



# Agents intelligents

IFT-17587

Concepts avancés pour systèmes intelligents

Luc Lamontagne

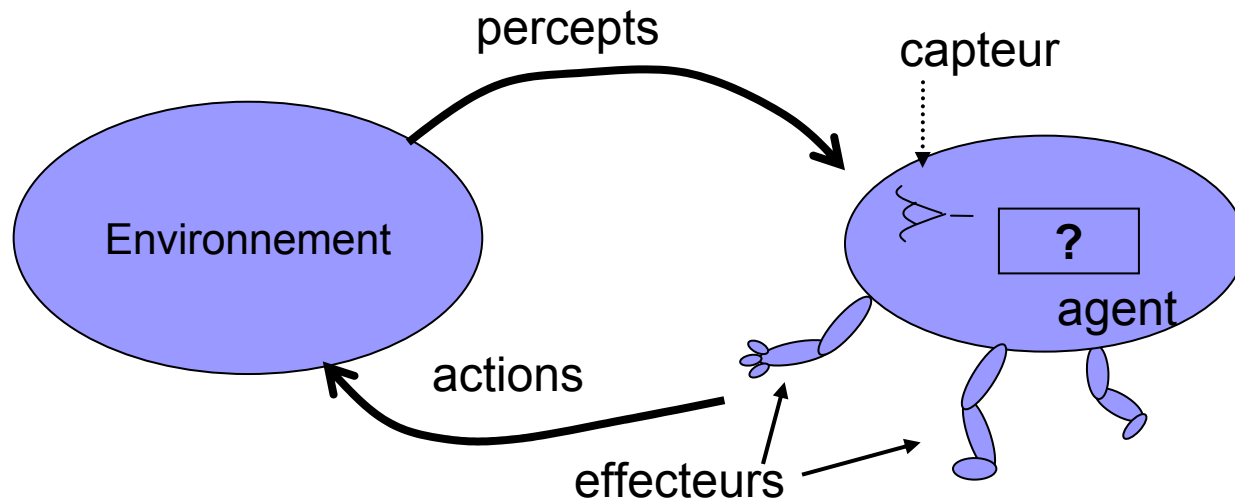
# Plan

- Qu'est-ce qu'un agent intelligent?
- Comment les agents intelligents doivent agir?
- Structure d'un agent intelligent
- Types d'environnements

# Qu'est-ce qu'un agent intelligent?

## ■ Agent intelligent:

- tout ce qui perçoit son environnement
  - à l'aide de ses capteurs
- et qui agit sur son environnement
  - à l'aide de ses effecteurs.



# Agents vs Objets

## ■ Agent

- entité autonome interagissant avec son environnement

## ■ Objet

- entité passive possédant un état et sur lequel on peut effectuer des opérations.

