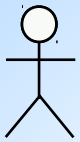
redondant

Nouveautés

redondant

Gestion des stocks

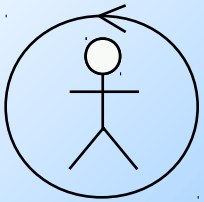
1 seul acteur externe



Internaute

Webmaster

Libraire



Service Clients

<<actor>>

Nouveautés

<<actor>>

Gestion des stocks

<<actor>>

Paiement sécurisé

SOMMAIRE

- Introduction
- Identification des acteurs
- **Identification des cas d'utilisation**
- Relations entre cas d'utilisation
- Structuration en paquetages

DÉFINITION

Un cas d'utilisation :

- un **ensemble de séquences d'actions**
  - réalisées par le système et
  - qui **produisent un résultat** observable
  - intéressant pour un acteur particulier.

- **modélise un service** rendu par le système.
  - Il exprime les interactions acteurs/système.

DÉFINITION

Un cas d'utilisation :

- spécifie un comportement attendu du système
  - considéré comme un tout,
  - sans imposer le mode de réalisation de ce comportement.

Il permet de décrire *ce que* le futur système devra faire, sans spécifier *comment* il le fera.

COMMENT LES IDENTIFIER ?

- L'ensemble des cas d'utilisation :

  ↳ doit décrire exhaustivement les exigences fonctionnelles du système.

- Chaque cas d'utilisation correspond donc
  - ↳ à une fonction métier du système,
  - ↳ selon le point de vue d'un de ses acteurs.

COMMENT LES IDENTIFIER ?

Pour chaque acteur, il faut :

- rechercher les différentes intentions *"métier"* selon lesquelles il utilise le système.

- déterminer dans le cahier des charges les services fonctionnels attendus du système.

NOM DES CAS D'UTILISATION

Verbe à l'infinitif suivi d'un complément

Utiliser le point de vue de l'acteur et non pas celui du système.

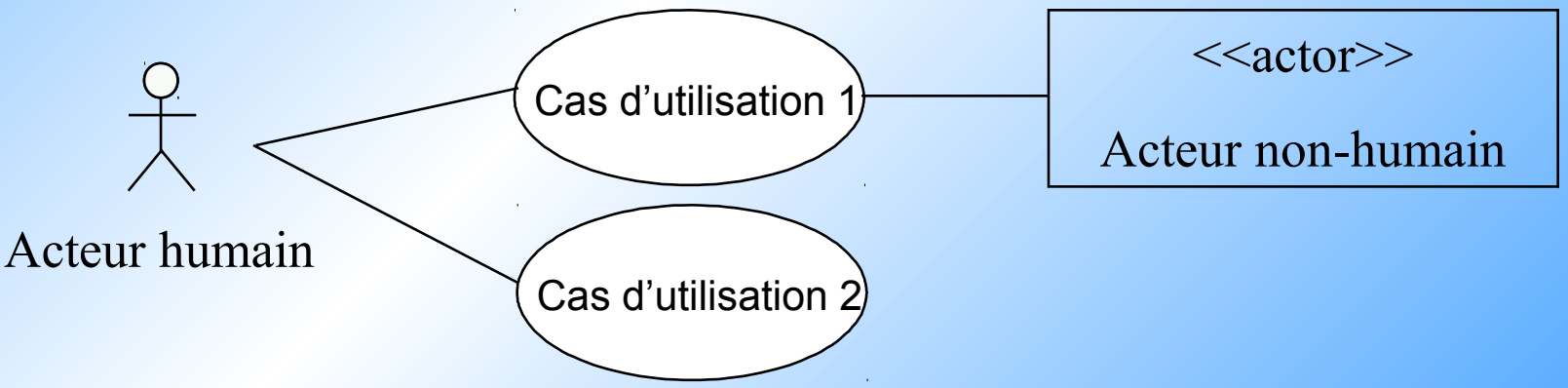
Exemple

Cas d'utilisation d'un distributeur de billets par l'acteur *"Porteur de CB"* doit être intitulé :
- *"Retirer de l'argent"* (*point de vue de l'acteur*),
- et non *"Distribuer de l'argent"* (*point de vue du système*).

COMMENT LES REPRÉSENTER ?

Le diagramme de cas d'utilisation est :

- un schéma qui montre les cas d'utilisation (*ovales*)
- reliés par des associations (*lignes*) à leurs acteurs.

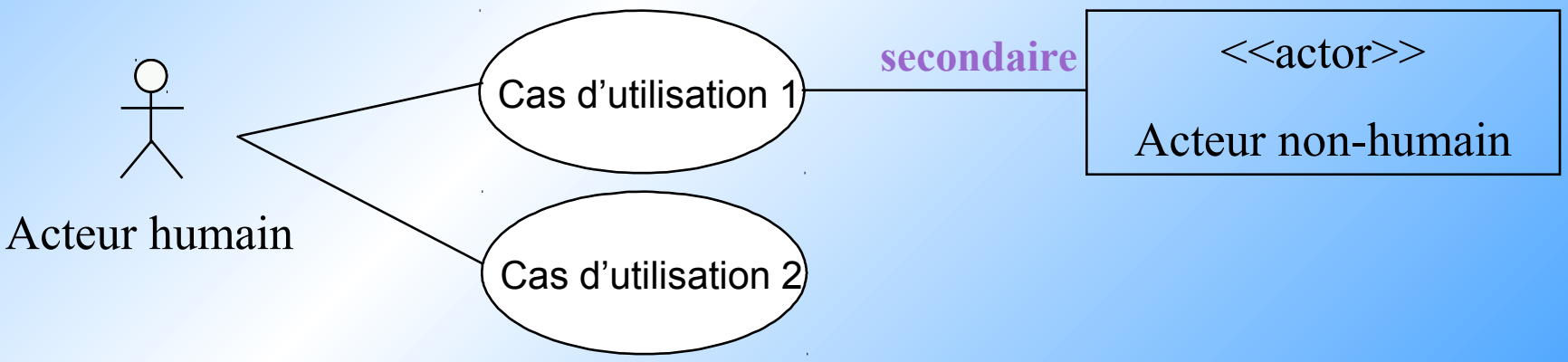


COMMENT LES REPRÉSENTER ?

Conventions recommandées

- Par défaut, le rôle d'un acteur est principal.

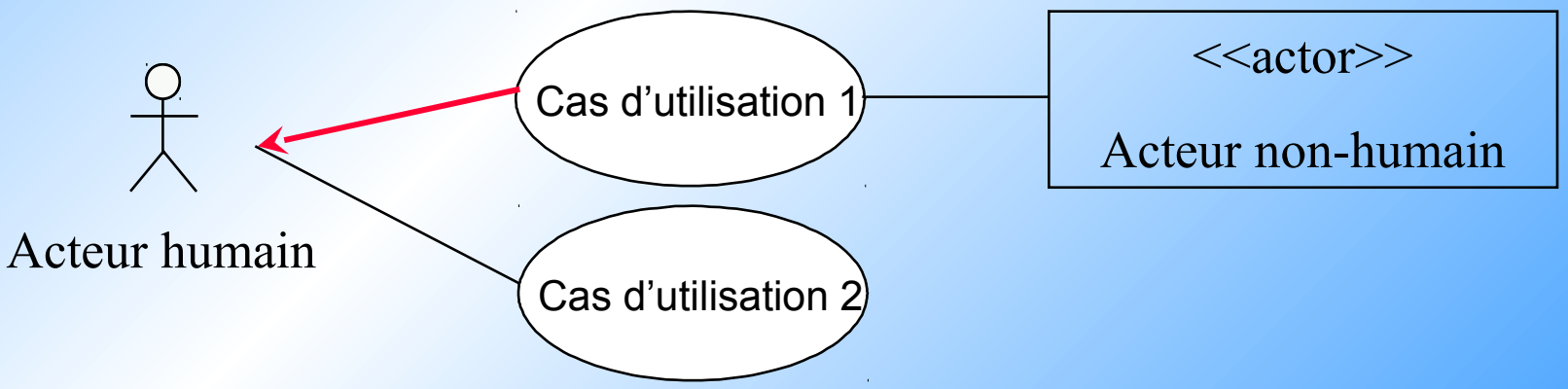
- Si ce n'est pas le cas, indiquez explicitement que le rôle est secondaire sur l'association, du côté de l'acteur.



COMMENT LES REPRÉSENTER ?

Conventions recommandées

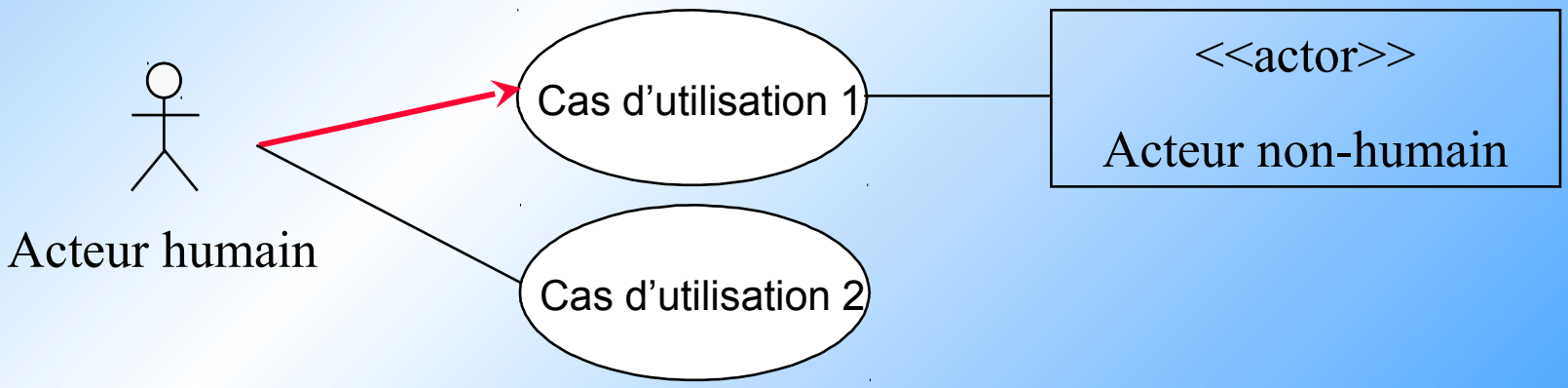
- Si un acteur a pour rôle unique de **consommer des informations** du système sans modifier l'état de celui-ci au niveau métier, représentez cette particularité en ajoutant une flèche vers l'acteur sur son association avec le cas d'utilisation.



COMMENT LES REPRÉSENTER ?

Conventions recommandées

- Si un acteur a pour rôle unique de **fournir des informations** au système sans en recevoir, représentez cette particularité en ajoutant sur l'association une flèche vers le cas d'utilisation.



OBJECTIF DES CAS D'UTILISATION

- dialoguer avec le client,
- analyser les besoins métier,
- disposer d'un support d'analyse,
- aider à démarrer l'analyse orientée objet en identifiant les classes candidates.

Les cas d'utilisation ne constituent pas une fin en soi.

LIMITATION DES CAS D'UTILISATION

Limiter à 20 le nombre des cas d'utilisation

Le niveau de granularité des cas d'utilisation étant très variable, cette limite arbitraire oblige à ne pas se poser trop de questions philosophiques et à rester synthétique.

Il est ainsi courant d'avoir une quinzaine de cas d'utilisation, comprenant chacun une dizaine de scénarios.
On peut considérer ces ordres de grandeur comme une limite pratique qui peut aider à mieux situer la frontière entre cas d'utilisation et scénario.

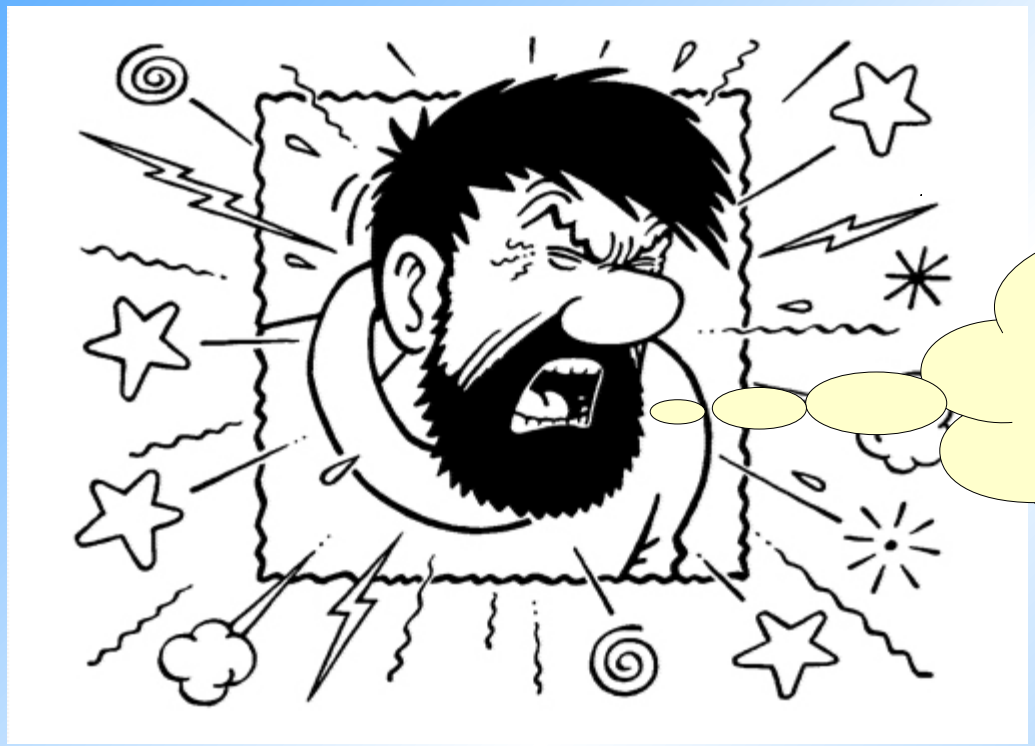
Une fois les cas d'utilisation identifiés, il faut encore les décrire !

REMARQUE

Pas de notion temporelle

- Il ne doit pas y avoir de notion temporelle dans un diagramme de cas d'utilisation.

- Il ne faut pas se dire que l'acteur fait ceci,
  puis le système lui répond cela,
  ce qui implique une réaction de l'acteur,
  et ainsi de suite.



Pas de notion temporelle

ÉTUDE DE CAS BANCAIRE

Établir une liste des cas d'utilisation par acteur

SOLUTION

Porteur de carte

- Retirer de l'argent

Client de la banque

- Retirer de l'argent
- Consulter le solde d'un ou plusieurs comptes
- Déposer du numéraire
- Déposer des chèques

SOLUTION

Opérateur de maintenance

- Recharger le distributeur
- Récupérer les cartes avalées
- Récupérer les chèques déposés

SOLUTION

Système d'autorisation Visa

- néant

Système d'information de la banque

- néant

Tous les acteurs n'utilisent pas forcément le système !

