



Génie logiciel

Cas d'utilisation

Louis-Edouard LAFONTANT





UML: Cas d'utilisation

Bien souvent, les clients et les utilisateurs ne sont pas des informaticiens. Il leur faut donc un moyen simple d'exprimer leurs besoins.

=> Diagrammes de cas d'utilisation (CU)

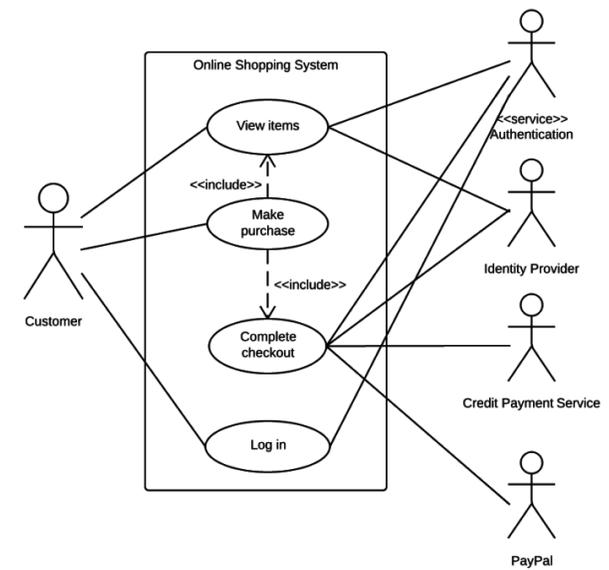
- Recueillir besoin
- Analyser besoin
- Organiser besoin
- Recenser les grandes fonctionnalités d'un système

Cas d'utilisation UML

Capture le comportement d'un système ou d'un composant tel qu'un utilisateur extérieur le voit.

Scinde les fonctionnalités en unités cohérentes (cas d'utilisation), ayant un sens pour les acteurs.

Permettent d'exprimer le besoin des utilisateurs d'un système



Éléments d'un CU

Systeme

- Cadre du système à développer

Acteur

- A un but en utilisant le système
- Joue un rôle d'interaction

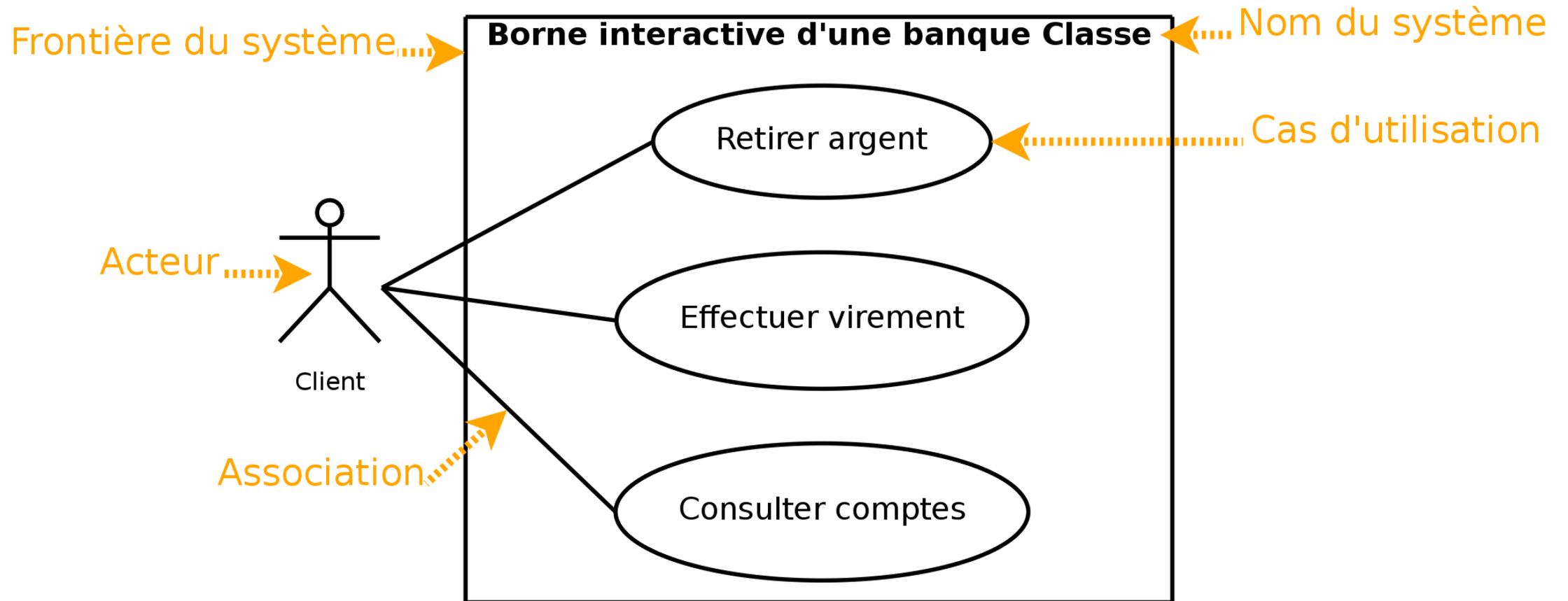
But (objectif)

