

# **IFT 615 – Intelligence artificielle**

## **Agents intelligents**

Fro duald Kabanza et Hugo Larochelle  
[planiart.usherbrooke.ca/cours/ift615](http://planiart.usherbrooke.ca/cours/ift615)



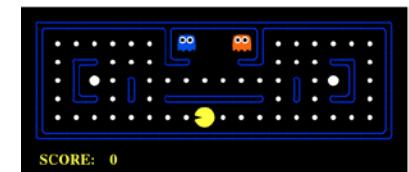
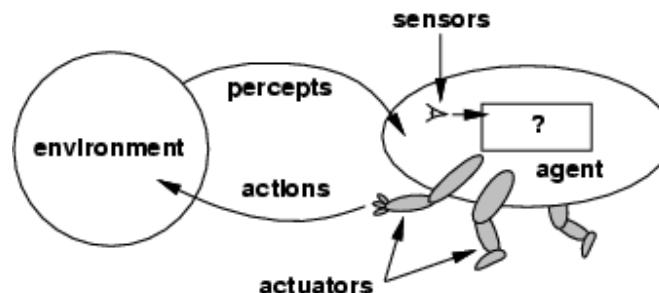
# Sujets couverts

- Agents intelligents
- Rationalité
- Modèle générique PEAS de conception des agents
  - ◆ mesure de **Performance**, modélisation de l'**Environnement**, et l'implémentation des **Actionneurs** ainsi que des **Senseurs**
- Types d'environnements
  - ◆ Déterministe, stochastique, etc.
- Types d'agents
  - ◆ Reflex, orienté-but, orienté-utilité, etc.
- Exemple – Le monde des wumpus (Wumpus world)

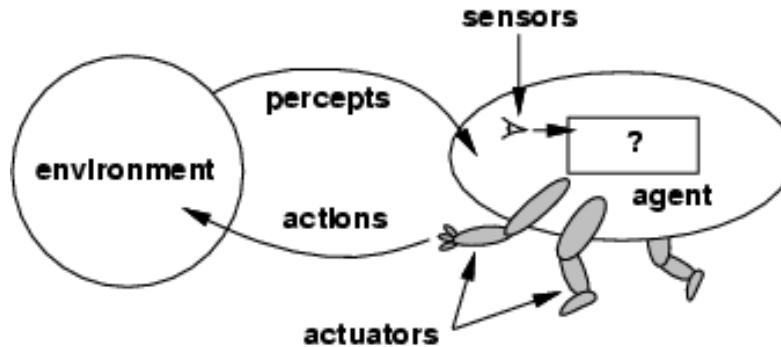


# C'est quoi un agent?

- Un agent est n'importe quel entité qui perçoit son environnement par des **capteurs** (*sensors*) et agit sur cet environnement par des **actionneurs** (*actuators*)

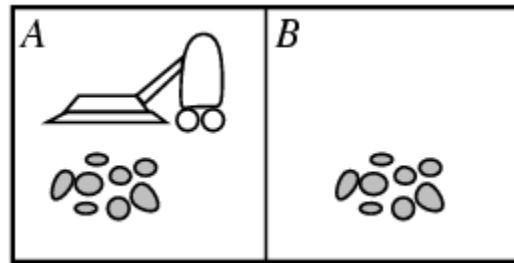


# Fonction mathématique « agent »



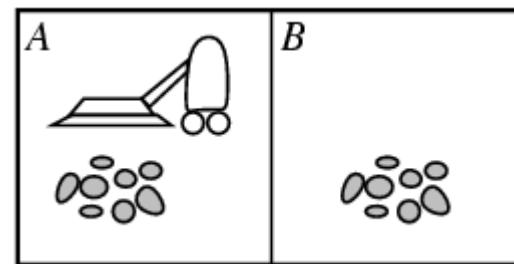
- La **fonction agent**  $f$  prend en entrée une séquence d'**observations** (percepts) et retourne une **action** :  
$$f : P^* \rightarrow A$$
- En pratique la fonction est implémentée par un programme sur une architecture matérielle particulière

# Exemple : Aspirateur robotisé



- Observations (données sensorielles) : position et état des lieux  
Par exemple : [A,Clean],  
[A,Dirty],  
[B,Clean],
- Actions : *Left, Right, Suck, NoOp*

# Exemple : Aspirateur robotisé



- $f$ :
  - $[A, Clean] \rightarrow Right$
  - $[A, Dirty] \rightarrow Suck$
  - ...
  - $[A, Clean] [A, Clean] [A, Dirty] \rightarrow Suck$
  - $[A, Clean] [A, Clean] [A, Clean] \rightarrow Right$
  - ...

# Ébauche d'un agent

```
function SKELETON-AGENT(percept) returns action
  static: memory, the agent's memory of the world
    memory  $\leftarrow$  UPDATE-MEMORY(memory, percept)
    action  $\leftarrow$  CHOOSE-BEST-ACTION(memory)
    memory  $\leftarrow$  UPDATE-MEMORY(memory, action)
  return action
```



# Agents rationnels

- Un agent rationnel doit agir « correctement » en fonction de ce qu'il perçoit et de ses capacités d'action :
  - ◆ l'**action correcte** est celle permettant à l'agent de réussir le mieux
- **Mesure de performance** :
  - ◆ une fonction objective mesurant la qualité d'un comportement de l'agent
- Par exemple, une mesure de performance pour un robot aspirateur pourrait être :
  - ◆ la quantité de déchets aspirés
  - ◆ la propreté des lieux
  - ◆ la durée de la tâche
  - ◆ le bruit généré
- **Agent rationnel** : étant donné une séquence d'observations (données sensorielles) et des connaissances propres, un agent rationnel devrait **choisir une action qui maximise la mesure de performance**