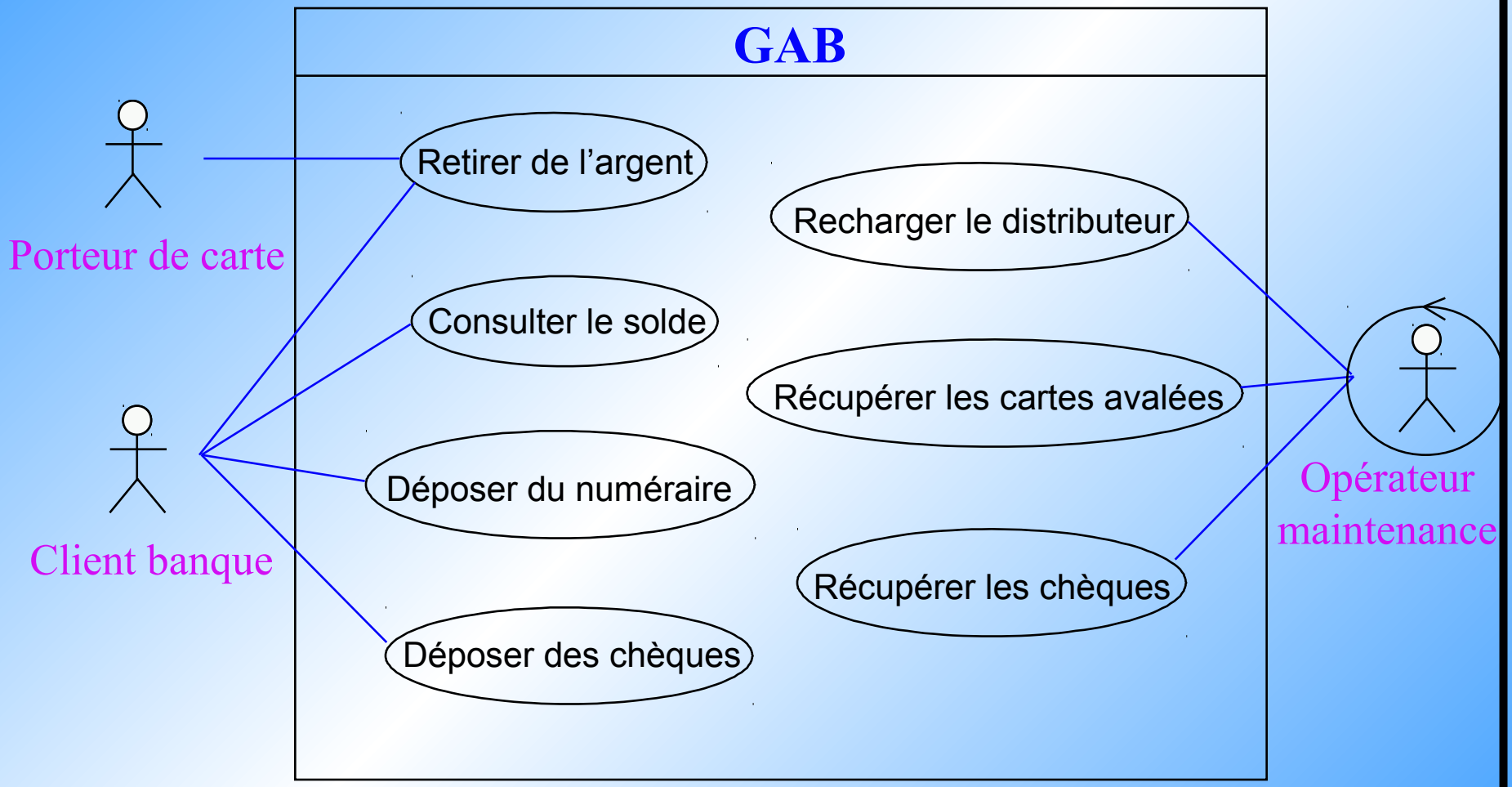
ÉTUDE DE CAS

Réaliser le diagramme de cas d'utilisation

en se concentrant sur les acteurs suivants :

Porteur de carte
Client de la banque
Opérateur de maintenance

SOLUTION

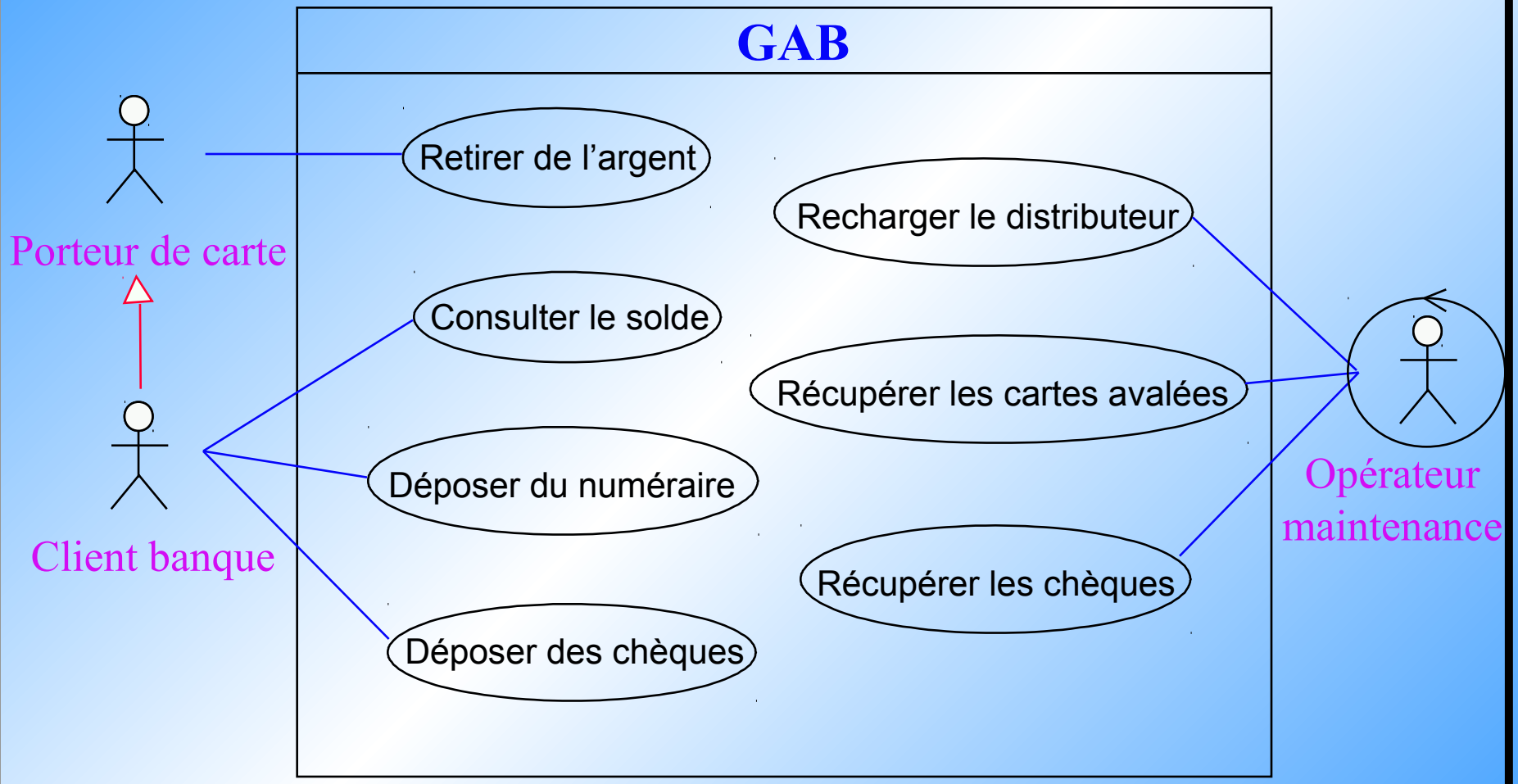

GAB
Retirer de l'argent
Consulter le solde
Déposer du numéraire
Déposer des chèques
Recharger le distributeur
Récupérer les cartes avalées
Récupérer les chèques
Porteur de carte
Client banque
Opérateur maintenance

ÉTUDE DE CAS

Proposer une autre version

en utilisant la relation de spécialisation / généralisation
entre les acteurs "*Porteur de carte*" et "*Client de la banque*"

SOLUTION


GAB
Retirer de l'argent
Recharger le distributeur
Consulter le solde
Récupérer les cartes avalées
Déposer du numéraire
Récupérer les chèques
Déposer des chèques
Porteur de carte
Client banque
Opérateur maintenance

ÉTUDE DE CAS

Ajouter les acteurs secondaires

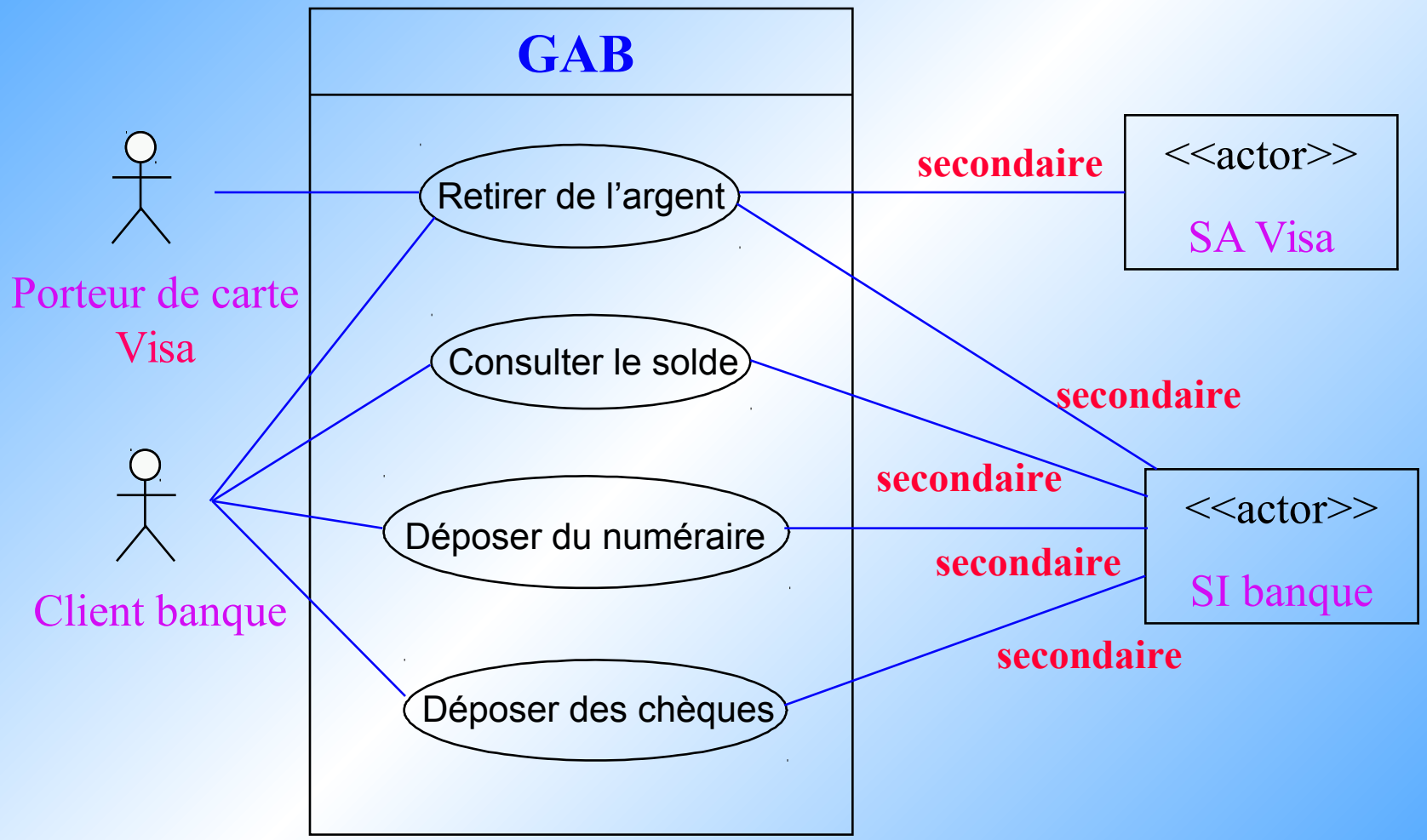
éliminer du diagramme l'opérateur de maintenance
afin de ne pas surcharger le diagramme

SOLUTION

- Les acteurs secondaires (*SA Visa et SI banque*) sollicités dans le cas d'utilisation "*Retirer de l'argent*" ne sont pas les mêmes dans le cas du "*Porteur de carte*" et dans celui du "*client de la banque*".

première version retenue

SOLUTION


GAB
Retirer de l'argent
Consulter le solde
Déposer du numéraire
Déposer des chèques
Porteur de carte Visa
Client banque
secondaire
<<actor>>
SA Visa
secondaire
secondaire
secondaire
secondaire
<<actor>>
SI banque

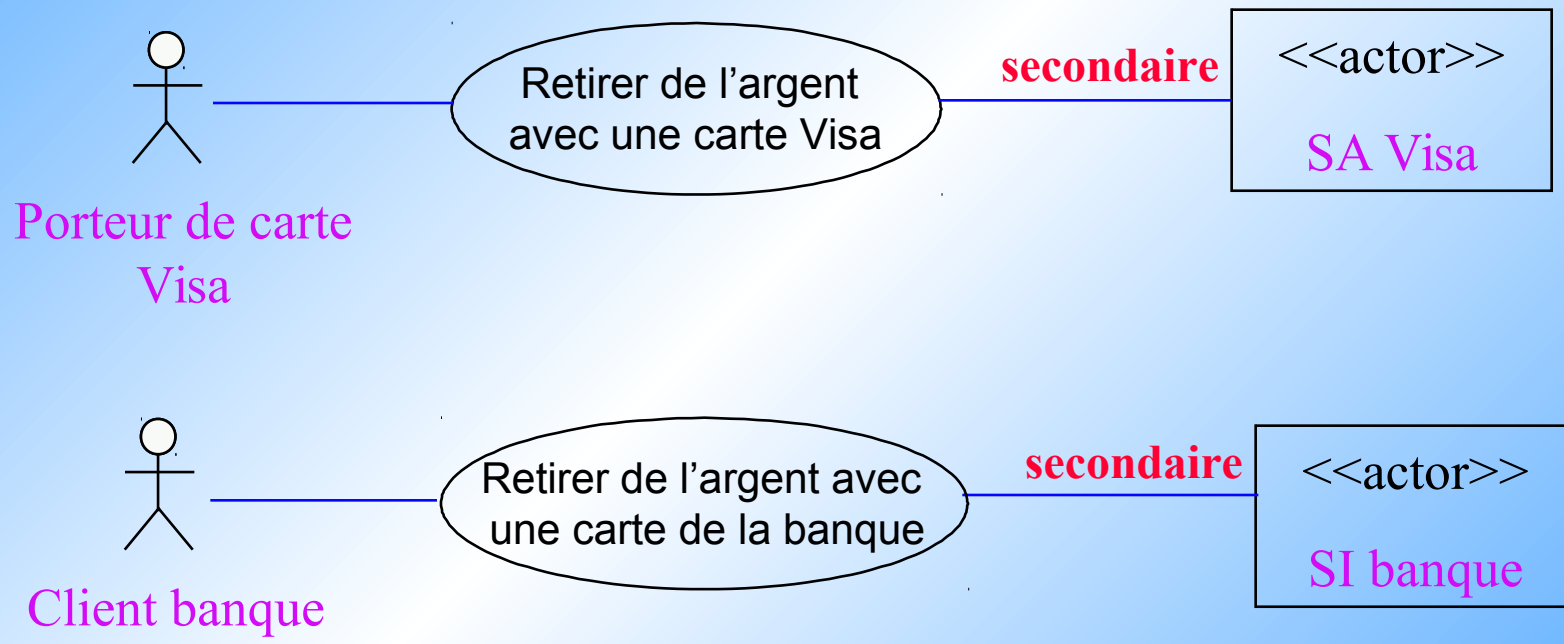
SOLUTION

- **Problème avec le cas d'utilisation "*Retirer de l'argent*" :**

Si l'acteur principal est un *porteur de carte Visa*, il faut faire appel au *SA Visa* (*qui se chargera ensuite de contacter le SI de la banque du porteur*), alors que, s'il s'agit d'un *client de la banque*, le GAB contactera directement le *SI de la banque*.