- Ce que l'acteur veut accomplir en interagissant avec le système
- Chaque but est un CU
- Écrit en texte

Diagramme de cas d'utilisation UML

- Technique pour identifier et décrire les fonctionnalités d'un logiciel qui sont significatives pour ses utilisateurs
 - Décrit une **interaction** entre le logiciel et ses utilisateurs (acteurs)
 - Besoins **fonctionnels** des utilisateurs cibles
- Expression du comportement du logiciel tel que perçu par les acteurs
 - **CU initié par un acteur**, puis peut interagir avec d'autres acteurs
 - Représente le flux complet des événements à travers le système

Diagramme CU UML



Acteur

- Élément **externe** au logiciel
- Joue un rôle du POV du système
 - Utilisateur
 - Initiateur
 - Personne qui joue un rôle critique dans un CU
 - Pas nécessairement humain
 - Horloge, alarme

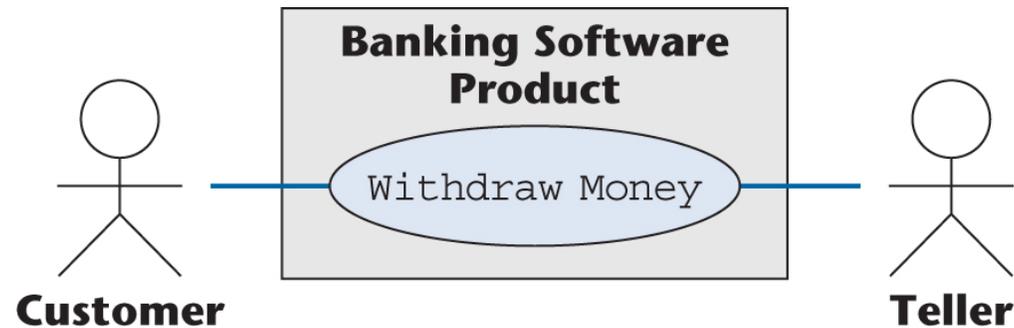


Acteur multiples

- Une personne peut jouer **plusieurs rôles**
 - Client d'une banque peut être l'emprunteur ou le prêteur
- Réciproquement, un acteur peut participer dans **plusieurs CU**
- Exemple: Emprunteur peut être un acteur dans
 - Le CU Emprunte argent
 - Le CU Paye intérêts
 - Le CU Rembourse capital