## ■ Un agent → degré d'abstraction plus élevé qu'un objet.

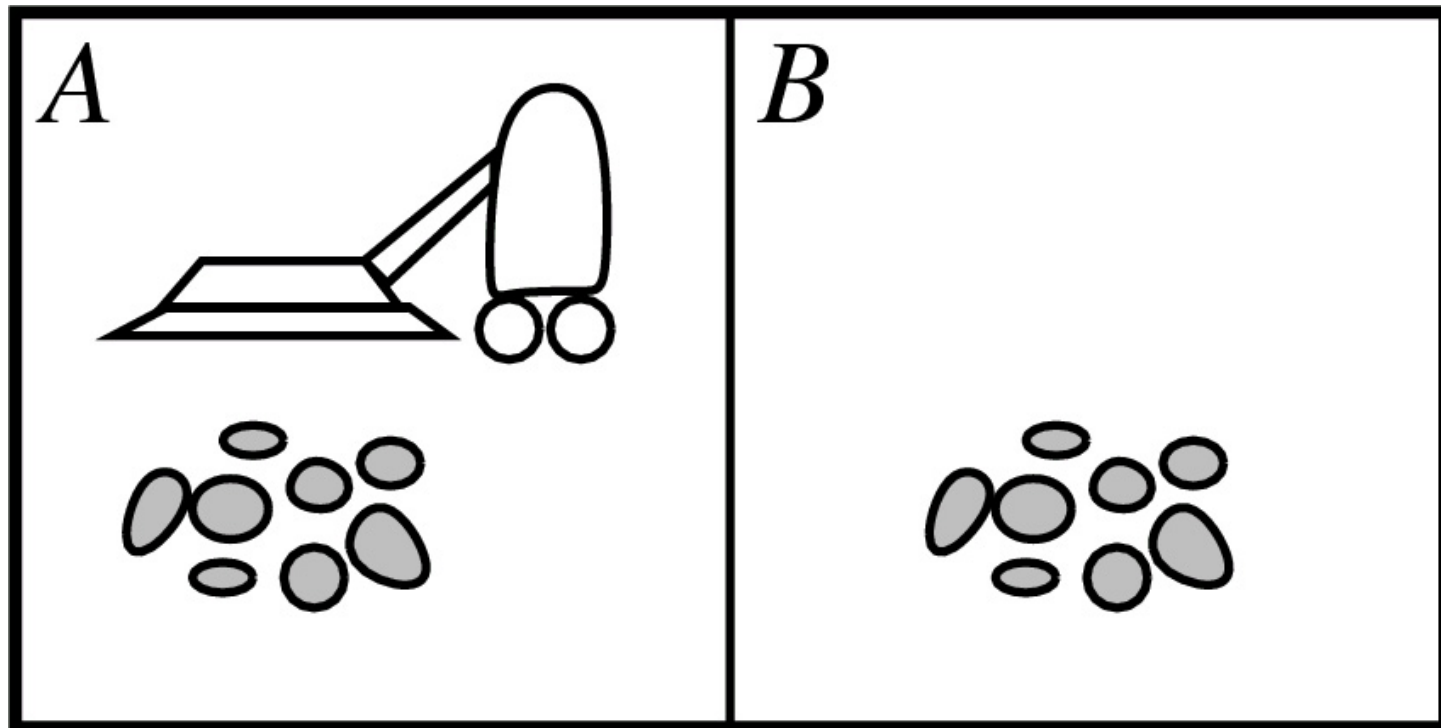
- Un agent peut être constitué de plusieurs objets.
- Un agent logiciel peut être à la fois client et serveur

## ■ Paradigme de programmation mettant en évidence

- l'autonomie
- les interactions (par ex. la programmation orientée agent)

# Exemple très simple d'agent

- Un agent aspirateur



# Tabulation de la fonction d'agent

<b>Percepts</b>	<b>Action</b>
[A, Propre]	Droite
[A, Sale]	Aspire
[B, Propre]	Gauche
[B, Sale]	Aspire
[A, Propre] , [A, Propre]	Aspire
[A, Propre] , [A, Sale]	Aspire
...	
[A, Propre] , [A, Propre] , [A, Propre]	Aspire
[A, Propre] , [A, Propre] , [A, Sale]	Aspire
...	

# Agent rationnel

- Agent rationnel:
  - doit exécuter l'action qui maximise sa mesure de performance
  - en fonction de sa perception du monde et de ses connaissances.
- Mesure de performance
  - Externe
  - Fixée par le concepteur
  - Propre à la tâche
  - Critère de succès du comportement de l'agent

# Choix de la mesure de performance

- Il faut faire attention à ce qu'on choisit
  - Ex: Quantité de saleté ramassé en 8 heures
  - L'agent pourrait ramasser la saleté, la redéposer et ainsi de suite.
- Une meilleure option:
  - Récompenser l'agent pour un plancher propre
  - Ex: un point pour chaque carré propre à chaque intervalle de temps
    - Peut-être avec une pénalité pour l'électricité consommé.



# Agent rationnel

- Rationnel distinct de l'omniscience
  - Tout savoir, connaissance infinie
- L'agent peut accomplir ses actions pour :
  - Accomplir ses buts
  - Modifier sa perception de l'environnement pour obtenir des informations utiles
- Agent autonome
  - Comportement déterminé par ses propres expériences
  - Capacité d'adaptation et d'apprentissage
- Notion d'agent
  - Un outil d'analyse de systèmes
  - non pas une caractérisation absolue qui divise le monde en deux: agents et non-agents.