# Agents rationnels

- **Rationalité ne veut pas dire « qui sait tout »**  
(par exemple, connaît tous les effets de ses actions)!
- **Rationnel ne veut pas dire « parfait »**
  - ◆ la rationalité maximise la performance escomptée
  - ◆ la perfection maximise la performance réelle
  - ◆ mais souvent on ne peut pas connaître la performance réelle avant l'action
- Un agent peut effectuer des actions d'observation pour cueillir des informations nécessaires à sa tâche
- Un agent est **autonome** s'il est capable d'adapter son comportement aux changements dans l'environnement (capable d'apprendre, de planifier, de raisonner)

# Modèle PEAS

- PEAS : Un modèle générique de conceptions des agents par la spécification des composantes suivantes :
  - ◆ mesure de **performance**
  - ◆ éléments de l'**environnement**
  - ◆ les **actions** que l'agent peut effectuer (**Actionneurs**)
  - ◆ la séquence des **observations** ou **percepts** de l'agent (**Senseurs**)
- **PEAS = *Performance, Environnement, Actuateurs, Senseurs***

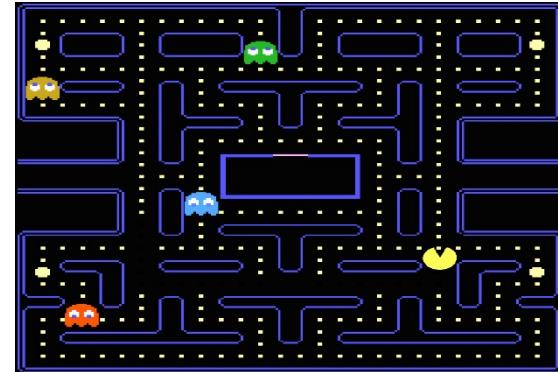
# Exemple : Modèle PEAS pour voiture autonome

- **Agent** : Voiture autonome
- **Mesure de performance** : sécurité, vitesse, respect du code routier, voyage confortable, maximisation des profits (pour un taxi)
- **Environnement** : route, trafic, piétons, clients
- **Actionneurs** : volant, changement de vitesse, accélérateur, frein, clignotants, klaxon
- **Senseurs** : caméras, sonar, GPS, odomètre, compteur de vitesse, témoins du moteur, etc.



# Exemple : Modèle PEAS pour Pacman

- **Agent** : Pacman
- **Mesure de performance** : score
- **Environnement** : le labyrinthe, les biscuits, les fantômes
- **Actionneurs** : se déplacer, manger, crier
- **Senseurs** : senseur de fantômes, senseur de biscuits, senseur pour la position,





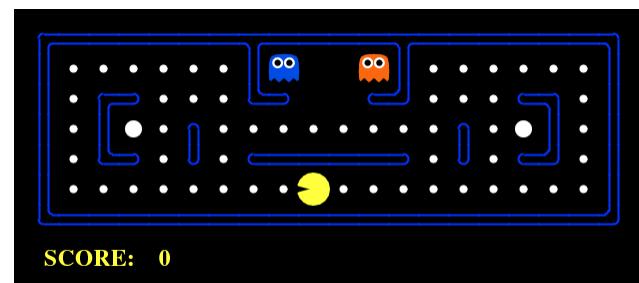
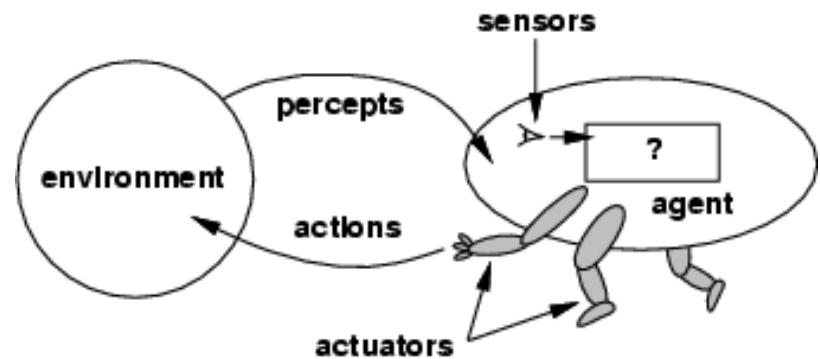
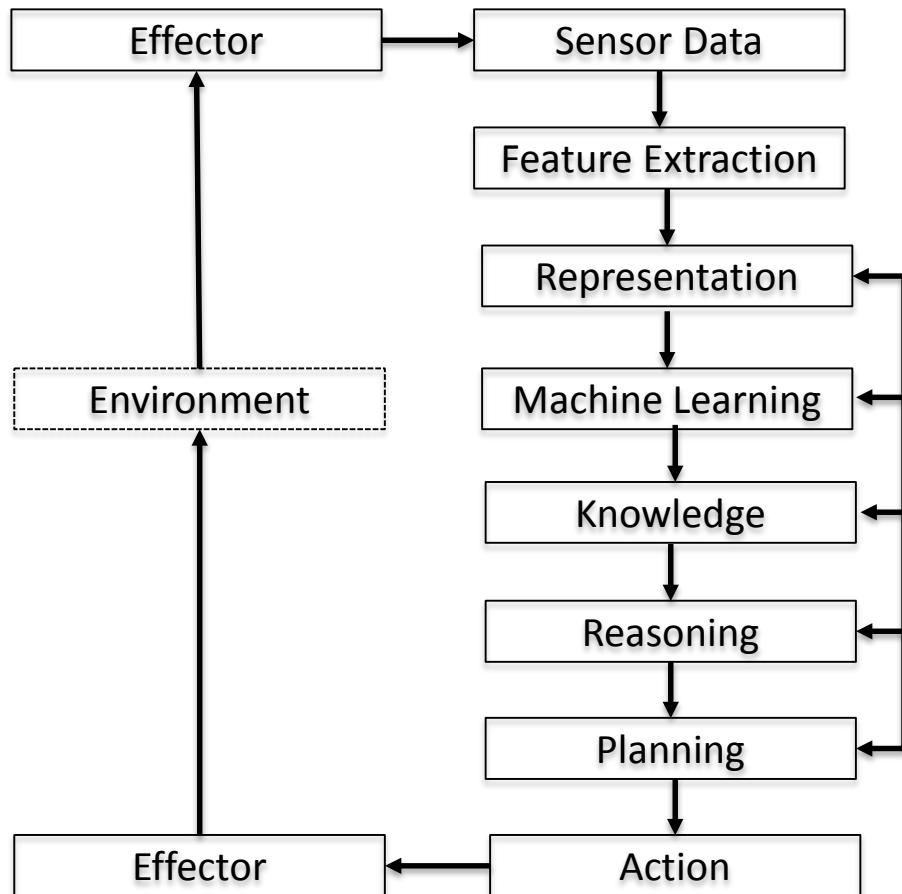
# Caractéristiques d'environnement