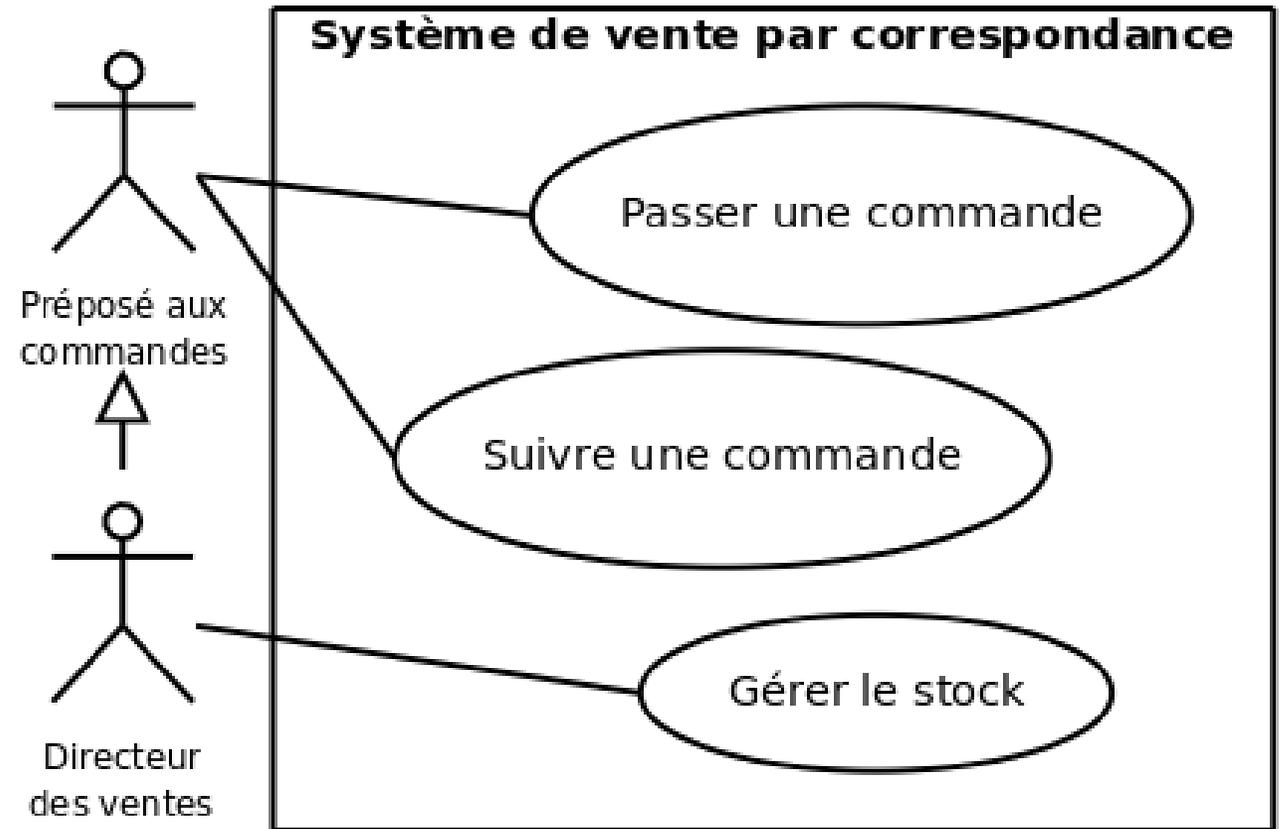
Identification des acteurs

- Facile à identifier: utilise fréquemment le logiciel
- Acteurs **principaux**
 - Initiateur du CU, bénéficiaire principal et direct du but du CU
- Acteurs **secondaires**
 - Contribuent à la réalisation du CU



Relation entre acteurs

- La seule relation possible entre deux acteurs est la généralisation
- Un acteur A est une généralisation d'un acteur B si l'acteur A peut être substitué par l'acteur B.
- Dans ce cas, tous les cas d'utilisation accessibles à A le sont aussi à B, mais l'inverse n'est pas vrai



Cas d'utilisation

- Généralisation des scénarios (succès et échecs) qui ont le même **but**
- Scénario **principal**
- Scénarios **alternatifs** (optionnels)
- **Préconditions** nécessaires pour entamer le scénario principal
- **Postconditions** de l'état du système lorsque le CU est complété avec succès.

Scénario

- **Séquence** spécifique **d'interactions** entre les acteurs et le système et **d'actions** subséquentes du système
 - Exécution particulière d'un CU
 - Correspond à un succès ou un échec
- Description fonctionnelle, non pas structurelle
 - Spécifier ce que le système doit accomplir, non pas comment
 - Concentrer sur les préoccupations réelles des utilisateurs, pas de solutions d'implémentation