Proposer une autre solution

SOLUTION


Retirer de l'argent avec une carte Visa
secondaire
<<actor>>
SA Visa
Porteur de carte Visa
Retirer de l'argent avec une carte de la banque
secondaire
<<actor>>
SI banque
Client banque

Cas d'utilisation du site Web

Rechercher les cas d'utilisation pour l'internaute

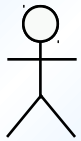
SOLUTION

Cas d'utilisation pour l'internaute

- Rechercher des ouvrages
- Gérer son panier
- Effectuer une commande
- Consulter l'historique des commandes

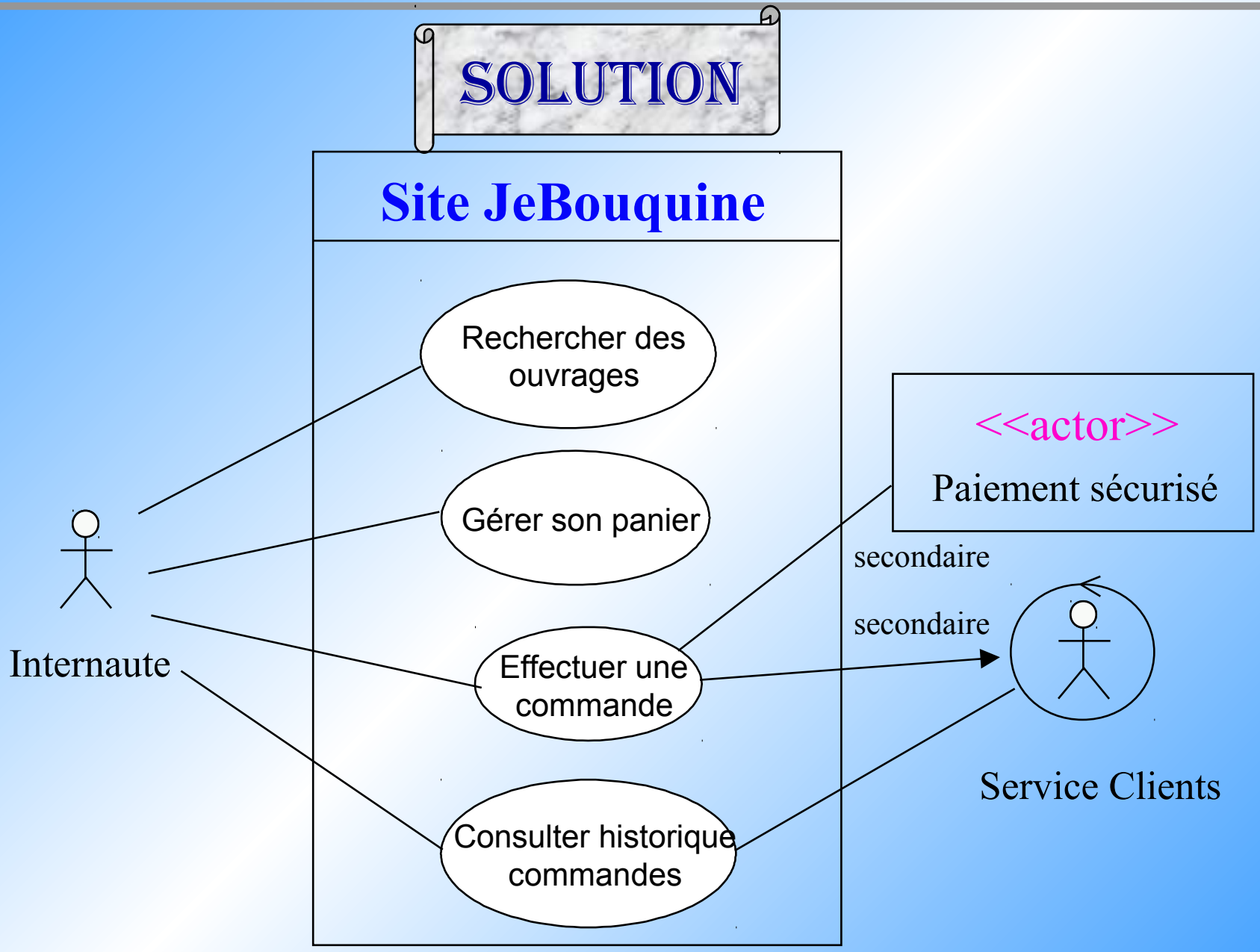
CAS D'UTILISATION DU SITE WEB

Faire le diagramme de cas d'utilisation pour l'internaute

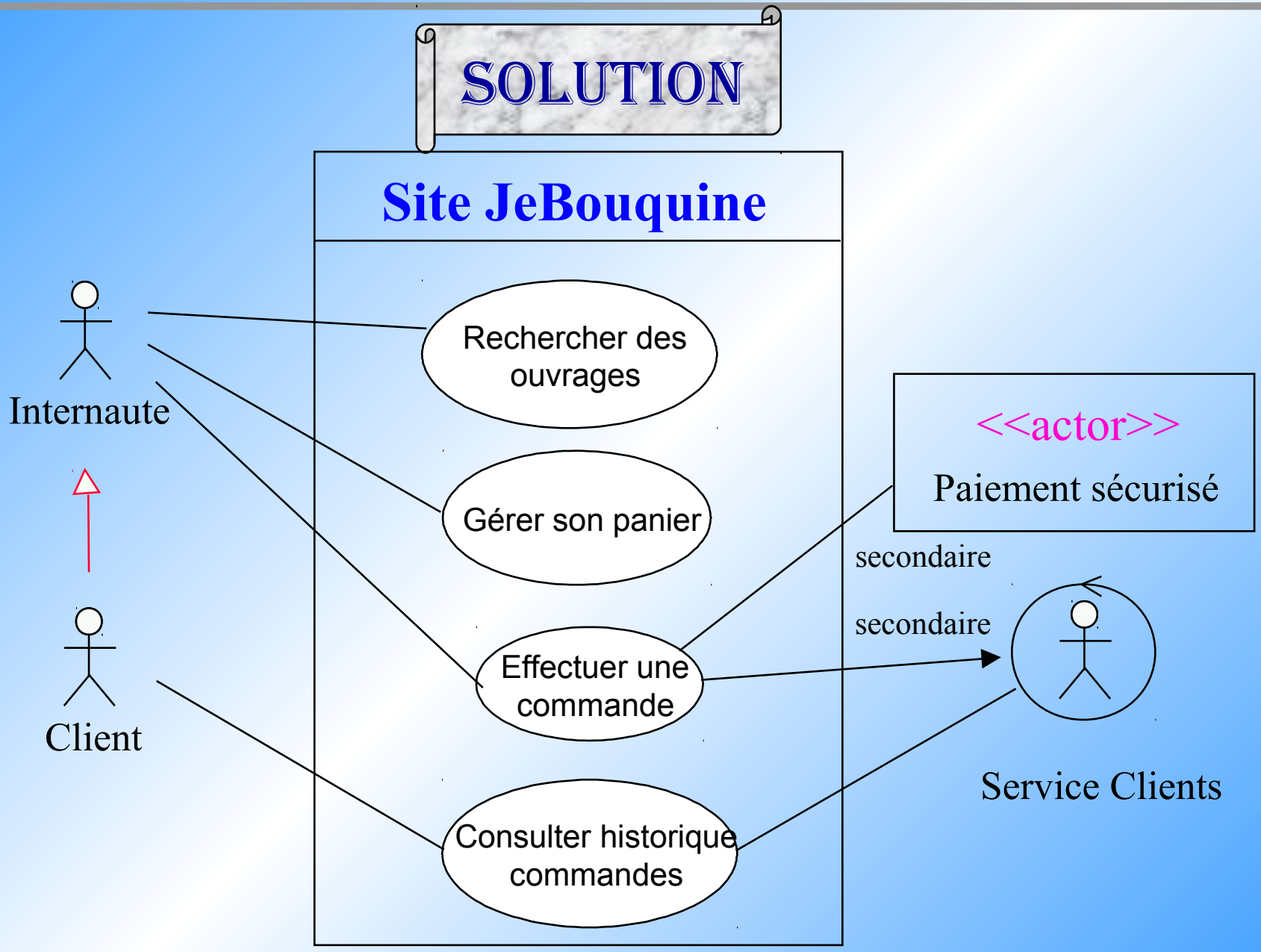


Internaute

SOLUTION


Site JeBouquine
Rechercher des ouvrages
Gérer son panier
Effectuer une commande
Consulter historique commandes
Internaute
<<actor>>
Paiement sécurisé
secondaire
secondaire
Service Clients

SOLUTION


Site JeBouquine
Internaute
Client
Rechercher des ouvrages
Gérer son panier
Effectuer une commande
Consulter historique commandes
<<actor>>
Paiement sécurisé
secondaire
secondaire
Service Clients

SOLUTION

- L'utilisation de la **flèche** sur l'association avec l'acteur *Service Clients* signale un sens unique de transmission d'information.
Cet acteur ne fait que recevoir des messages du système, sans jamais lui en envoyer.

- Un cas d'utilisation peut faire participer plusieurs acteurs et un acteur peut participer à plusieurs cas d'utilisation.

SOLUTION

Chaque cas d'utilisation doit **avoir un objectif en soi** et **pouvoir être réalisé indépendamment des autres**.

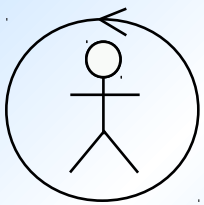
Exemple :

- Un internaute peut visiter le site dans le seul but de rechercher des ouvrages, sans intention d'acheter.
- Il peut aussi gérer un panier virtuel pour faire une simulation, ou obtenir un devis.
- Il peut encore se connecter pour surveiller l'état de sa dernière commande.

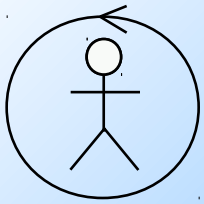
Tous ces objectifs sont bien indépendants et autosuffisants : ce sont de vrais cas d'utilisation.

Cas d'utilisation du site Web

Rechercher les cas d'utilisation pour les employés de *jeBouquine*



Webmaster



Libraire

SOLUTION

Cas d'utilisation pour les employés de *jeBouquine*

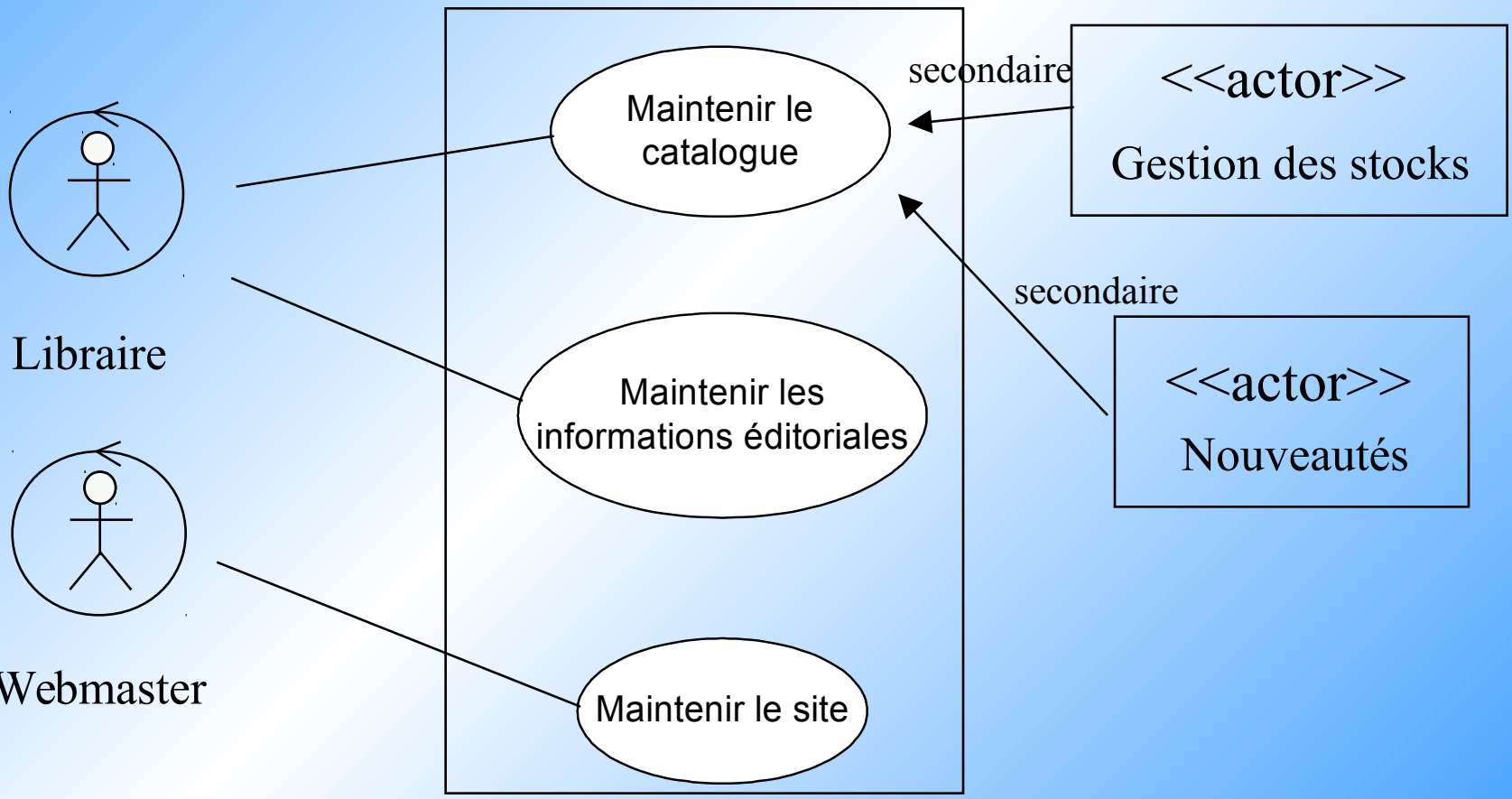
- Maintenir le catalogue, qui fait intervenir les deux systèmes *"Nouveautés"* et *"Gestion des stocks"*
- Maintenir les informations éditoriales
- Maintenir le site

Cas d'utilisation du site Web

Faire le diagramme de cas d'utilisation pour les employés

SOLUTION

Cas d'utilisation pour les employés de *jeBouquine*



ORGANISATION DES CAS D'UTILISATION

2 façons différentes et complémentaires :

- en ajoutant des relations d'inclusion,
  d'extension,
  de généralisation entre les cas d'utilisation.