- Différents problèmes auront des environnements avec des caractéristiques différentes
- Caractéristiques que l'on distingue:
  - ◆ **Complètement observable** (vs. partiellement observable)
  - ◆ **Déterministe** (vs. stochastique)
  - ◆ **Épisodique** (vs. séquentiel)
  - ◆ **Statique** (vs. dynamique)
  - ◆ **Discret** (vs. continu)
  - ◆ **Agent unique** (vs. multi-agent)

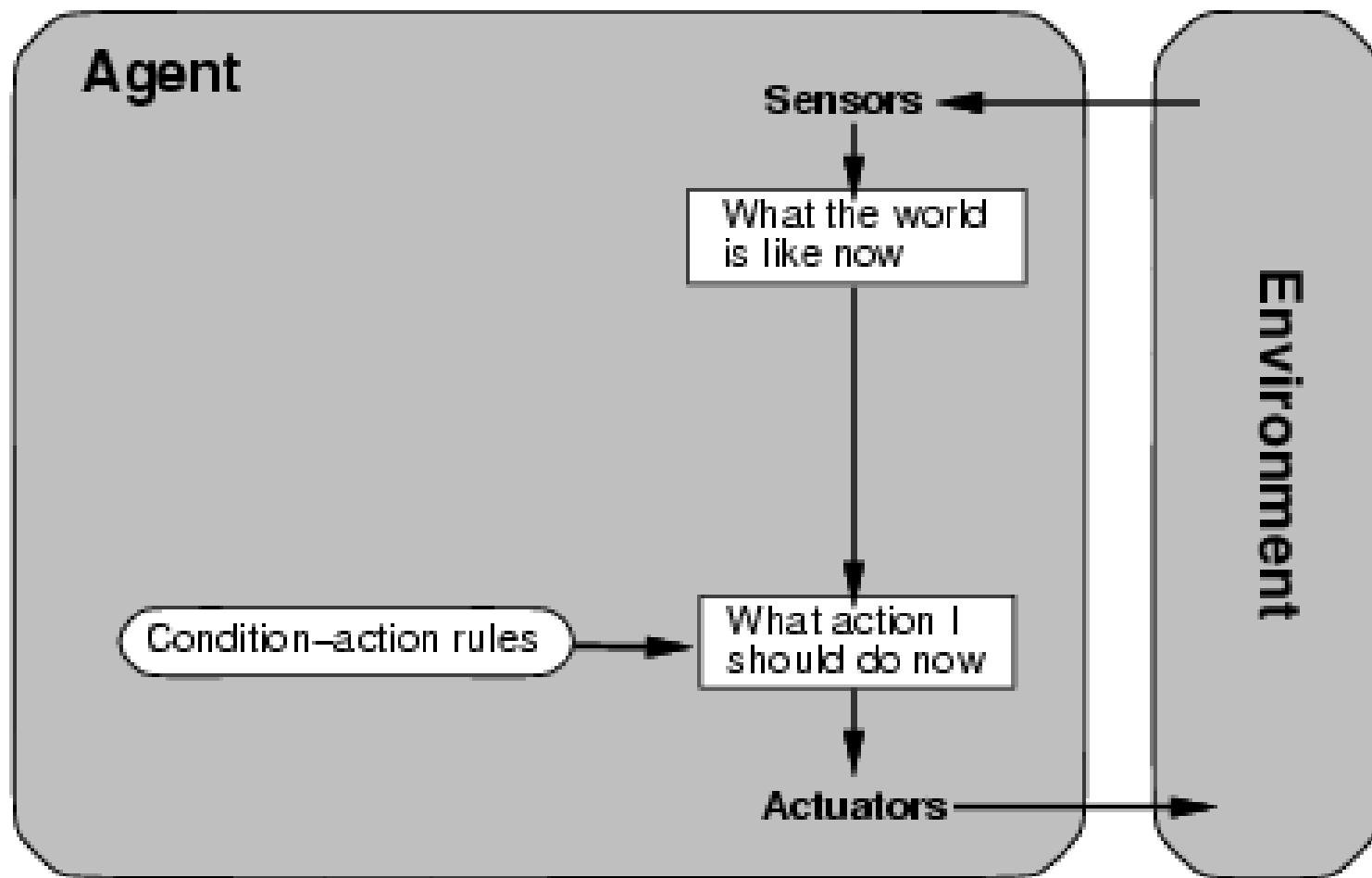
# Architectures des agents

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents

# Rappel - Problèmes à résoudre



# *Simple reflex agents*

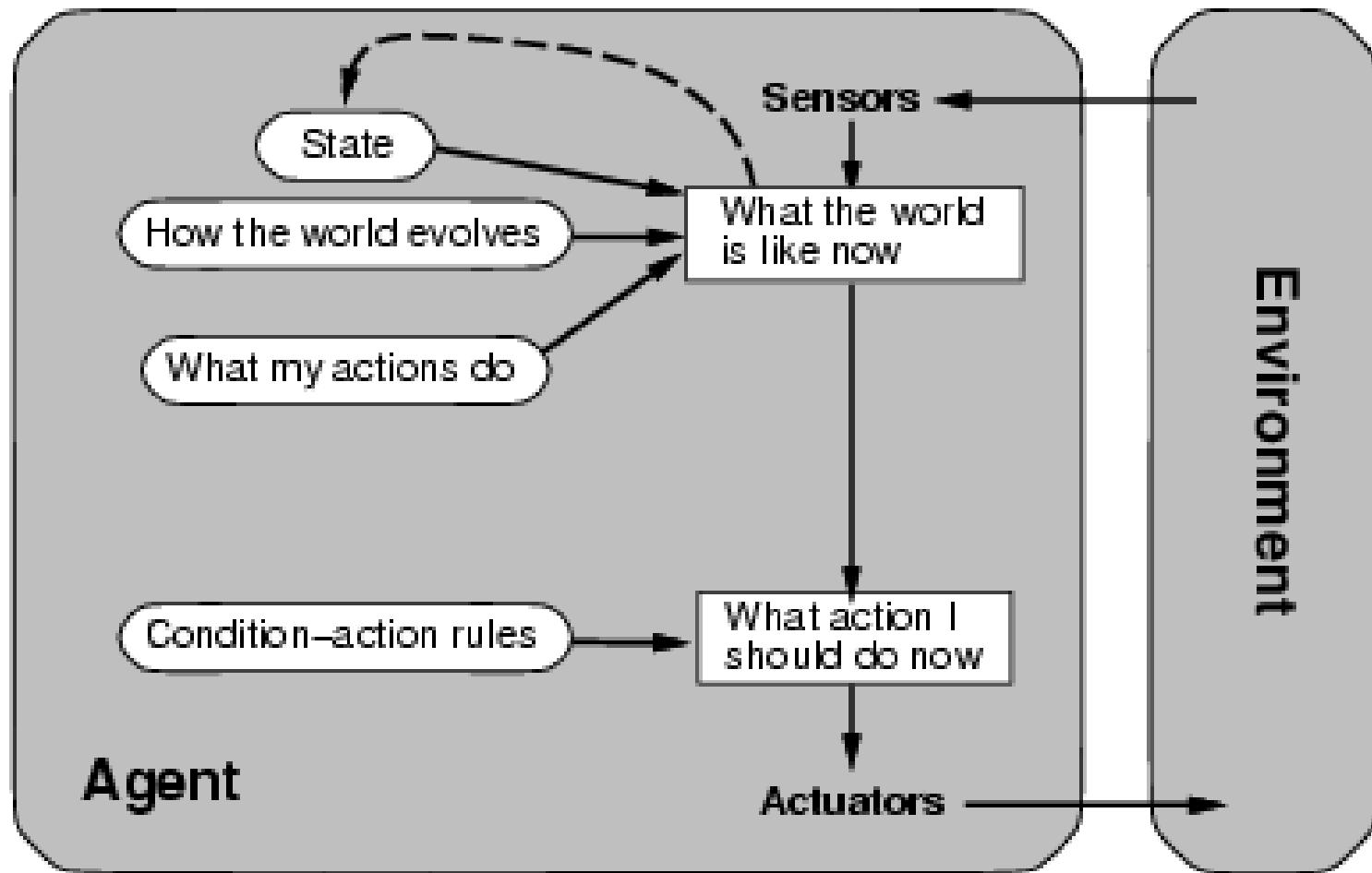


# *Simple reflex agents*

```
function SIMPLE-REFLEX-AGENT(percept) returns an action
  persistent: rules, a set of condition-action rules

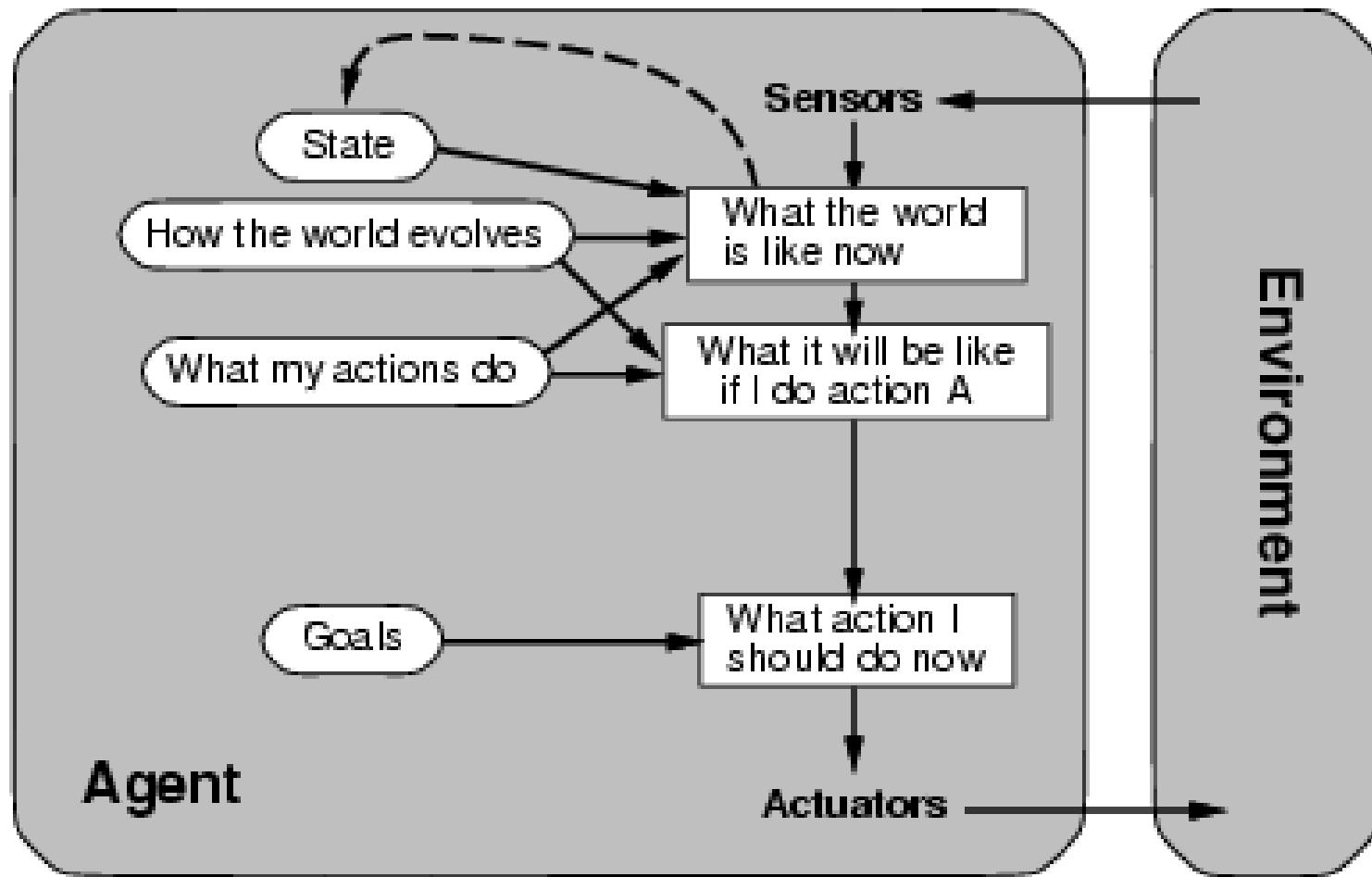
  state  $\leftarrow$  INTERPRET-INPUT(percept)
  rule  $\leftarrow$  RULE-MATCH(state, rules)
  action  $\leftarrow$  rule.ACTION
  return action
```