# Environnement de la tâche

- La première étape lors de la conception d'un agent est de spécifier l'environnement de la tâche (*task environment*) qui contient les quatre éléments suivants (**PEAS**):
  - Mesure de la performance (*Performance measure*)
  - Environnement (*Environment*)
  - Effecteurs (*Actuators*)
  - Capteurs (*Sensors*)

# Exemple

## ■ Conducteur de taxi automatique

- **Mesure de performance**: sécurité, vitesse, légalité, confort, profits, etc.
- **Environnement**: routes, autres autos, piétons, clients, etc.
- **Effecteurs**: volant, accélérateur, frein, clignotant, klaxon, etc.
- **Capteurs**: caméras, sonar, odomètre, indicateur de vitesse, capteurs du moteur, etc.

# Autre exemple

- Système de diagnostic médical
  - **Mesure de performance**: patient en santé, éviter les poursuites, minimiser les coûts
  - **Environnement**: patient, hôpital, personnel
  - **Effecteurs**: l'écran du système
    - questions, tests, diagnostic, traitements, références
  - **Capteurs**: clavier
    - saisie des symptômes, évidences, réponses du patient

# Propriétés de l'environnement

- Complètement observable vs partiellement observable
- Déterministe vs stochastique
- Épisodique vs séquentielle
- Statique vs dynamique
- Discret vs continu
- Un agent vs plusieurs agents

# Propriétés de l'environnement

- **Complètement observable vs partiellement observable:**
  - Est-ce que les capteurs de l'agent lui donne accès à l'état complet de l'environnement à tout moment ?
- **Déterministe vs stochastique:**
  - Est-ce que le prochain état de l'environnement est complètement déterminé par son état courant et l'action de l'agent ?
- **Épisodique vs séquentielle:**
  - Un épisode est une séquence perception-action.
  - Le prochain épisode ne dépend pas des actions effectuées dans les épisodes précédents.

# Propriétés de l'environnement

- **Statique vs dynamique:**
  - Est-ce que l'environnement change pendant que l'agent délibère?
- **Discret vs continu :**
  - La distinction entre discret et continu peut être appliquée
    - à l'état de l'environnement
    - à la façon dont le temps est géré
    - et aux perceptions et aux actions de l'agent.
- **Un agent vs multi-agents :**
  - Est-ce qu'il y a plus qu'un agent interagissant ensemble ?

# Propriétés de l'environnement

- Situation la plus difficile:
  - Partiellement observable
  - Stochastique
  - Séquentielle
  - Dynamique
  - Continue
  - Multiagent
- Exemple, conduite automatisée d'un taxi



# Exemple

Environnement	Observable	Déterministe	Épisodique	Statique	Discret	Agents
Mots-croisés	Complètement	Déterministe	Séquentiel	Statique	Discret	Un
Échec avec une horloge	Complètement	Stratégique	Séquentiel	Semi	Discret	Multi
Poker	Partiellement	Stratégique	Séquentiel	Statique	Discret	Multi
Backgammon	Complètement	Stochastique	Séquentiel	Statique	Discret	Multi
Conduire un taxi	Partiellement	Stochastique	Séquentiel	Dynamique	Continu	Multi
Diagnostic médical	Partiellement	Stochastique	Séquentiel	Dynamique	Continu	Un
Analyse d'image	Complètement	Déterministe	Épisodique	Semi	Continu	Un
Robot ramasseur de pièces	Partiellement	Stochastique	Épisodique	Dynamique	Continu	Un
Contrôleur de raffinerie	Partiellement	Stochastique	Séquentiel	Dynamique	Continu	Un
Enseignant interactif	Partiellement	Stochastique	Séquentiel	Dynamique	Discret	Multi

# Fonctions et programmes d'agent

- Un agent est complètement spécifié par une fonction d'agent
  - Mise en correspondance des percepts et des actions
- But : trouver un moyen d'implanter cette fonction de manière concise
- Problème des tables de fonction :
  - Immense
  - Prends un temps énorme à construire
  - Aucune autonomie
  - Même avec apprentissage, nécessite un temps considérable pour apprendre les éléments du tableau

# Structure d'un agent

- 4 types de base en ordre de généralité grandissante:
  - Agent simple réflexe
  - Agent réflexe avec état interne
  - Agent basé sur les buts
  - Agent basé sur l'utilité
- Tous ces types peuvent être transformés en agents apprenants.