Scénario

Principal

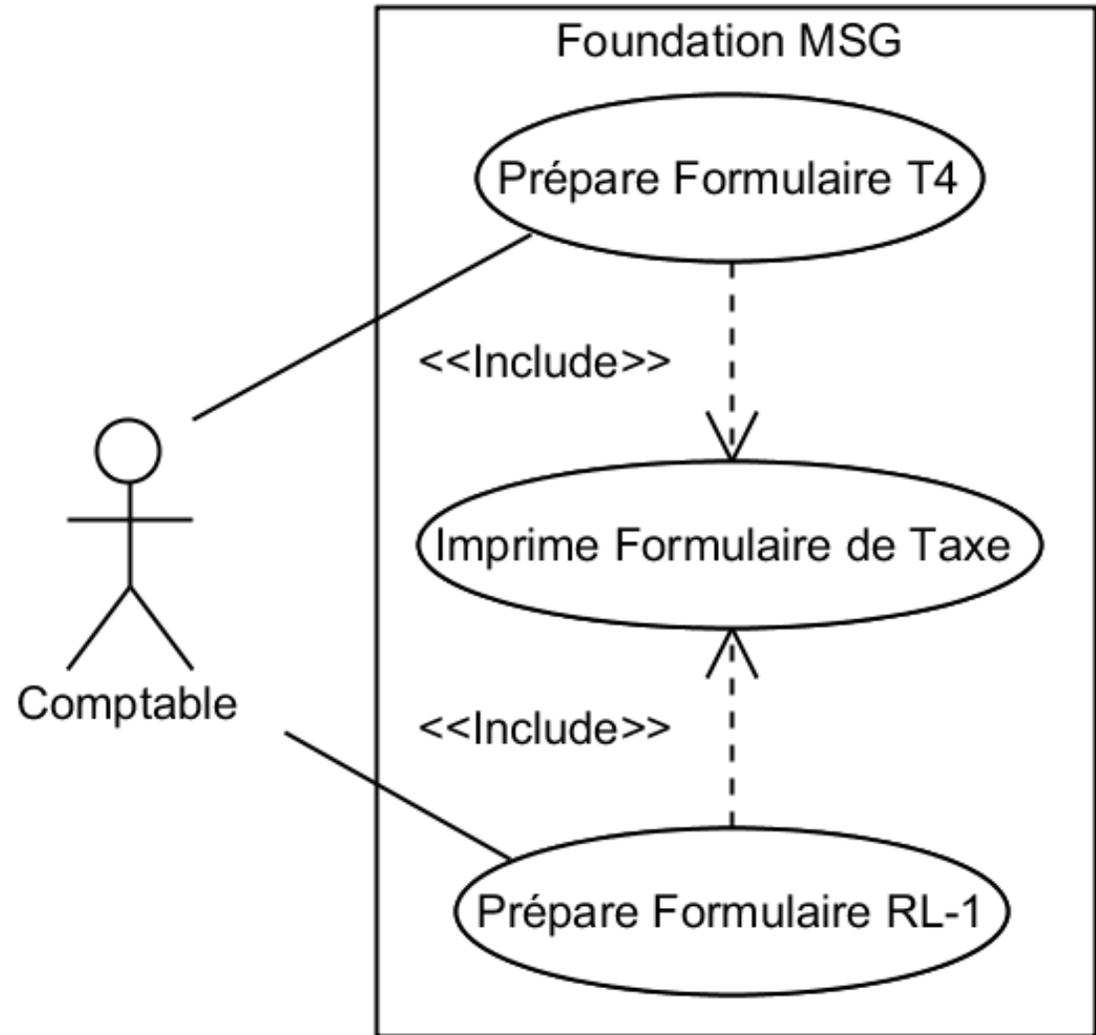
- Décrit sous la forme d'échanges d'évènements entre un **acteur et le système**
- Étapes **ordonnées** numériquement
- Décrit ce qui se déroule **normalement**, quand il n'y a pas d'erreur

Alternatifs

- **Variantes** du scénario principal
 - Alternatives possibles de certaines étapes
- **Exceptions** que le système doit traiter
 - En cas d'erreurs

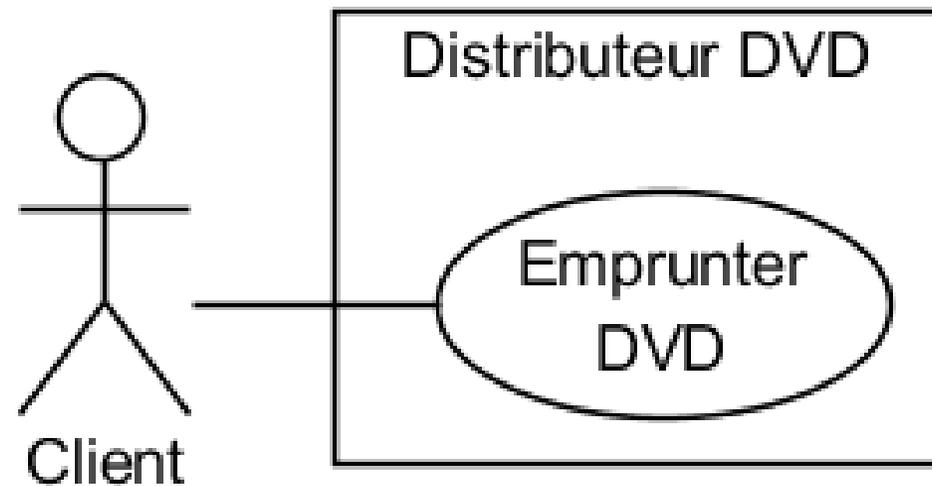
Relation « include »

- On utilise cette relation généralement si un CU est réutilisé dans plus d'un CU
- Sens: du CU de base vers le CU inclus
- Exécution: CU inclus toujours exécuté par le CU de base
- Utilisation: CU de base n'est pas autonome sans le CU inclus



Exemple CU

- **Nom:** Emprunter un DVD
- **But:** Client du magasin emprunte un DVD via le distributeur automatique
- **Acteurs:** Client (principal)
- **Préconditions:** Le client possède une carte qu'il a achetée en magasin. Le distributeur est alimenté en DVDs.