# *Model-based reflex agents*

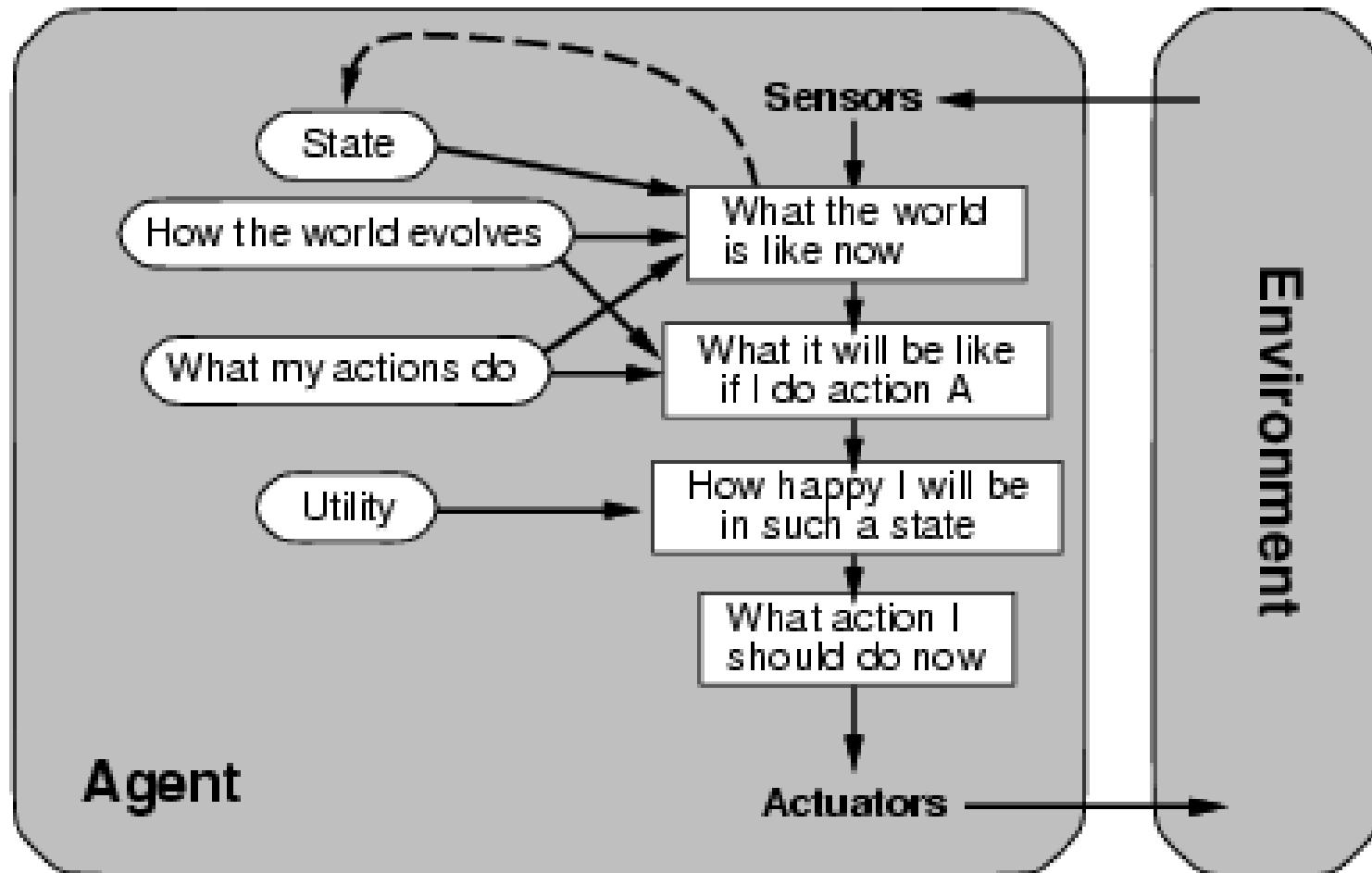




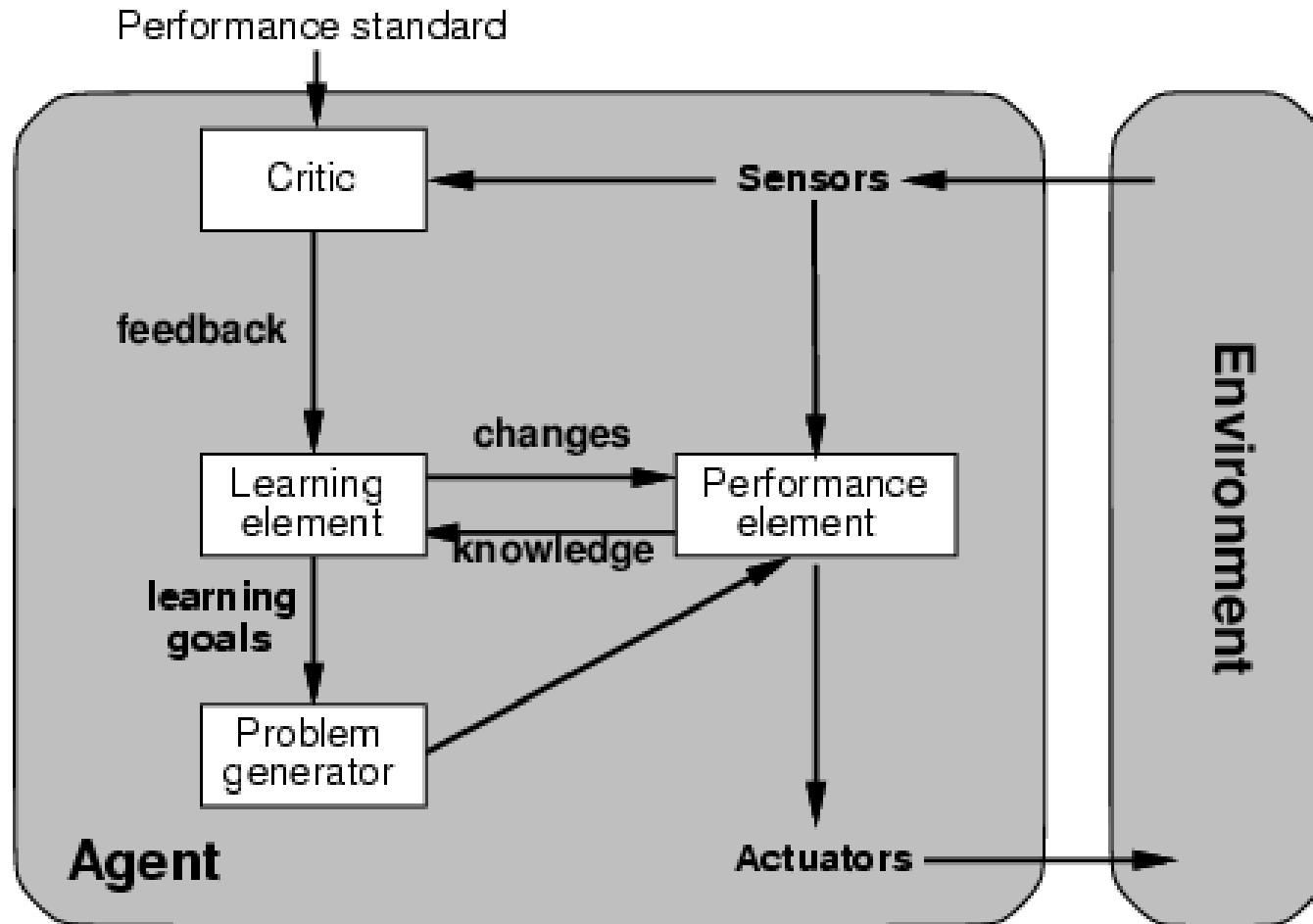
# Goal-based agents



# *Utility-based agents*

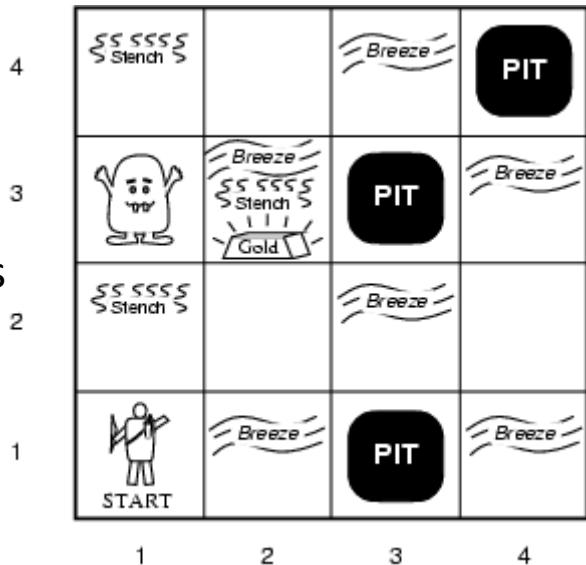


# Learning agents



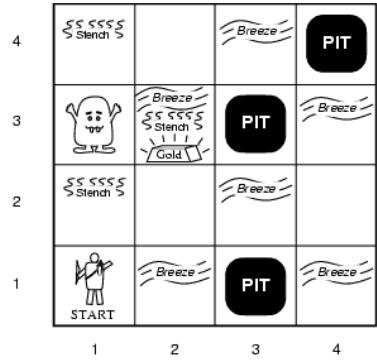
# Exemple : le monde des wumpus (Section 7.2)

- Mesure de performance
  - ◆ or +1000, mort -1000
  - ◆ -1 par pas, -10 pour une flèche
- Environnement
  - ◆ puanteur dans les chambres adjacentes au wumpus
  - ◆ brise dans les chambres adjacentes à une fosse
  - ◆ scintillement si l'or est dans la chambre
  - ◆ le wumpus meurt si on lui tire une flèche de face
  - ◆ on a une seule flèche
  - ◆ on peut ramasser l'or dans la même chambre
  - ◆ on sort de la grotte en grimpant à la case [1,1]
- Capteurs: *Stench* (puanteur), *Breeze* (brise), *Glitter* (scintillement), *Bump* (choc), *Scream* (cri).
- Actionneurs : *Left turn*, *Right turn*, *Forward*, *Grab*, *Climb*, *Shoot*