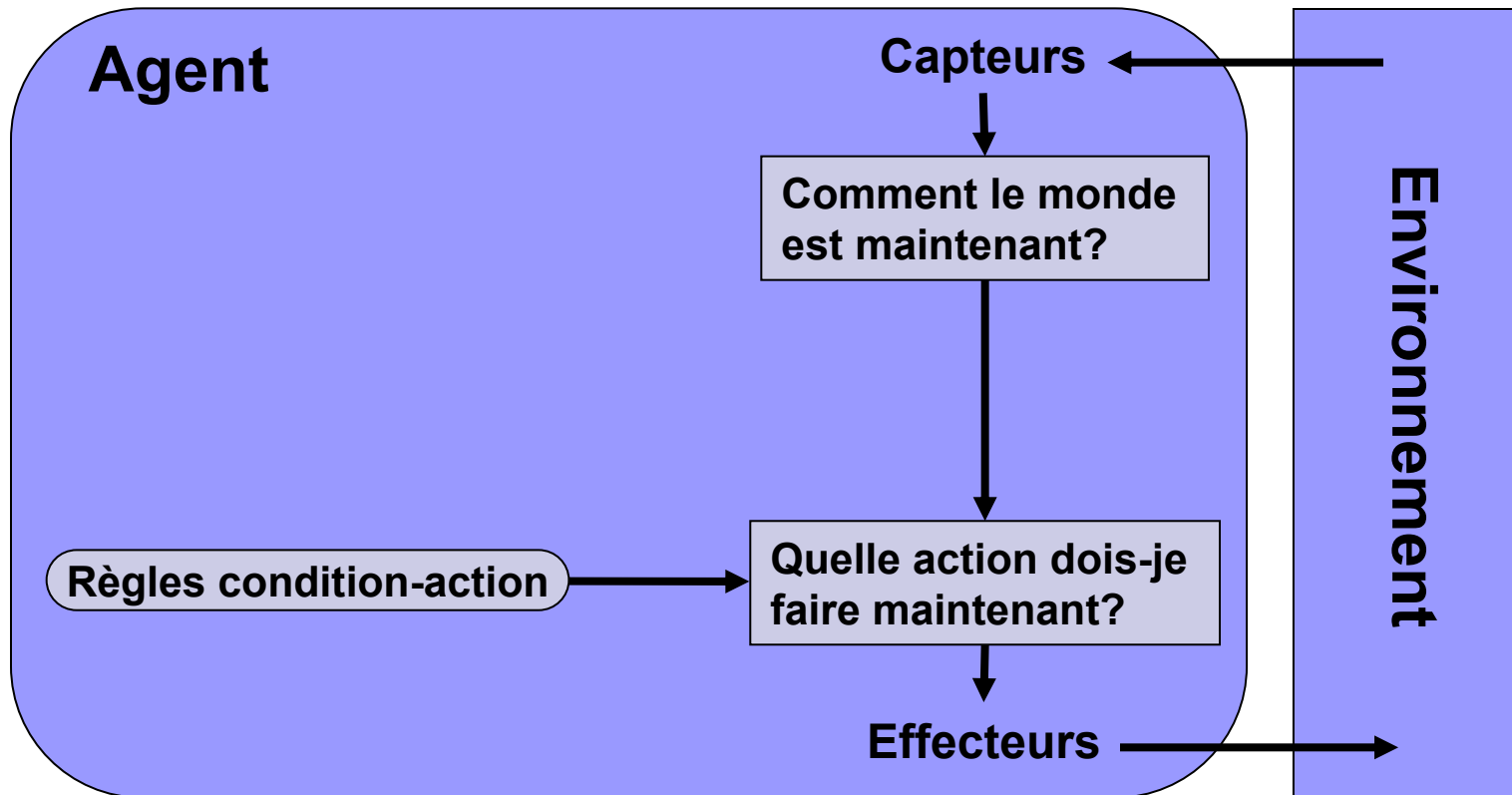
# Agent simple réflexe

- Ce type d'agent choisit ses actions en se basant uniquement sur le percept courant, en ignorant les percepts précédents.