- en les regroupant en paquetages,
  afin de définir des blocs fonctionnels de plus haut niveau.

SOMMAIRE

- Introduction
- Identification des acteurs
- Identification des cas d'utilisation
- **Relations entre cas d'utilisation**
- Structuration en paquetages

TYPES DE RELATIONS

3 types de relations standardisées

- **Une relation d'inclusion**, formalisée par un mot-clé <<inclut>> ou <<include>> : le cas d'utilisation de base en incorpore explicitement un autre, de façon **obligatoire**.

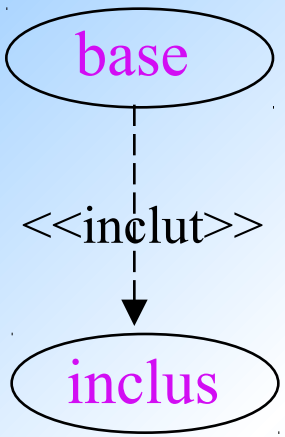
- **Une relation d'extension**, formalisée par un mot-clé <<étend>> ou <<extend>> : le cas d'utilisation de base en incorpore implicitement un autre, de façon **optionnelle**.

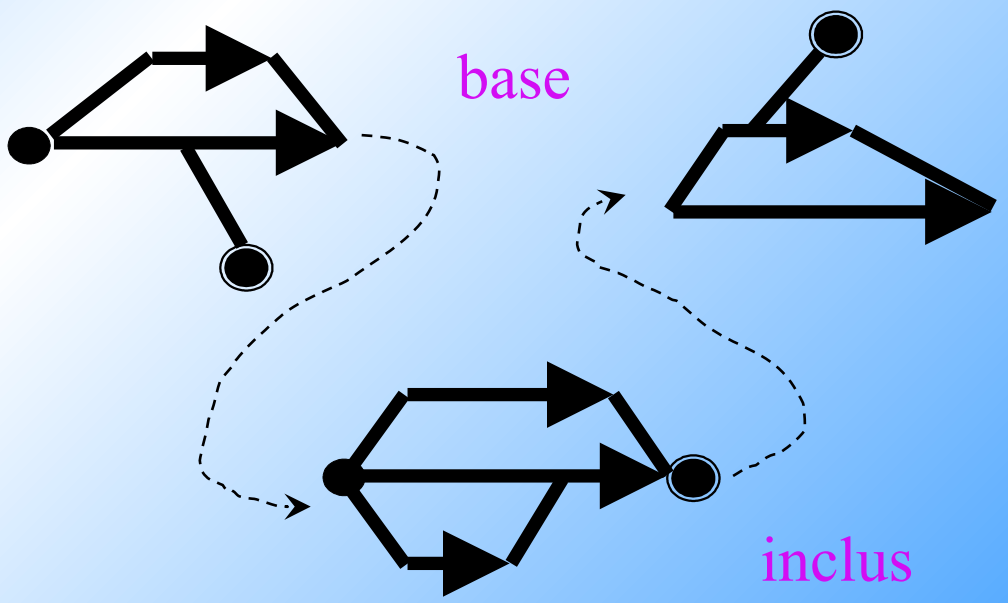
- **Une relation de généralisation/spécialisation** : les cas d'utilisation descendants héritent de la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des interactions spécifiques supplémentaires.

RELATION D'INCLUSION

Relation <<INCLUT>>

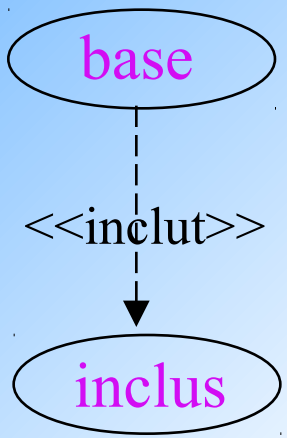
- Le cas de **base** en incorpore explicitement un autre, à un endroit spécifié dans ses enchaînements.
- Le cas d'utilisation **inclus** n'est jamais exécuté seul, mais seulement en tant que partie d'un cas de base plus vaste.





RELATION D'INCLUSION

Relation <<INCLUT>>



- Dans une relation <<inclut>>, le cas d'utilisation de base utilise systématiquement les enchaînements provenant du cas inclus.

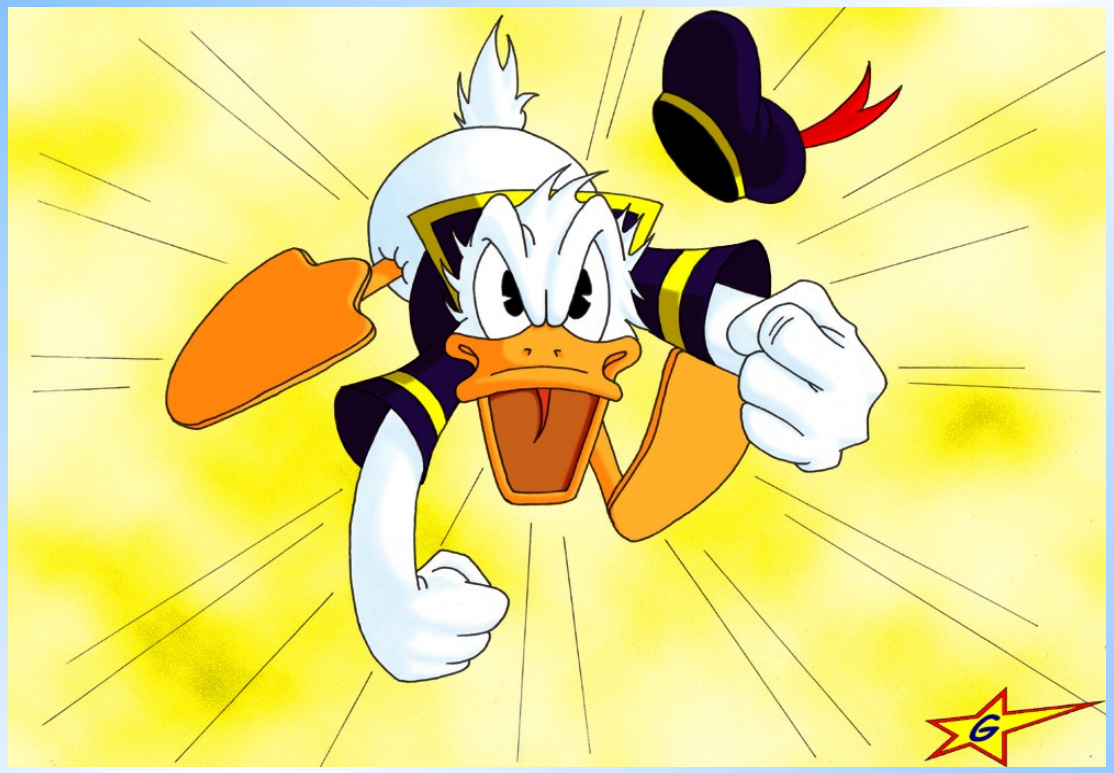
- On utilise fréquemment cette relation pour éviter de décrire plusieurs fois le même enchaînement, en factorisant le comportement commun dans un cas d'utilisation à part.

Relation d'inclusion

Inclusion : Pas de décomposition fonctionnelle !

Une habitude malheureusement assez répandue consiste à utiliser la relation d'inclusion pour décomposer un cas d'utilisation en sous-cas d'utilisation, retombant dans le travers de la décomposition fonctionnelle.

Relation d'inclusion



Inclusion : Pas de décomposition fonctionnelle !

EXERCICE

Exemple d'une entreprise de routage

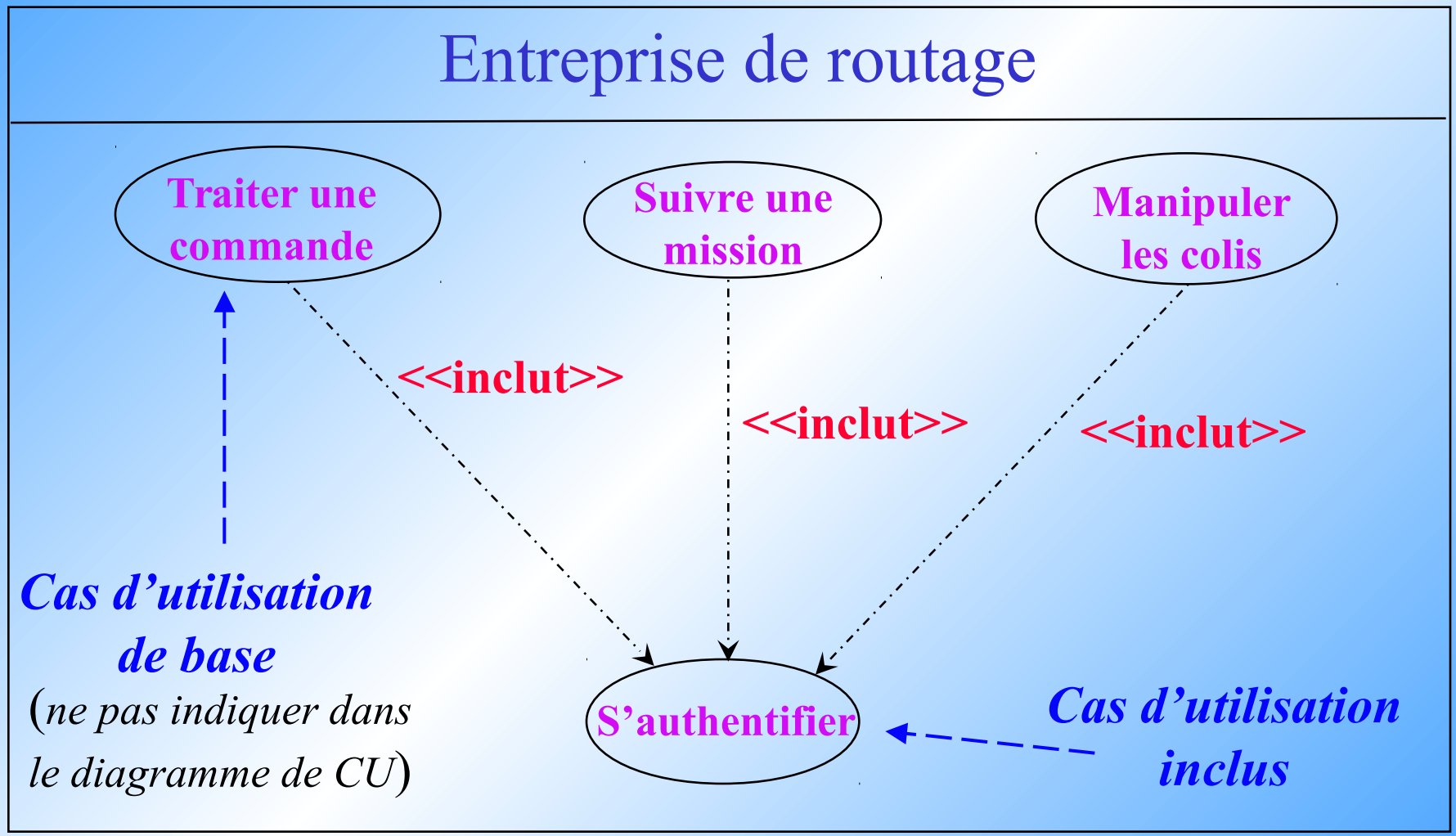
Dans les cas d'utilisation *"Traiter une commande"*, *"Suivre une mission"* et *"Manipuler les colis"*, on spécifie que l'acteur principal doit s'**authentifier**.

Le processus d'authentification implique un flot d'événements entre l'acteur et le système :

- ✓ saisie d'un login,
- ✓ saisie d'un mot de passe,

avec les différents cas d'erreur possibles.

EXERCICE


Entreprise de routage
Traiter une commande
Suivre une mission
Manipuler les colis
<<inclut>>
<<inclut>>
<<inclut>>
Cas d'utilisation de base
(ne pas indiquer dans le diagramme de CU)
S'authentifier
Cas d'utilisation inclus

EXERCICE

Exemple d'une entreprise de routage

- L'authentification fait partie de la capture des besoins puisqu'il est visible de l'utilisateur final.

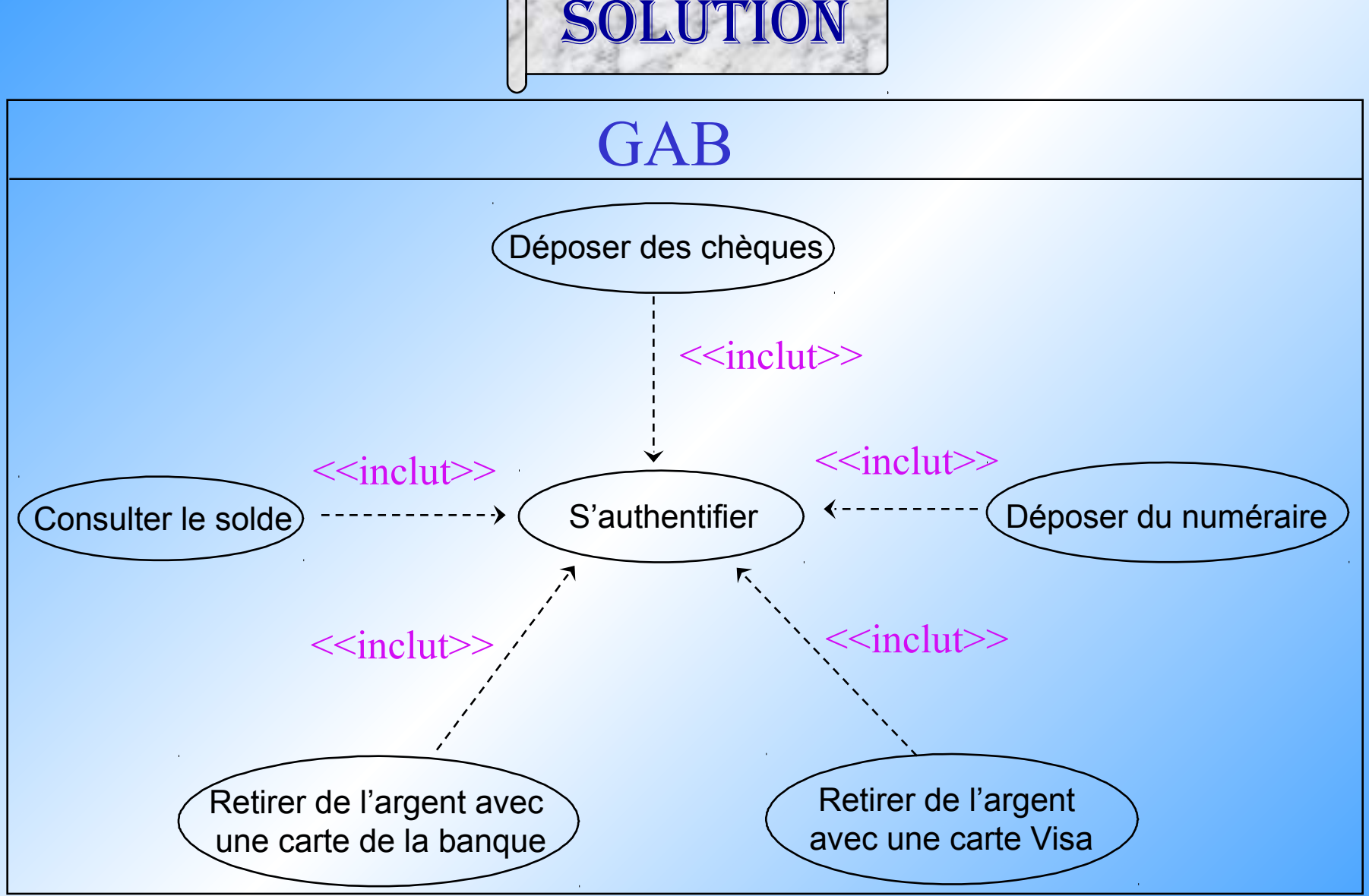
- Il correspond tout à fait à la notion de cas d'utilisation inclus, **ne s'exécutant jamais seul**, mais seulement lorsqu'il est appelé par un cas d'utilisation plus large.