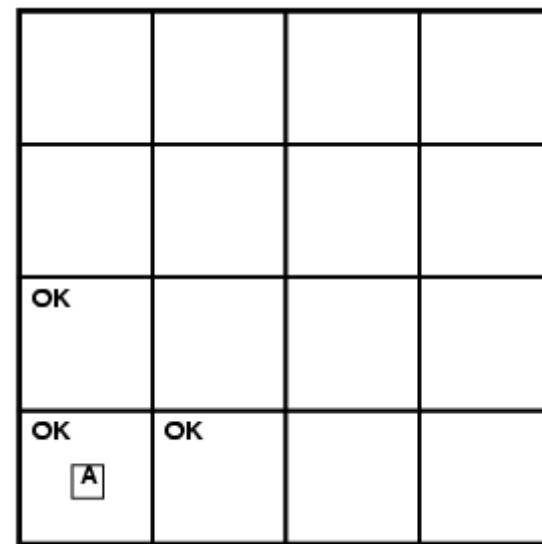


# Caractéristiques du monde des wumpus

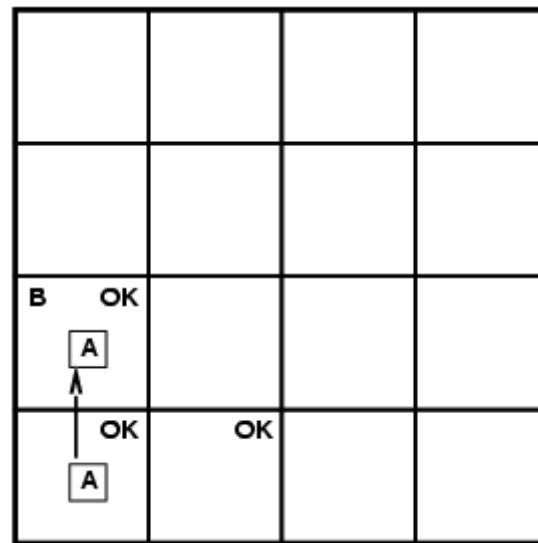
- Complètement observable? Non – seulement perception locale.
- Déterministe? Oui – l'effet de chaque action est prévisible.
- Épisodique? Non – séquentiel au niveau des actions.
- Statique? Oui – le wumpus et les fosses ne bougent pas.
- Discret? Oui.
- Agent unique? Oui – La seule action du wumpus est de nous « bouffer » si on atteint sa chambre.