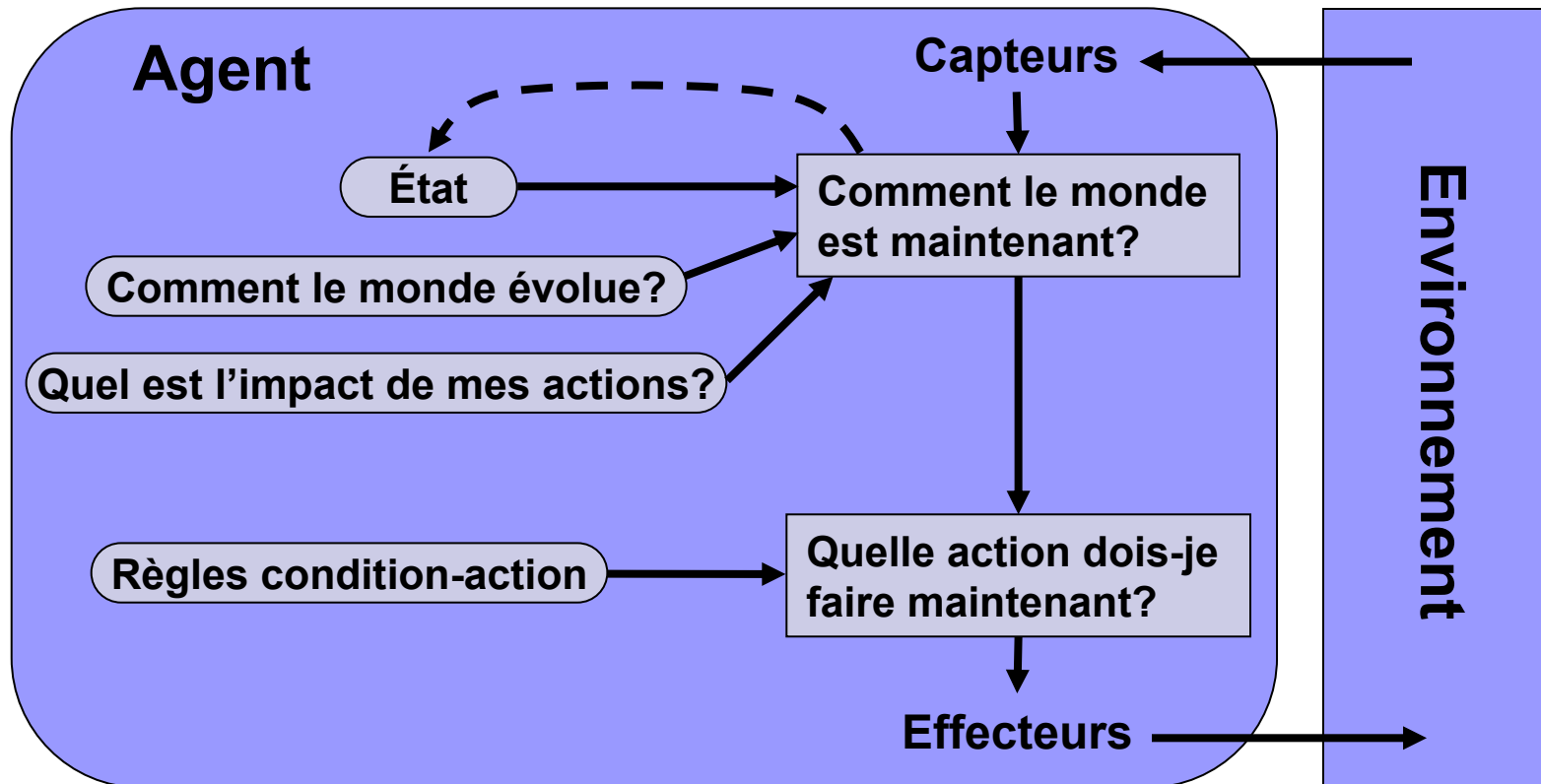
**function** REFLEX-VACUUM-AGENT(*[location, status]*) **returns** an action

*if status = Dirty then return Suck*  
*else if location = A then return Right*  
*else if location = B then return Left*