Scénario principal

1. Le client insert sa carte.
2. Le système vérifie la validité de la carte.
3. Le système vérifie que le crédit de la carte est supérieur ou égal à 1 dollar.
4. Appel du cas Rechercher une vidéo.
5. Le client a choisi une vidéo.
6. Le système indique, d'après la valeur de la carte, pendant combien de temps (tranches de 6 heures) le client peut garder le DVD.
7. Le système délivre le DVD.
8. Le client prend le DVD.
9. Le système rend la carte au client.
10. Le client prend sa carte.

Scénario alternatifs

- 2a. Le crédit de la carte est inférieur à 1 dollar.
- 2a.1. Le système indique que le crédit de la carte ne permet pas au client d'emprunter une vidéo.
- 2a.2. Le système invite le client à aller recharger sa carte au magasin.
- 2a.3. Le scénario continue à l'étape 8.

Scénario alternatifs (exceptions)

- 1a. La carte introduite n'est pas valide.
 - 1a.1. Le système indique que la carte n'est pas reconnue.
 - 1a.2. Le distributeur éjecte la carte.

- 7a. Le client ne prend pas le DVD
 - 7a.1. Au bout de 15 secondes le distributeur avale le DVD.
 - 7a.2. Le système annule la transaction. Toutes les opérations mémorisées par le système sont défaites.
 - 7a.3. Le distributeur éjecte la carte.

Postconditions

- Le système a enregistré les informations suivantes :
 - La date et l'heure de la transaction, à la minute près : les tranches de 6 heures sont calculées à la minute près.
 - L'identifiant du client.
 - L'identifiant du DVD emprunté.

Contraintes non-fonctionnelles

- Le distributeur doit fonctionner 24/7 (99% sur le mois).
- La vérification de la validité de la carte doit permettre la détection des contrefaçons à 99.99%.

Activités pour la spécification des exigences

1. Identification des **acteurs**

- Type d'utilisateurs à considérer

2. Identification des **scénarios**

- Observer les utilisateurs dans leurs activités quotidiennes
- Exemple concret par fonctionnalité
- Approfondir notre compréhension du domaine

3. Identification des **cas d'utilisation**

- Généraliser les scénarios pour représenter le système en entier
- Abstraire des scénarios pour décrire tous les cas possibles
- Définit le cadre du système

Activités pour la spécification des exigences

4. **Raffinement** des CU

- S'assurer que les exigences soient complètes
- Description détaillée du comportement
- Gérer les erreurs et conditions particulières/exceptionnelles

5. Identification des **relations** entre les CU

- Trouver les dépendances entre les CU
- Factoriser les fonctionnalités communes
- S'assurer que les exigences soient consistantes

6. Identification des **besoins non fonctionnels**

- S'entendre sur les aspects perceptibles par l'utilisateur
- Performance, documentation, consommation des ressources, qualité

Directives pour rédiger un CU

Les bonnes pratiques 😊

Conventions numériques

- 1, 2, 3, 4: séquençage des étapes
- 1.1, 1.1.1: sous-étapes et sous-sous-étapes
- 3a, 3b: alternatives de l'étape 3, remplace étape 3, commence une fois que l'étape 2 est complétée
- (1-5)a: alternative à n'importe quel moment entre les étapes 1 et 5
- 2a ||: alternative concurrente avec l'étape 2, ne la remplace pas

Guide à suivre pour vos CUs

- **Cas d'utilisation:** nom du CU en expression verbale
- **But:** mise en contexte, description brève
- **Préconditions:** préconditions nécessaires avant d'entamer le scénario principal.
- **Acteurs:** liste des acteurs principaux, puis des acteurs secondaires
- **Scénario principal:** étapes ordonnées numériquement
 - 1, 2, 2.1, 2.2, 2.2.1, ...
- **Scénarios alternatifs:** ramifications du déroulement principal, alternatives possible à certaines étapes, cas d'erreur
 - 1a.1: remplace étape 1 du scénario principal
 - 1b.1: autre remplacement de la même étape
 - 1b.2: suit 1b.1, indépendamment du scénario principal
- **Postconditions:** état du système lorsque le scénario principal achève avec succès