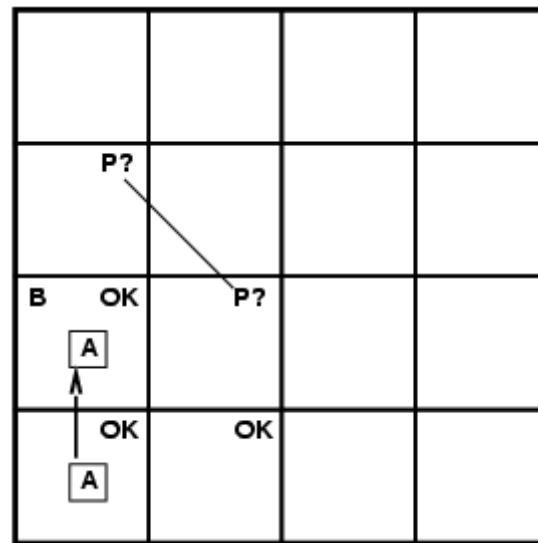
# Exploration du monde des wumpus



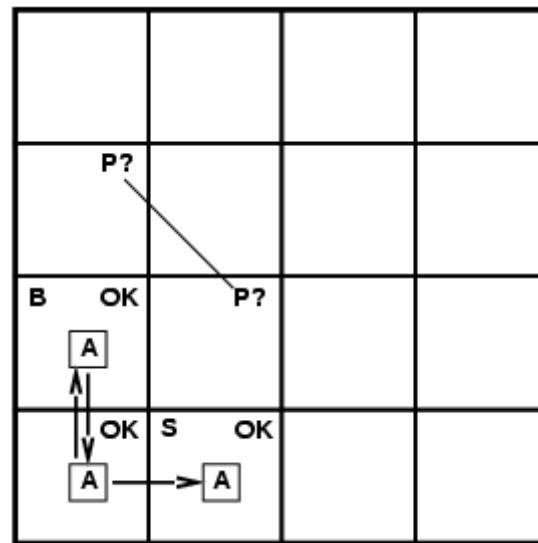
# Exploration du monde des wumpus



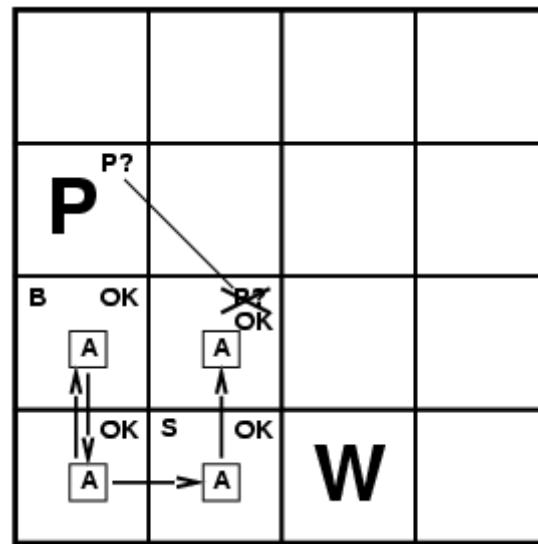
# Exploration du monde des wumpus



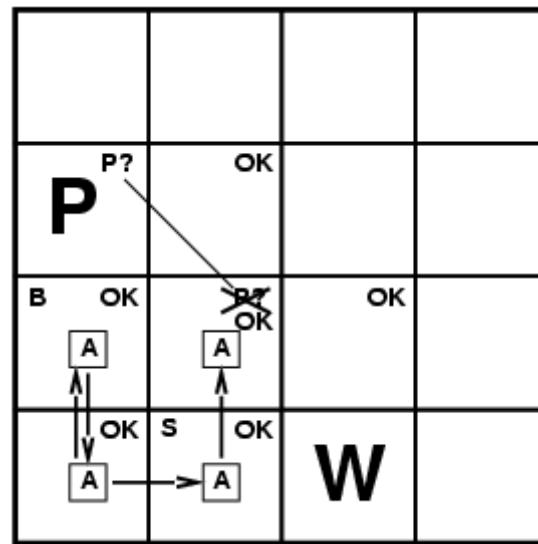
# Exploration du monde des wumpus



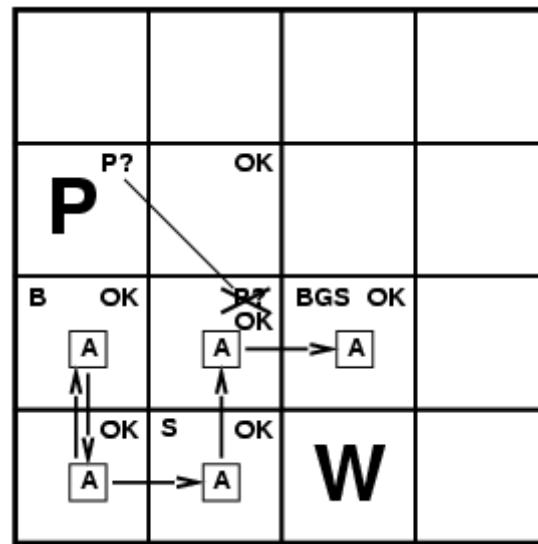
# Exploration du monde des wumpus



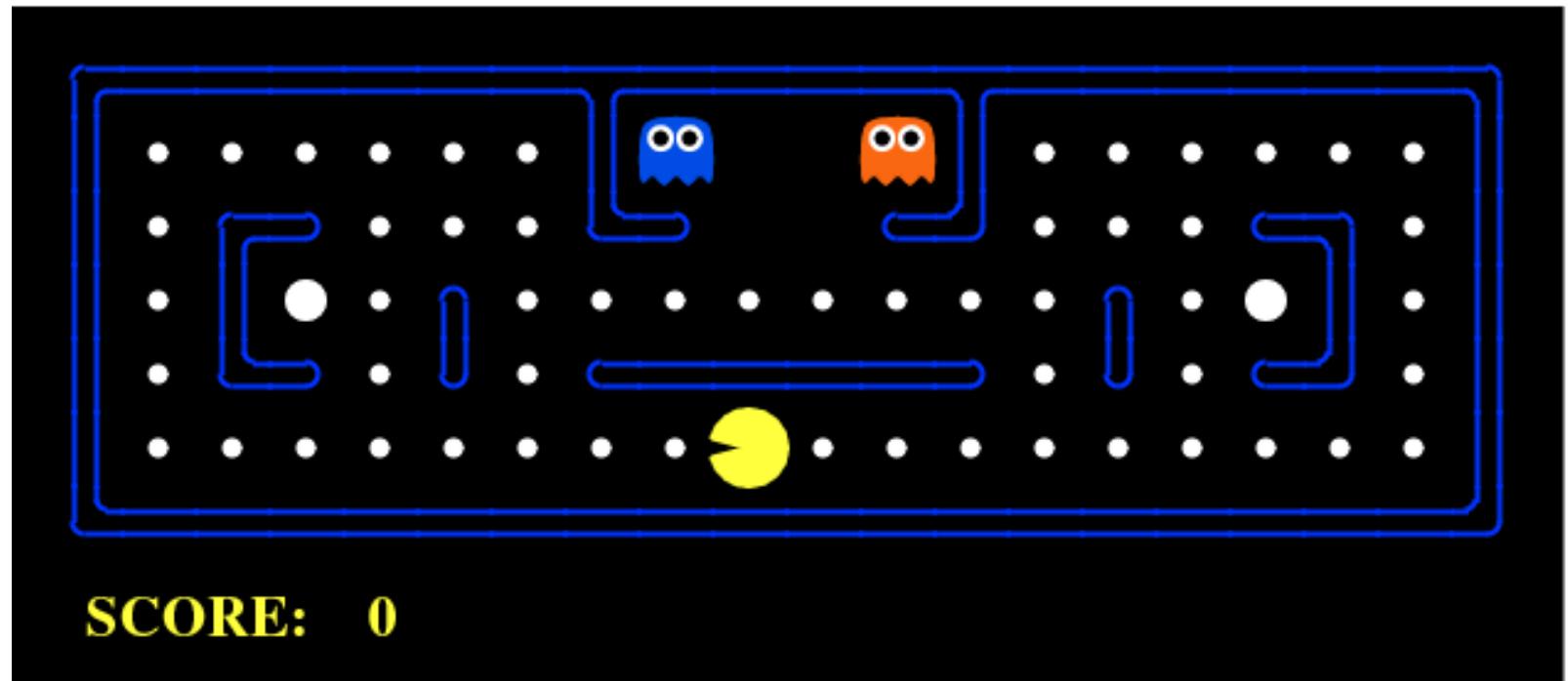
# Exploration du monde des wumpus



# Exploration du monde des wumpus



# Exercice – Quel type d'environnement pour Pacman?



# Conclusion

- En résumé, l'intelligence artificielle s'intéresse à tout sujet qui permettrait de reproduire toute capacité de l'intelligence humaine
- Un agent est quelque chose qui perçoit et agit sur son environnement
- Idéalement, on aimerait concevoir un agent rationnel
  - ◆ par rationnel, on veut dire qui maximise sa performance espérée (moyenne)
- L'espace des agents possibles est très large
  - ◆ dépend de la tâche à résoudre
  - ◆ chaque algorithme qu'on va voir est associé à un type d'agent spécifique
- Il existe plusieurs types d'environnement
  - ◆ leurs caractéristiques vont déterminer quel algorithme on devrait utiliser

# **Vous devriez être capable de...**

- Donner une définition de l'intelligence artificielle
- Expliquer pourquoi l'approche par intelligence artificielle peut être plus appropriée
- Définir ce qu'est un agent et donnez des exemples
- Faire une analyse d'un agent selon le modèle PEAS
- Déterminer les caractéristiques d'un environnement donné

# Prochain cours

- Recherche heuristique globale