ÉTUDE DE CAS

Trouver des inclusions entre cas d'utilisation

GAB

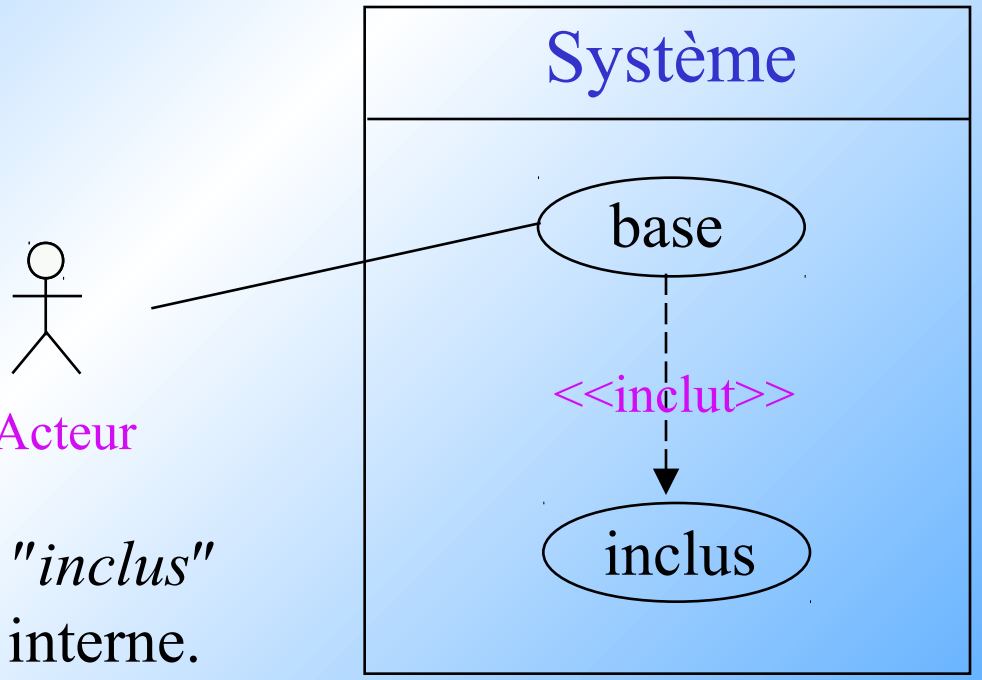
SOLUTION


GAB
Déposer des chèques
<<inclut>>
<<inclut>>
<<inclut>>
Consulter le solde
S'authentifier
Déposer du numéraire
<<inclut>>
<<inclut>>
Retirer de l'argent avec une carte de la banque
Retirer de l'argent avec une carte Visa

RELATION D'INCLUSION

Cas interne

- Un cas d'utilisation est dit **interne** s'il n'est pas relié directement à un acteur.



- Le cas d'utilisation *"inclus"* est un cas d'utilisation interne.

RELATION D'EXTENSION

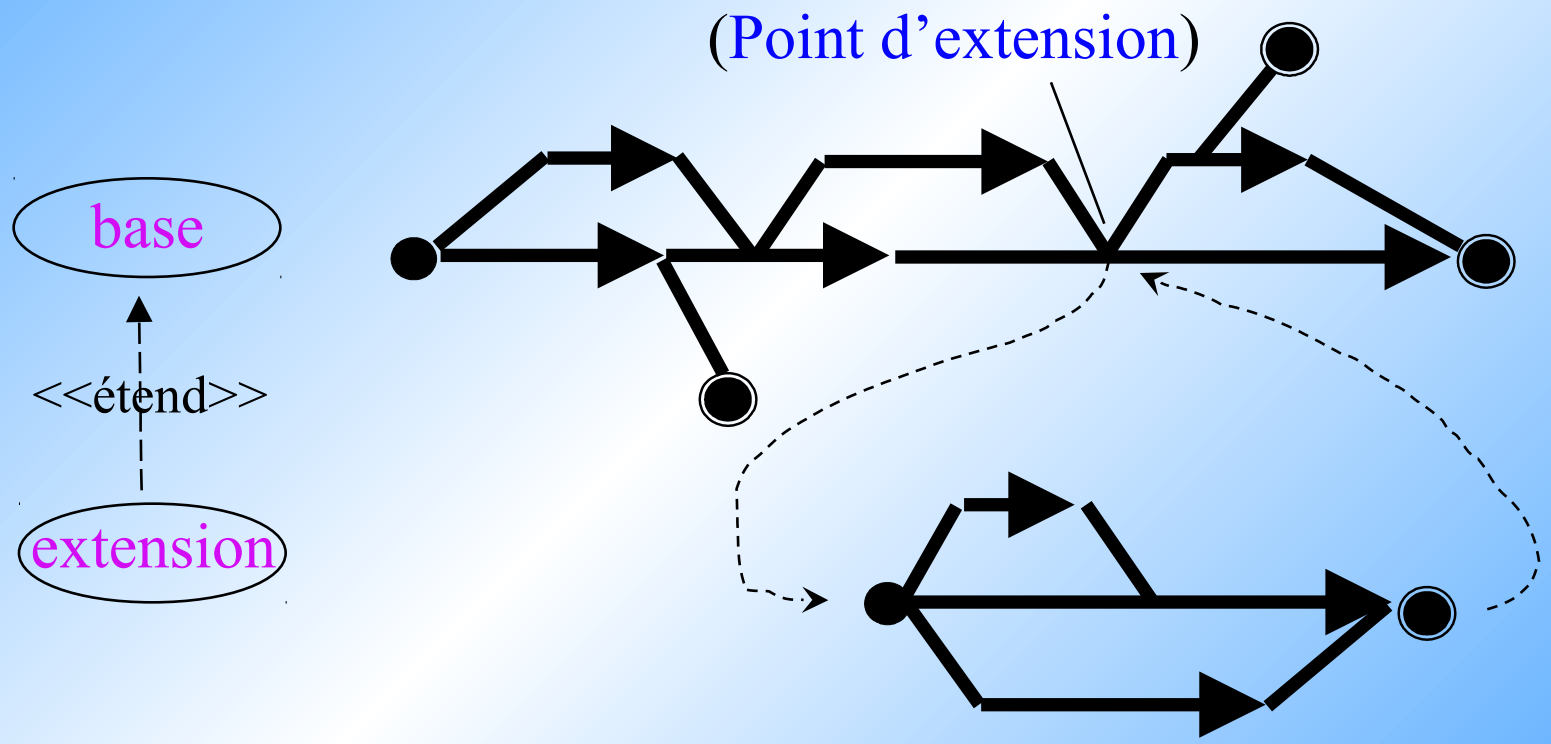
Relation <<ETEND>>

- Le cas de base en incorpore implicitement un autre, à un endroit spécifié indirectement dans celui qui étend.

- Le cas de base peut fonctionner tout seul, mais il peut également être complété par un autre, sous certaines conditions, et uniquement à certains points particuliers de son flot d'événements appelés **points d'extension**.

RELATION D'EXTENSION

Relation <<ETEND>>



RELATION D'EXTENSION

Relation <<ETEND>>

- Dans une relation d'extension, le cas d'utilisation de base recourt optionnellement aux enchaînements provenant du cas d'extension.

- On utilise principalement cette relation pour séparer le comportement ***optionnel*** (*les variantes*) du comportement obligatoire.

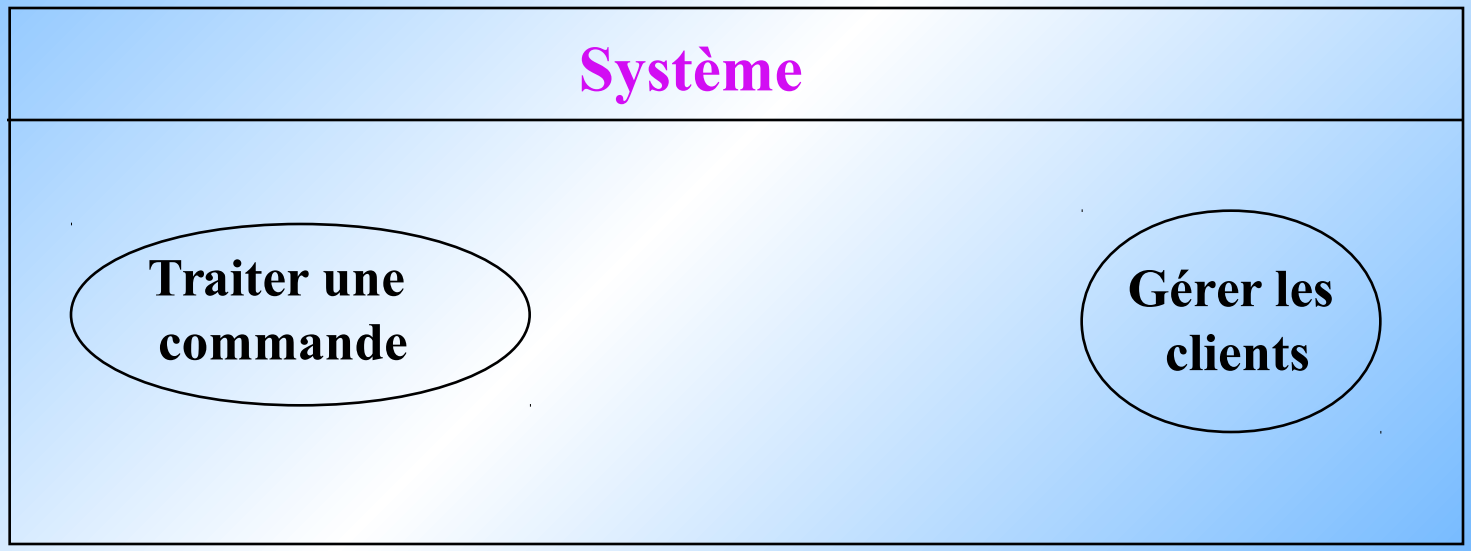
EXERCICE

Dans le cas d'utilisation *"Traiter une commande"*, un des enchaînements principaux consiste à créer une nouvelle commande.
Or, si le client est inconnu du système, le réceptionniste va devoir interrompre son processus de création de commande pour tenter auparavant de créer un nouveau client.
Si ce processus se déroule sans encombre, il pourra alors continuer sa création de commande.
Le processus de création de client fait partie intégrante du cas d'utilisation *"Gérer les infos clients"*.

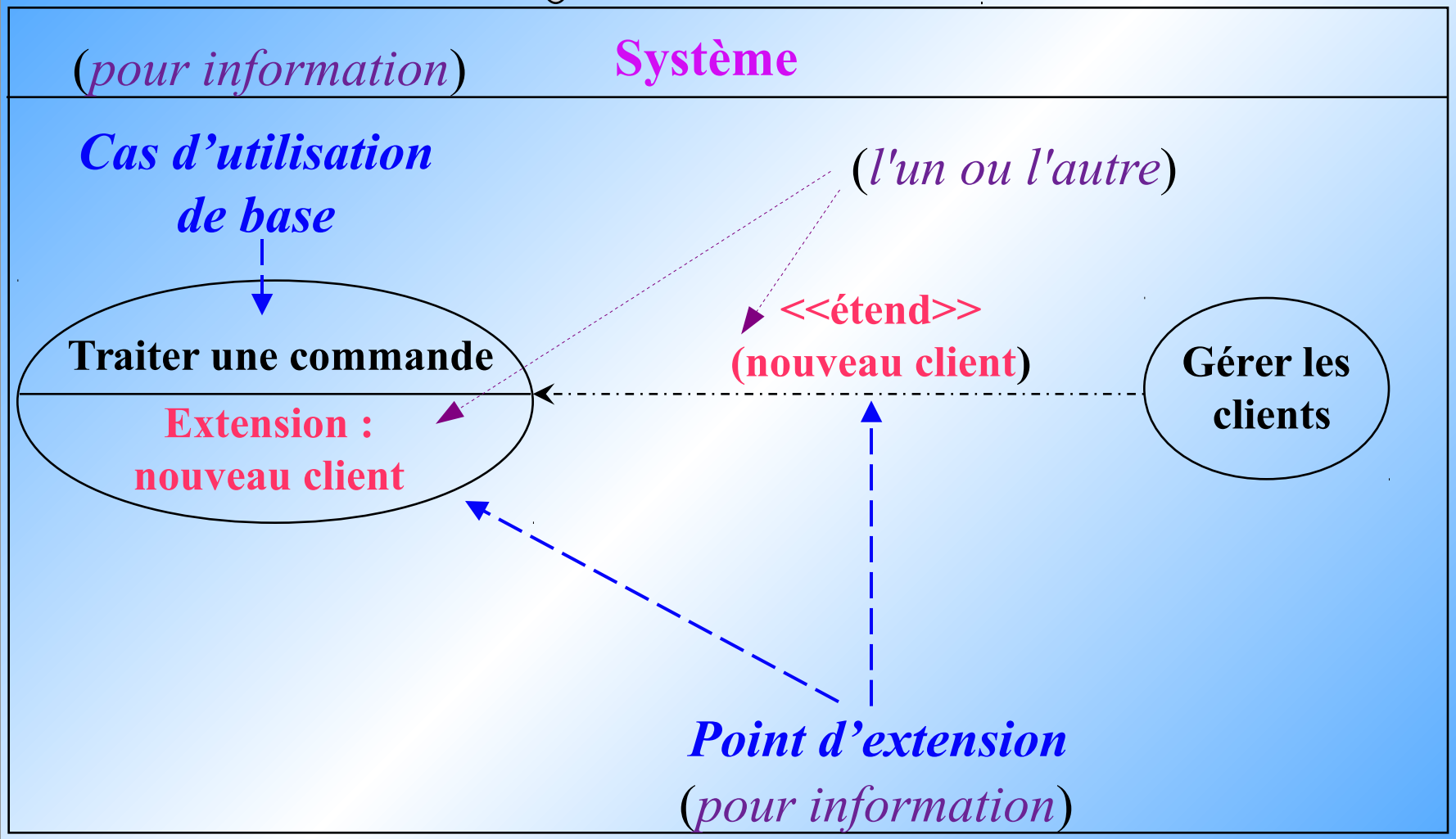
Faire le diagramme de cas d'utilisation

SOLUTION

A compléter



SOLUTION


(pour information)
Système
Cas d'utilisation de base
(l'un ou l'autre)
<<étend>>
(nouveau client)
Traiter une commande
Extension : nouveau client
Gérer les clients
Point d'extension
(pour information)

ÉTUDE DE CAS

Trouver une extension

SOLUTION

- Permettre au client de la banque de consulter son solde juste avant de choisir le montant de son retrait.