# Agent simple réflexe

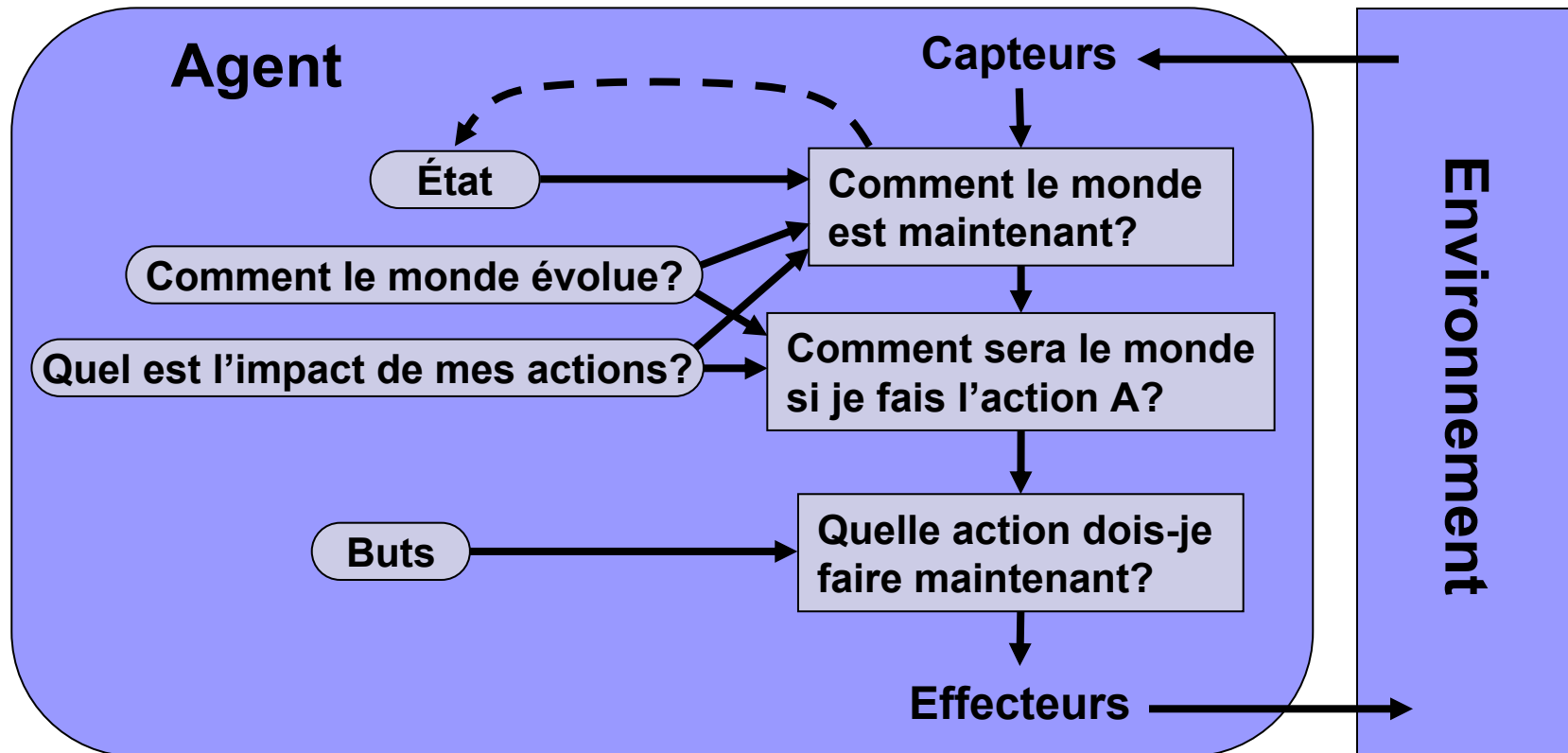


# Agent réflexe avec état interne



*Model-based reflex agent*

# Agent basé sur les buts



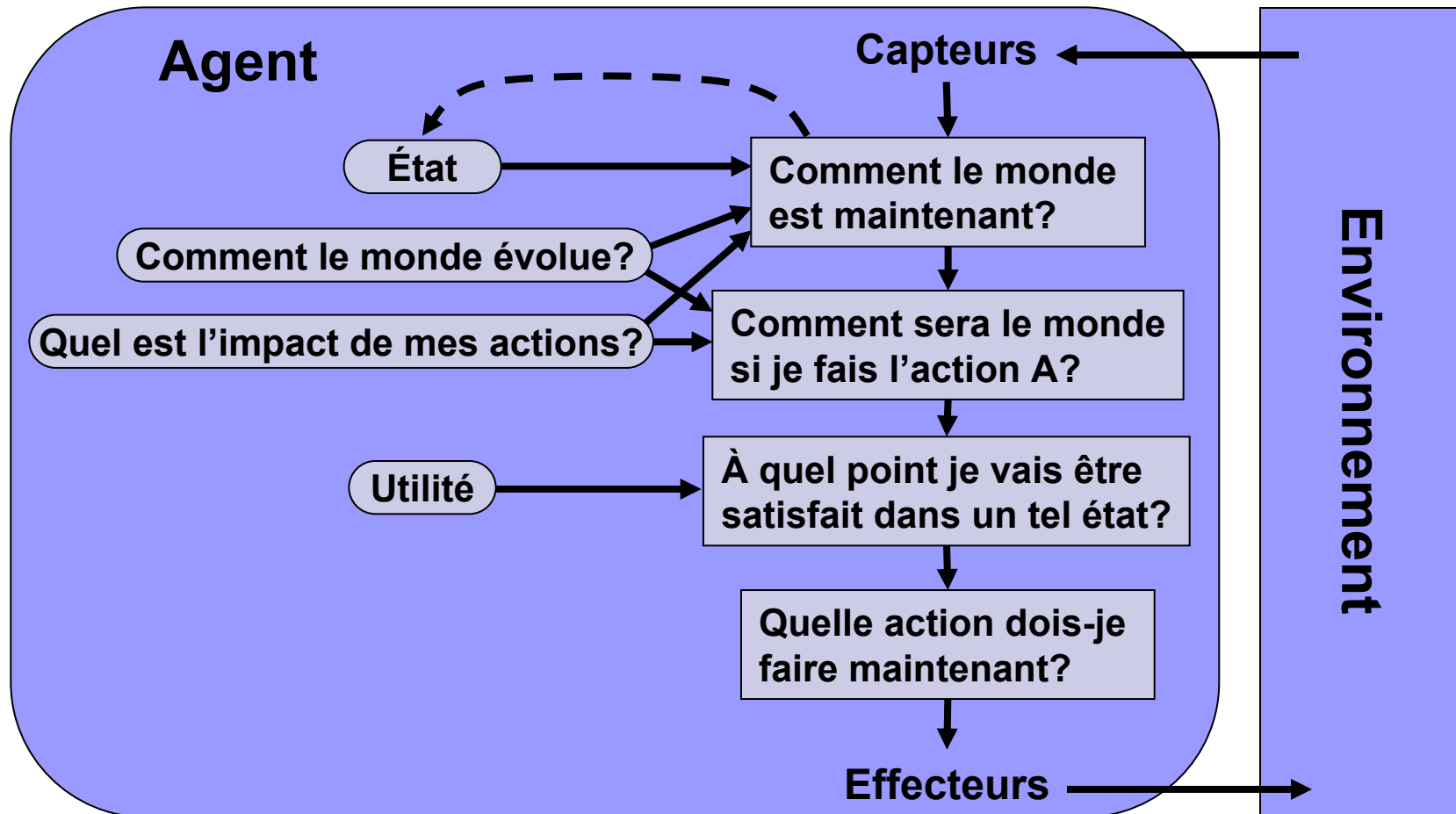
*Goal-based agent*

# Agent basé sur l'utilité

- Les buts ne font la distinction que entre un état « heureux » ou « pas heureux ».
- Fonction d'utilité: état  $\rightarrow$  valeur
- Aide dans deux cas où les buts échouent:
  - Buts en conflits (ex: vitesse et sécurité)
  - Lorsqu'il y a plusieurs buts