- Il pourrait ainsi ajuster le montant demandé avec ce qu'il lui reste sur son compte.

Affiner le diagramme de cas d'utilisation

SOLUTION

GAB

Cas d'utilisation de base

Retirer de l'argent avec une carte de la banque

Extension : vérification montant

<<étend>>
(vérification montant)

Consulter le solde

Point d'extension

SOLUTION

GAB

Retirer de l'argent avec une carte de la banque

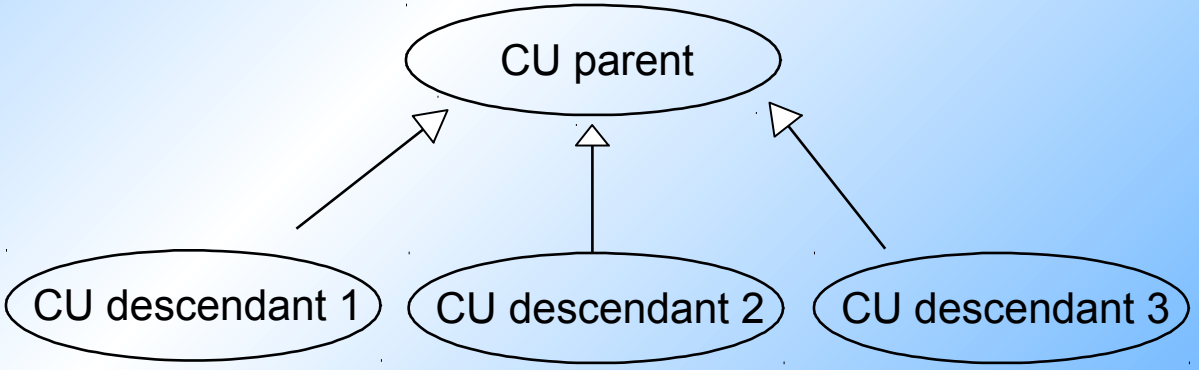
<<étend>>
(vérification montant)

Consulter le solde

Relation de généralisation

- Les cas d'utilisation peuvent être hiérarchisés par généralisation / spécialisation.

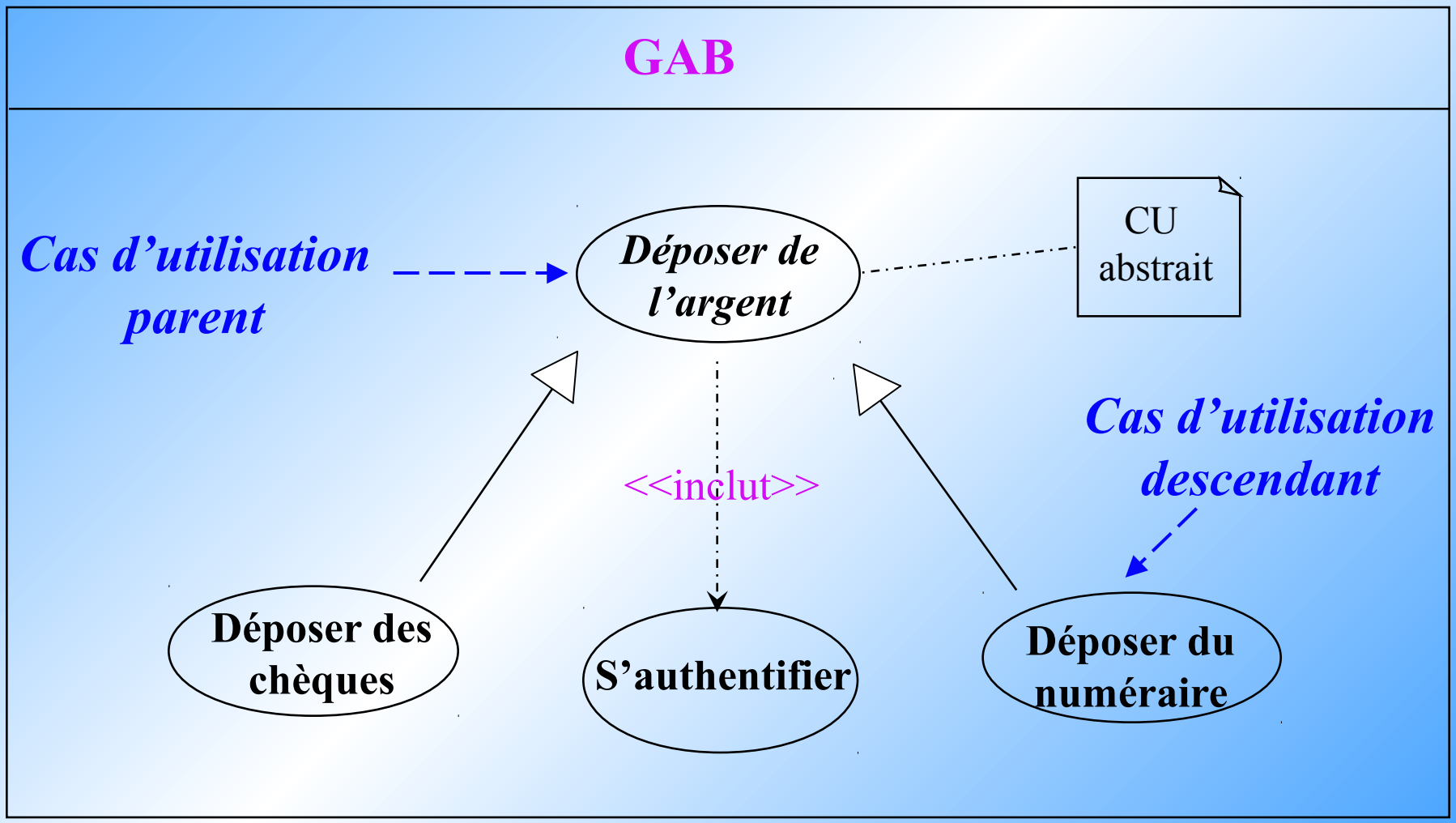
- Les cas d'utilisation descendants héritent de la sémantique de leur parent. Ils peuvent comprendre des interactions spécifiques supplémentaires, ou modifier les interactions héritées.



ÉTUDE DE CAS

Trouver une relation de généralisation

SOLUTION


GAB
Cas d'utilisation parent
Déposer de l'argent
CU abstrait
Cas d'utilisation descendant
<<inclut>>
Déposer des chèques
S'authentifier
Déposer du numéraire

SOLUTION

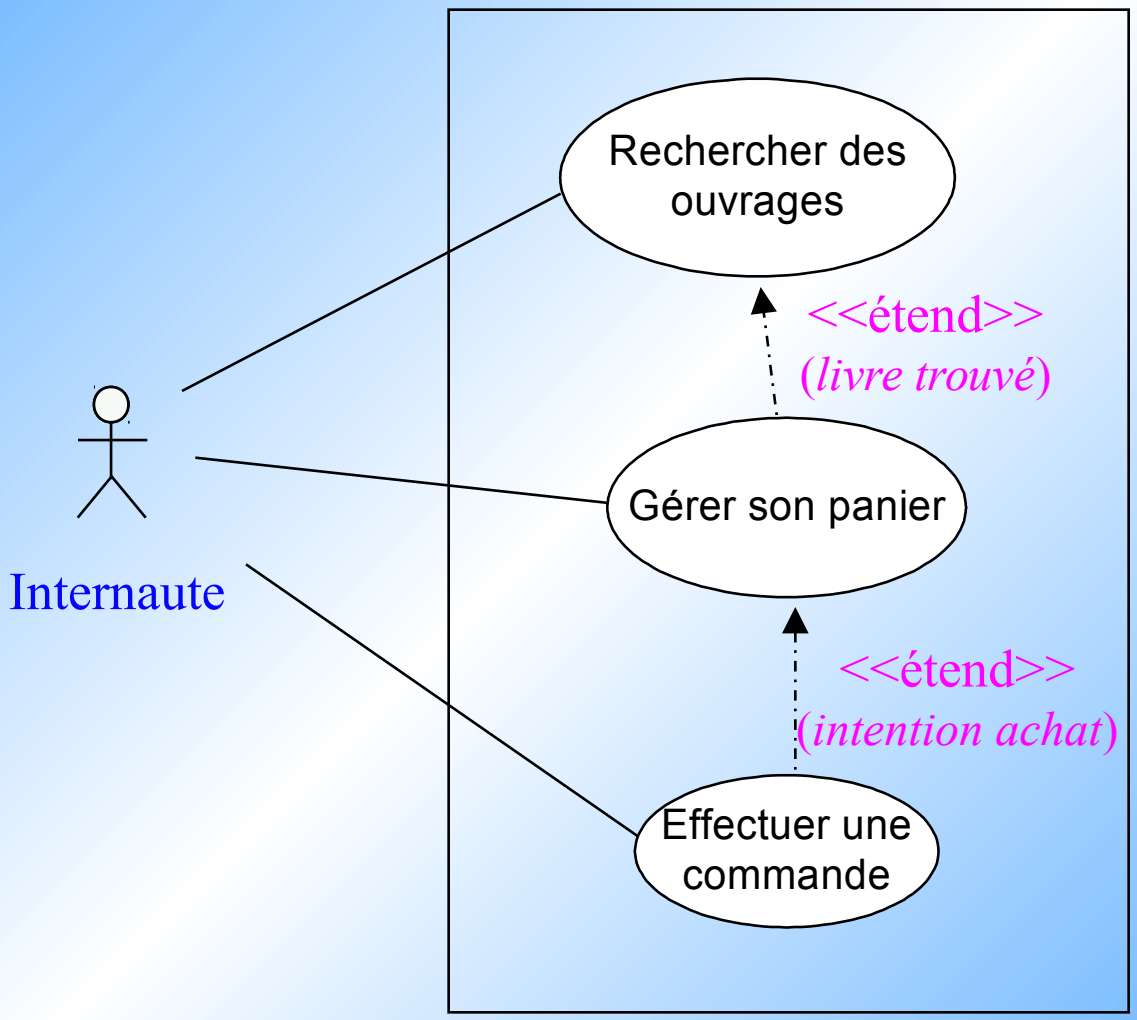
- Les deux cas d'utilisation *"Déposer des chèques"* et *"Déposer du numéraire"* parlent de la même chose :
la possibilité à un client de la banque d'effectuer un dépôt d'argent grâce au GAB.

- Cependant, le détail des enchaînements va varier notablement :
  - le dépôt de numéraire implique un dispositif de reconnaissance de billets, avec des interactions liées à chaque introduction de billets, aux erreurs possibles (*billet non reconnu, etc.*).
  - le système de tenue des comptes (*qui fait partie du SI banque*) est informé en temps réel du dépôt de numéraire afin de créditer le compte contrairement au dépôt de chèques.

Cas d'utilisation du site Web

Organiser les cas d'utilisation de l'internaute

SOLUTION


Rechercher des ouvrages
<<étend>>
(livre trouvé)
Gérer son panier
Internaute
<<étend>>
(intention achat)
Effectuer une commande

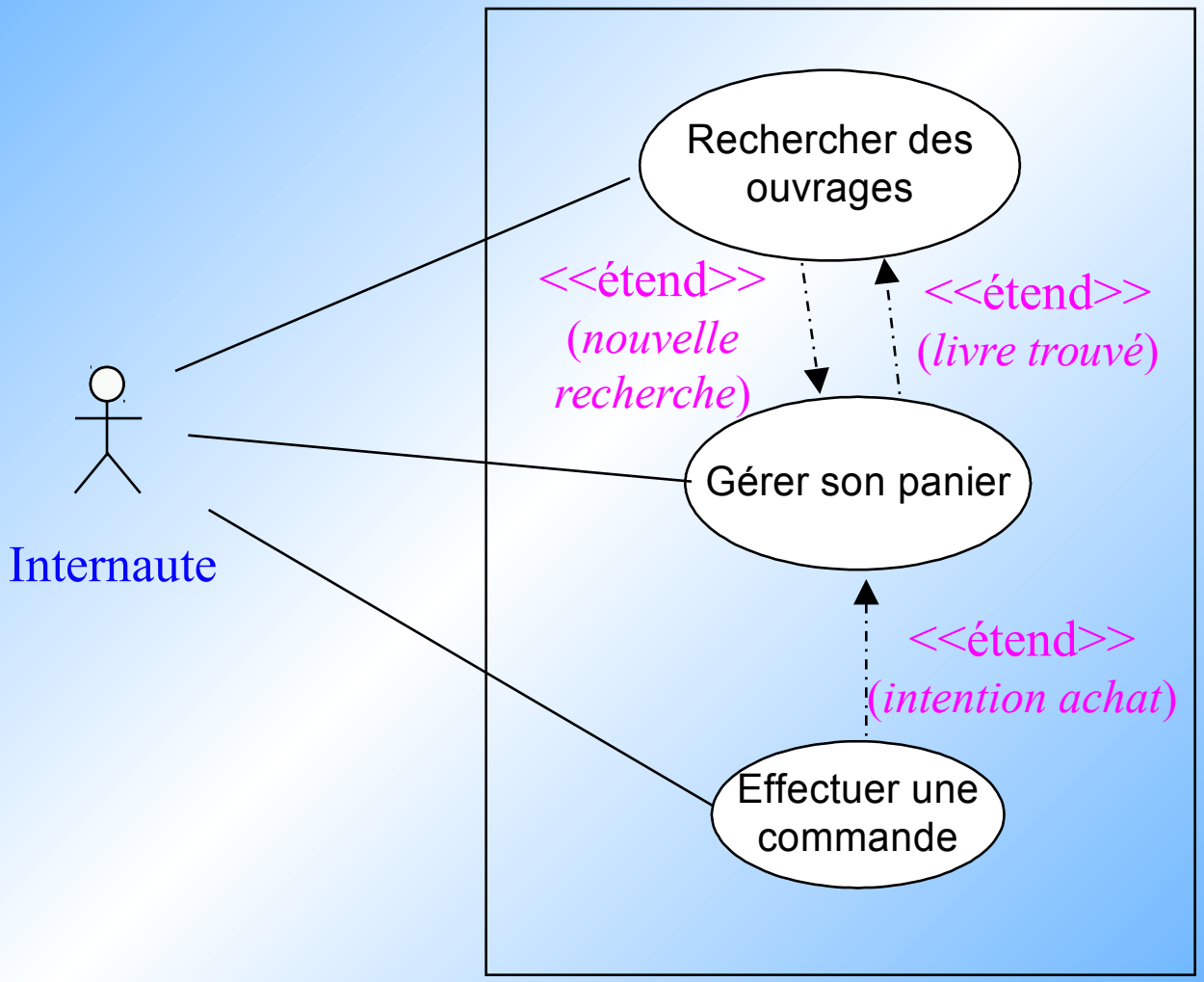
SOLUTION

Les trois cas d'utilisation principaux de l'internaute sont assez naturellement reliés par des relations d'extension :

- La recherche d'ouvrages *peut* donner lieu à leur mise dans le panier.