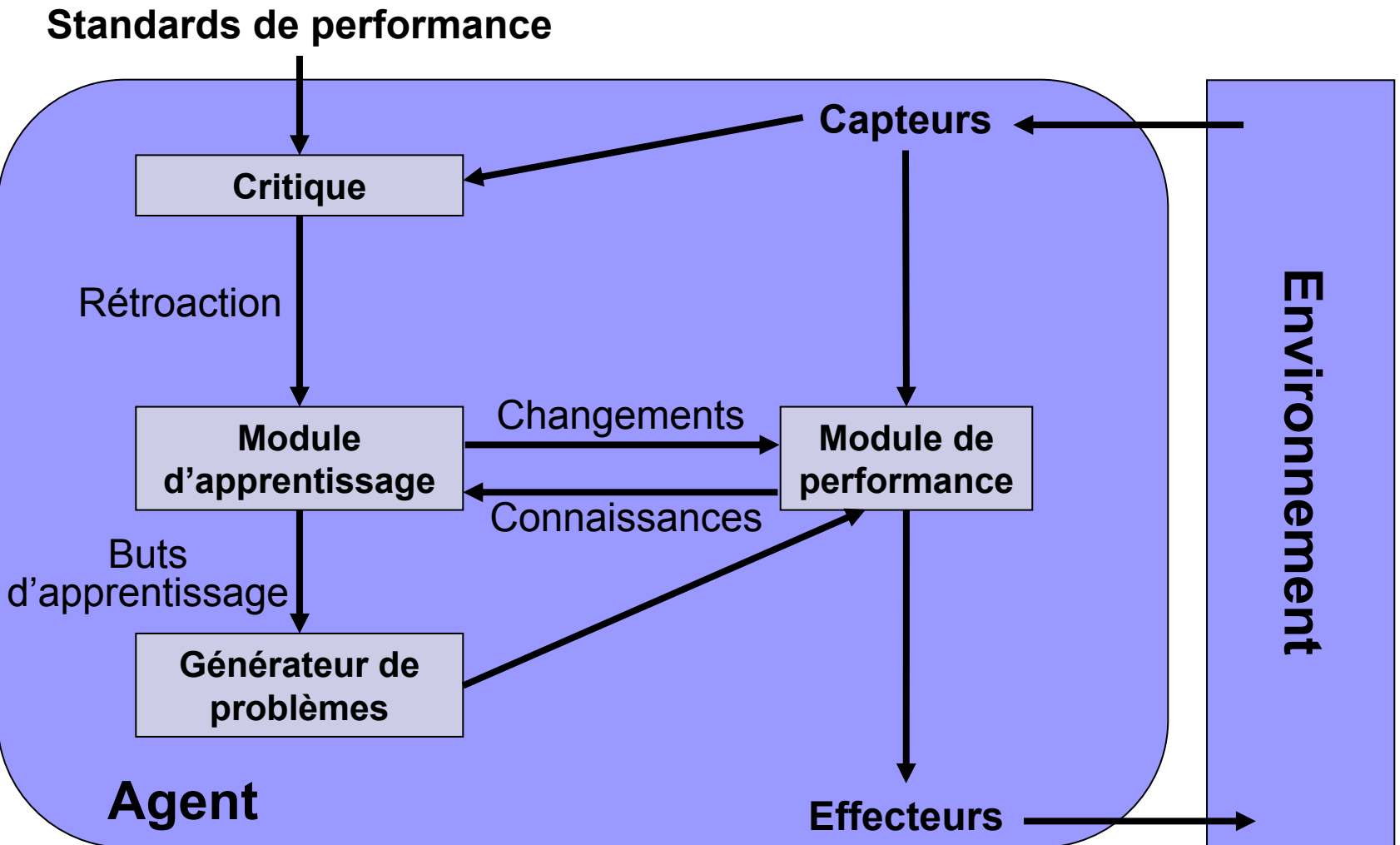
# Agent basé sur l'utilité



# Agent apprenant

- Souvent, il est très fastidieux où même impossible de définir le comportement de l'agent à la conception.
- L'apprentissage permet:
  - De simplifier la conception
  - À l'agent d'avoir plus de flexibilité
  - À l'agent d'agir dans des environnements inconnus et de devenir meilleur avec le temps.

# Agent apprenant



# Agent taxi apprenant

- Module de performance
  - Connaissances et procédures pour choisir les actions.
- Critique
  - Observe l'agent et donne des informations au module d'apprentissage.
- Module d'apprentissage
  - Modifie le module de performance.
- Générateur de problèmes
  - Identifie les possibilités d'amélioration et suggère des expérimentations.

# Conclusion

- Un agent est un artefact (robot, logiciel...) qui perçoit et agit sur un environnement
- La mesure de performance évalue le succès du comportement de l'agent
- L'environnement de tâches varie selon différentes dimensions.
- Le programme d'un agent est une implémentation de la fonction d'agent.
- 4 agents, aux fonctionnalités croissantes, basé sur les :
  - Réflexes, les modèles, les buts et l'utilité
- Chacun peut être amélioré par l'apprentissage.