- La gestion du panier *peut* donner lieu au passage d'une commande.

AUTRE SOLUTION

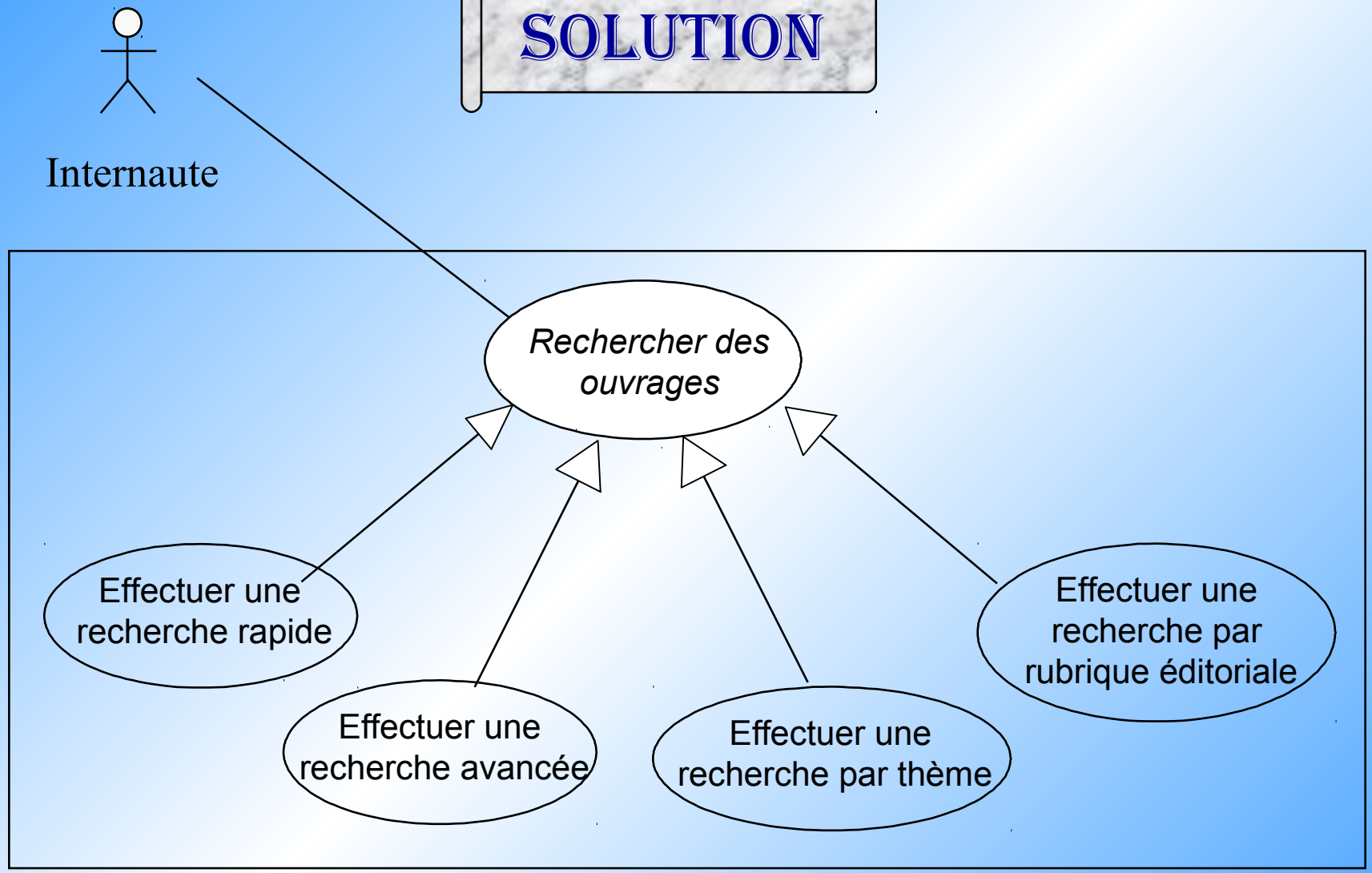


CAS D'UTILISATION DU SITE WEB

- Les différentes possibilités de recherche d'ouvrages peuvent être modélisées avec précision par une relation de généralisation/spécialisation.

Faire le diagramme de cas d'utilisation

SOLUTION


Internaute
Rechercher des ouvrages
Effectuer une recherche rapide
Effectuer une recherche avancée
Effectuer une recherche par thème
Effectuer une recherche par rubrique éditoriale

SOLUTION

- Le nom du cas d'utilisation *"Rechercher des ouvrages"* est en italique.

- Il s'agit d'une convention générale en UML pour indiquer qu'un élément est abstrait et n'est donc pas directement exécutable mais uniquement par le biais de ses spécialisations.

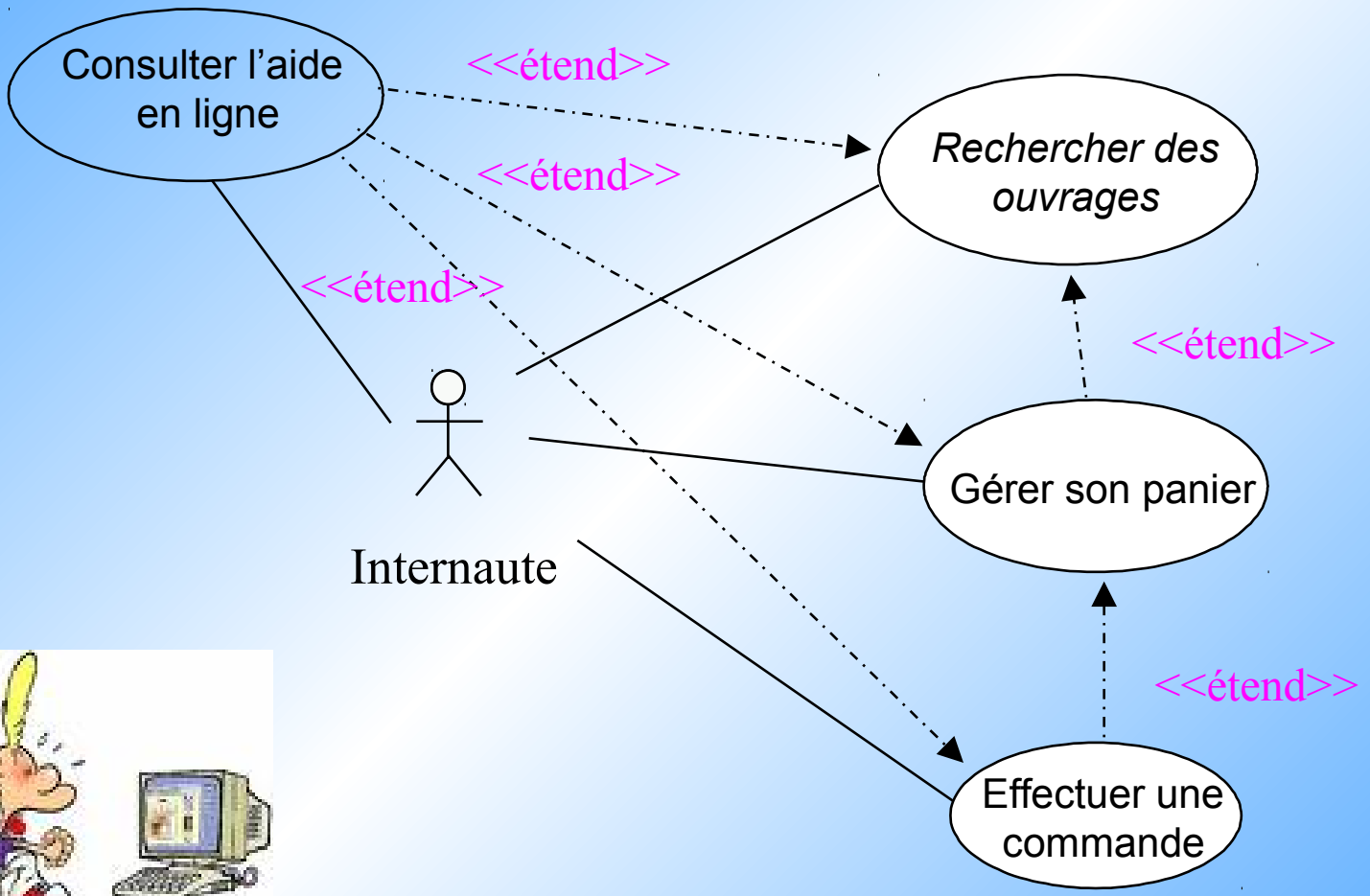
CAS D'UTILISATION DU SITE WEB

- Omission d'un cas d'utilisation de l'internaute :
  consulter l'aide en ligne

- Il ne s'agit pas d'un cas d'utilisation majeur, mais il donnera lieu à des développements importants.
Il ne faut pas l'oublier, même s'il est secondaire par rapport aux autres cas d'utilisation.

Faire le diagramme de cas d'utilisation

SOLUTION


Consulter l'aide en ligne
<<étend>>
<<étend>>
<<étend>>
Rechercher des ouvrages
<<étend>>
Gérer son panier
Internaute
<<étend>>
Effectuer une commande

SOLUTION

- On s'aperçoit que ce cas d'utilisation peut étendre absolument tous les autres !

- A tout moment, alors qu'il recherche des ouvrages, gère son panier, etc., l'internaute peut s'interrompre pour consulter l'aide en ligne avant de poursuivre son activité.

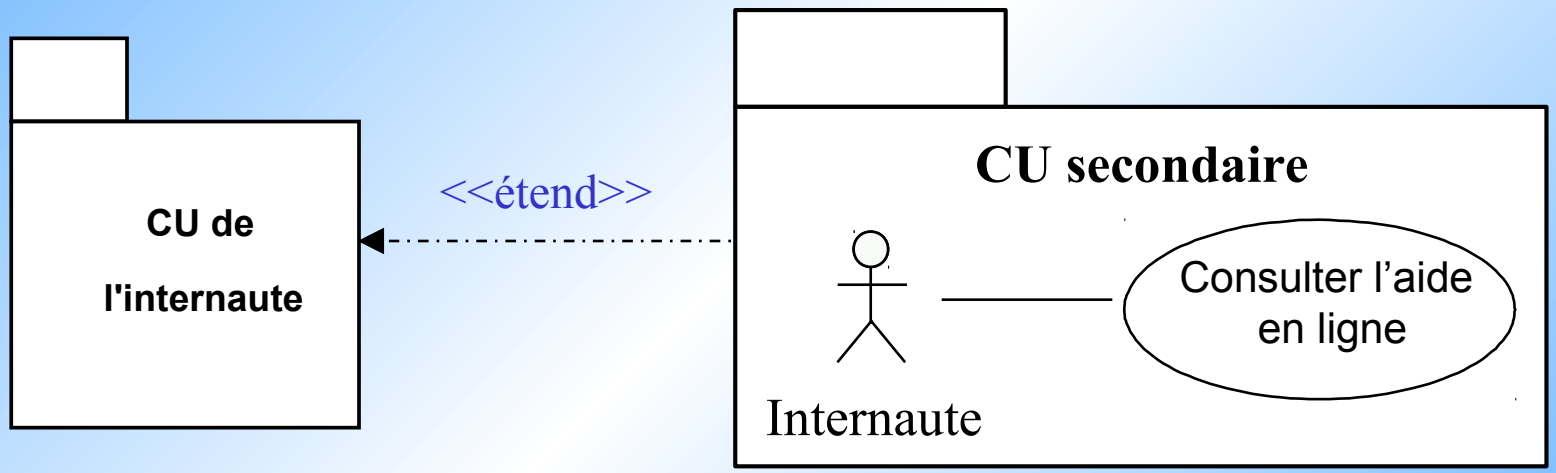
- On peut donc relier ce cas d'utilisation à tous les autres par une flèche pointillée <<étend>>.

SOLUTION

- Le diagramme ainsi obtenu n'est pas satisfaisant car il est devenu trop complexe par rapport à l'intérêt de l'information ajoutée.

- On préfère plutôt isoler le cas d'utilisation *Consulter l'aide en ligne* dans un paquetage à part, intitulé CU secondaires, et ne plus faire apparaître toutes les relations d'extension comme sur le diagramme précédent.

SOLUTION


CU de
l'internaute
<<étend>>
CU secondaire
Consulter l'aide
en ligne
Internaute

SOMMAIRE

- Introduction
- Identification des acteurs
- Identification des cas d'utilisation
- Relations entre cas d'utilisation
- **Structuration en paquetages**

STRUCTURATION EN PAQUETAGES

- Regrouper les cas d'utilisation en ensembles fonctionnels cohérents. Pour ce faire, on utilise le concept général d'UML qui s'appelle le paquetage.

- **Paquetage** : mécanisme général de regroupement d'éléments en UML, qui peut être utilisé par exemple pour regrouper des cas d'utilisation, mais aussi des acteurs, des classes, etc.

- Les éléments contenus dans un paquetage :

✓ doivent représenter un ensemble fortement cohérent,

✓ sont généralement de même nature et de même niveau sémantique.

Étude de cas

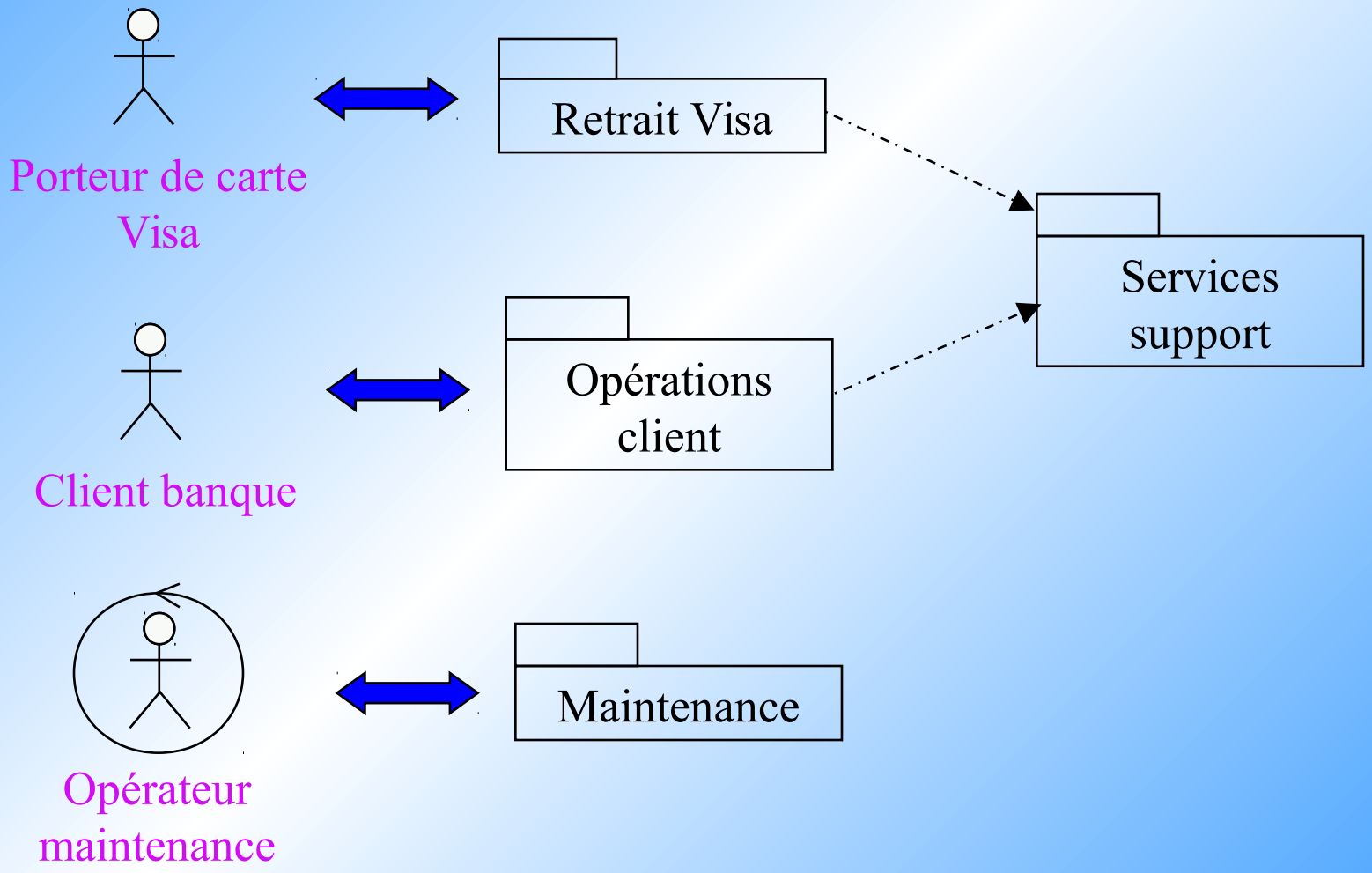
Proposer une structuration des cas d'utilisation en paquetages

SOLUTION

- Plusieurs stratégies sont possibles :
  - regroupement par acteur
  - regroupement par domaine fonctionnel

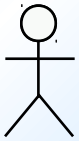
- Un regroupement par acteur principal est souhaitable car cela permet également de répartir les acteurs secondaires.

SOLUTION


Porteur de carte Visa
Retrait Visa
Client banque
Opérations client
Services support
Opérateur maintenance
Maintenance

SOLUTION

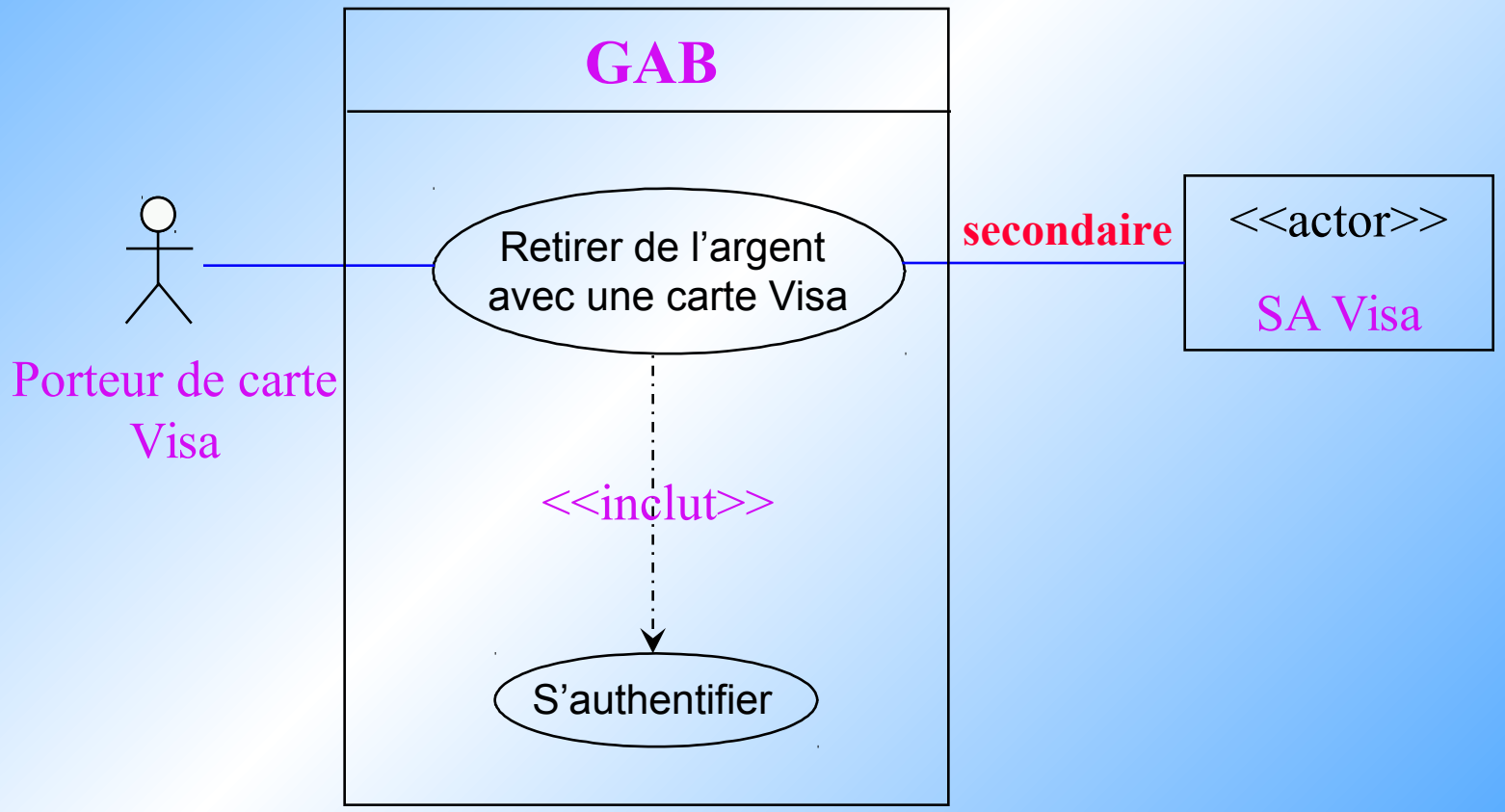
Faire le diagramme de cas d'utilisation pour le paquetage "Retrait Visa"



Porteur de carte
Visa

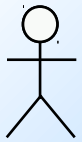
SOLUTION

Paquetage Retrait Visa



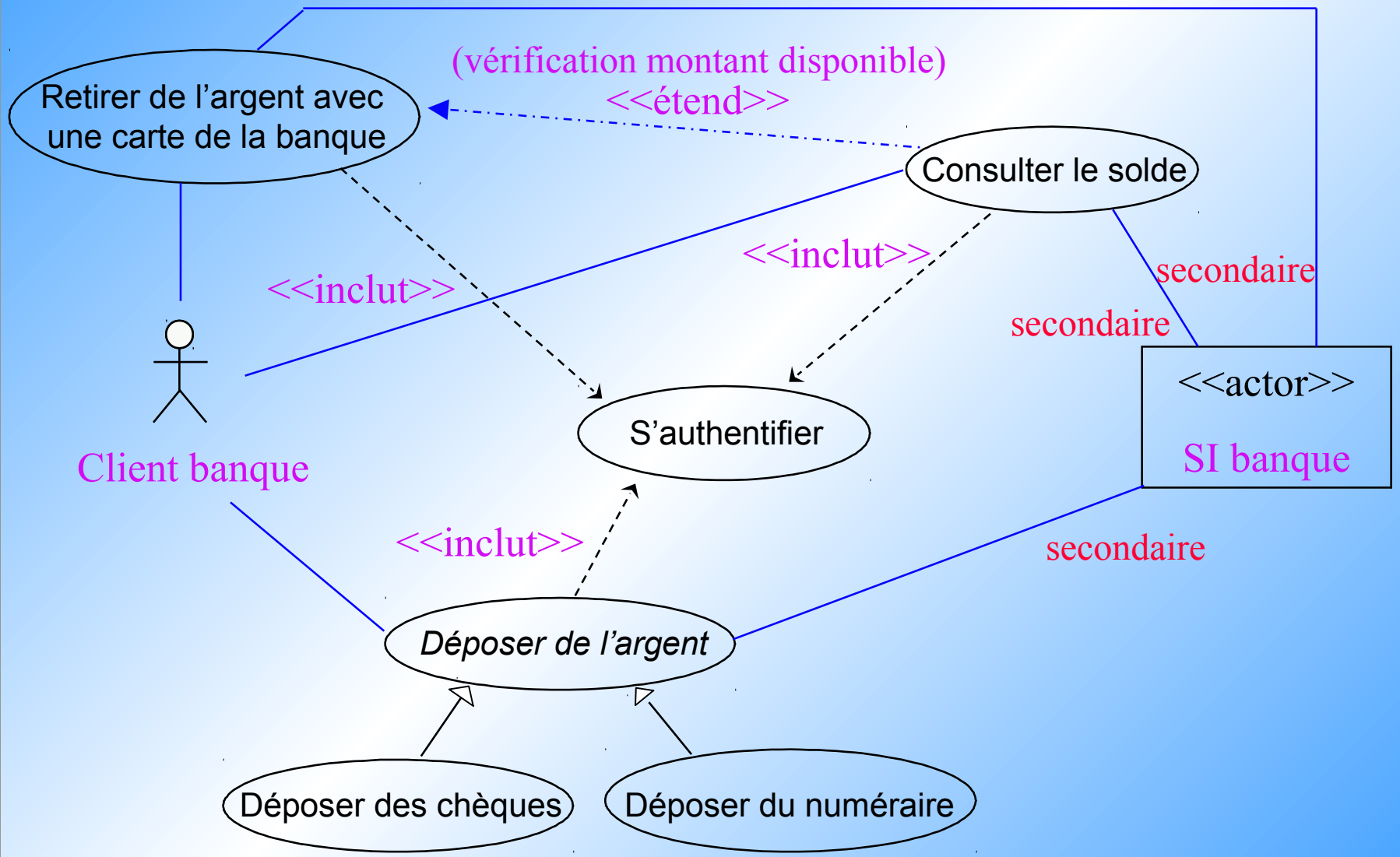
SOLUTION

Faire le diagramme de cas d'utilisation pour le paquetage "Opérations Client"

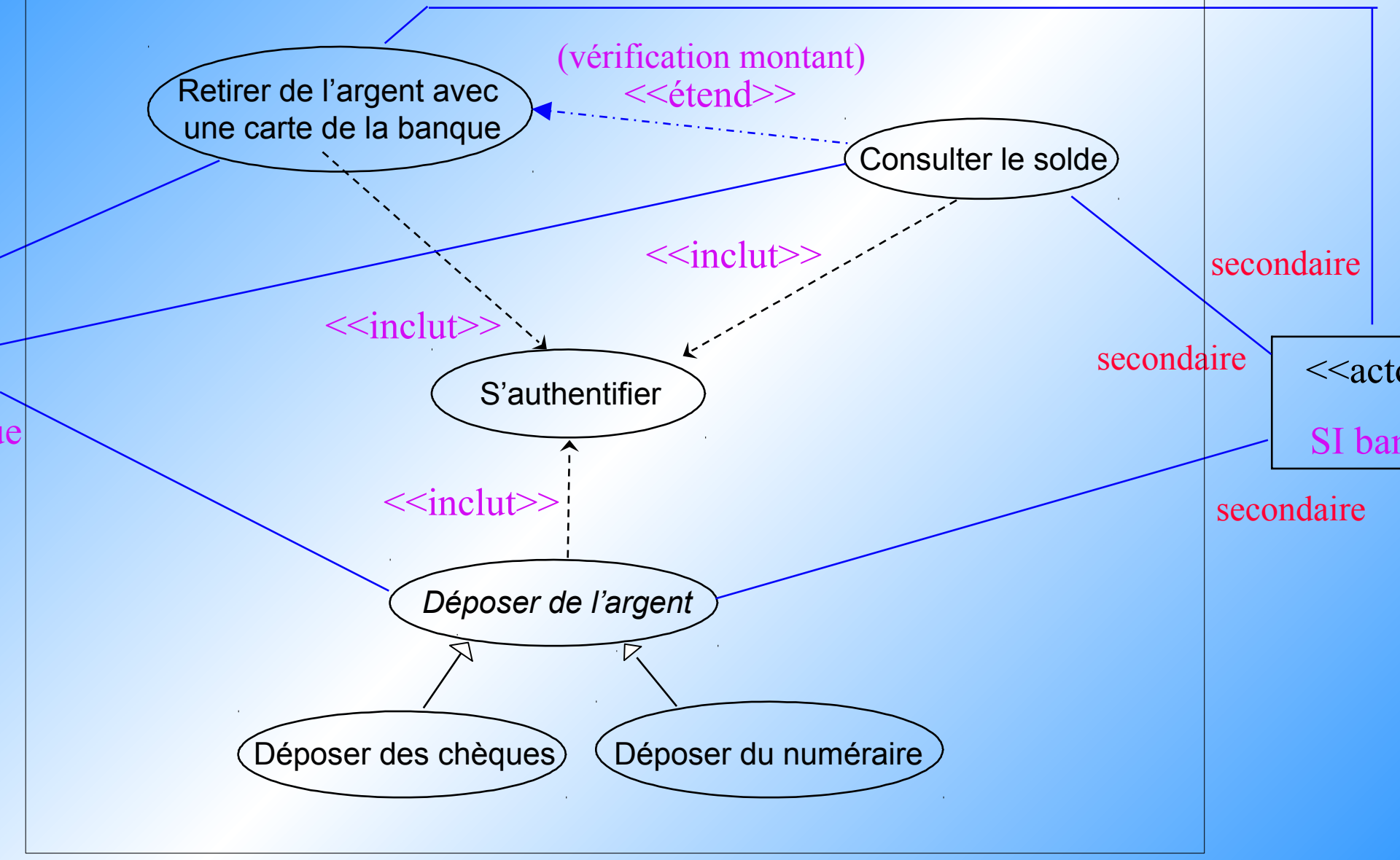


Client banque

Paquetage Opérations client

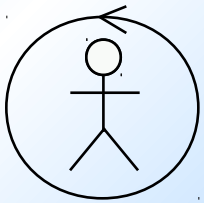


GAB


Retirer de l'argent avec une carte de la banque
(vérification montant)
<<étend>>
Consulter le solde
<<inclut>>
<<inclut>>
secondaire
secondaire
S'authentifier
<<acte
SI ban
<<inclut>>
secondaire
Déposer de l'argent
Déposer des chèques
Déposer du numéraire

SOLUTION

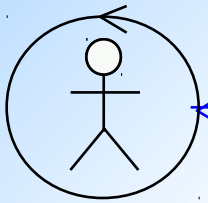
Faire le diagramme de cas d'utilisation pour le paquetage "Maintenance"



Opérateur maintenance

SOLUTION

Paquetage Maintenance



Opérateur maintenance