



Intelligence Artificielle

Une approche cognitive

Travail d'Étude et de Recherche

Master Informatique 1^{ère} année (GMIN20B)

William Dyce Thibaut Marmin
Namrata Patel Clément Sipieter

Université Montpellier 2
Encadrés par Violaine Prince et Guillaume Tisserant

○
○○
○
○○○
○○

○○○○○
○○○○○
○○○○○
○○○○○○
○

○○○
○○



Intelligence Artificielle : une approche cognitive

Introduction

Outils de travail

Analyse Générale

Conclusion & Perspectives

Analyse & Implémentation





Intelligence Artificielle : une approche cognitive

Introduction

Présentation du sujet

Outils de travail

Un modèle de conscience
artificielle

Conclusion & Perspectives

Analyse Générale

Analyse & Implémentation





Présentation du sujet

- Intelligence artificielle





Présentation du sujet

- Intelligence artificielle
- Approche cognitive





Présentation du sujet

- Intelligence artificielle
- Approche cognitive
- Rêve de l'IA



Un modèle de conscience artificielle

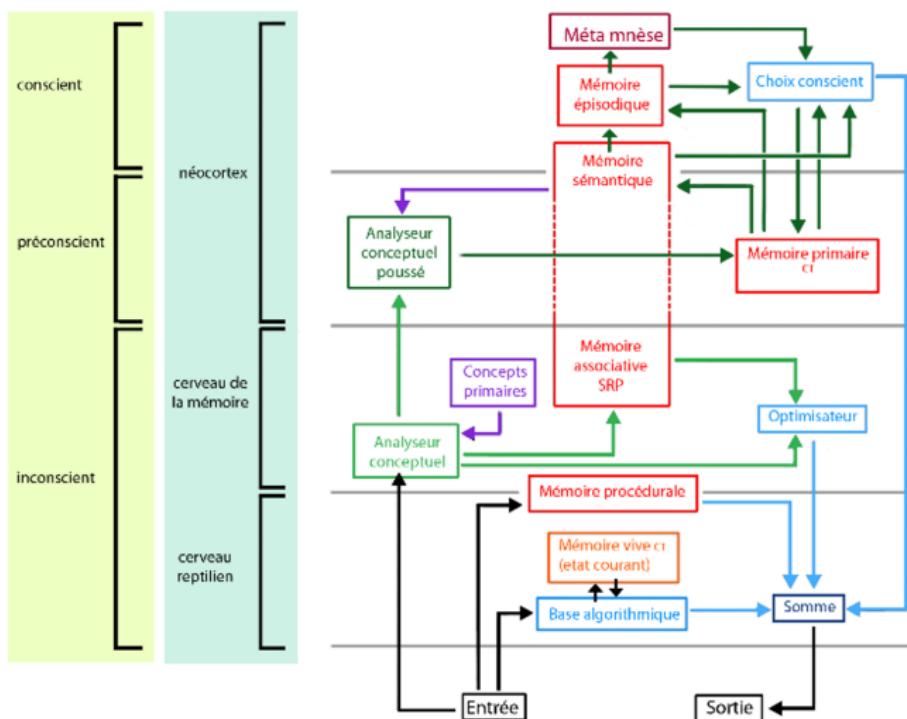
Origine du modèle

- « Conscience artificielle » par Guillaume Tisserant, Guillaume Maurin, Ndongo Wade et Anthony Willemot (Projet du module « Cognition » en M2, 2010)
- Modèle de représentation de la Conscience proposé dans le Chapitre 4



Un modèle de conscience artificielle

Version simplifiée du modèle





Intelligence Artificielle : une approche cognitive

Introduction

Outils de travail

Analyse Générale

Contraintes de réalisation

Conclusion & Perspectives

Restrictions appliquées au
modèle

Domaine d'application

Vision globale du modèle
opérationnel



Analyse & Implémentation

Contraintes de réalisation

Temporelle , d'effectifs et de compétences

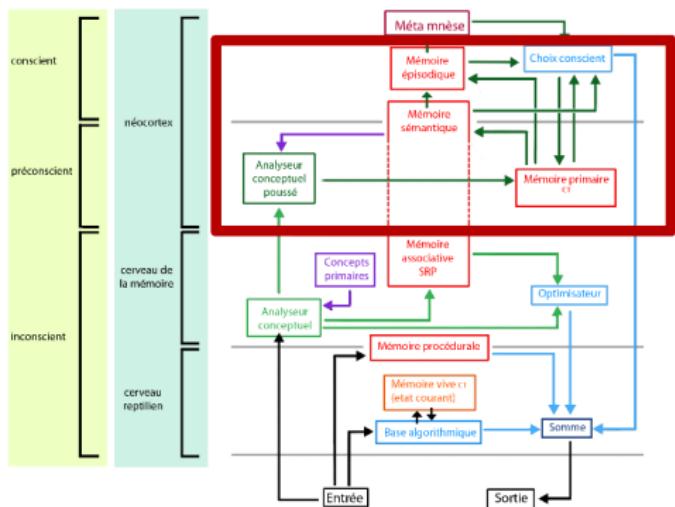
- Temps : travail à réaliser en trois mois
- Effectif : quatre membres dans l'équipe
- Compétences : compétences acquises **durant** le semestre nécessaires pour la réalisation du projet





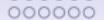
Restrictions appliquées au modèle

Environnement , fonctionnement



- Environnement :** Limitation à un type précis de jeu
- Fonctionnement :** Retrait et simulation des parties
 - Simulation de la partie inconsciente
 - Retrait de la métamnèse





Domaine d'application

Jeu de plateau

Pourquoi le jeu de plateau ?

- Convergence DECOL / IMAGINA



Domaine d'application

Jeu de plateau

Pourquoi le jeu de plateau ?

- Convergence DECOL / IMAGINA
- Activité purement cognitive



Domaine d'application

Jeu de plateau

Pourquoi le jeu de plateau ?

- Convergence DECOL / IMAGINA
- Activité purement cognitive
- Activité cognitive complète



Domaine d'application

Jeu de plateau

Pourquoi le jeu de plateau ?

- Convergence DECOL / IMAGINA
- Activité purement cognitive
- Activité cognitive complète
- Évaluation facile de la performance



Domaine d'application

Théorie des jeux

Théorème du MiniMax

- J. Von Neumann, 1928
- *Stratégie optimale pour un joueur donné*



Domaine d'application

Limites du MiniMax

Types de confrontation

- jeux compétitifs,
- à deux joueurs,
- à somme nulle,
- durée, nombre d'options finis.





Domaine d'application

Limites du MiniMax

Types de confrontation

- jeux compétitifs,
- à deux joueurs,
- à somme nulle,
- durée, nombre d'options finis.

Temps de calcul

- En moyenne $O(b^{\frac{d}{2}})$:
 - d : longueur de la partie
 - b : nombre options par tour
- En pratique : besoin d'heuristiques \Rightarrow perte d'optimalité



Vision globale du modèle opérationnel

Une approche cognitive

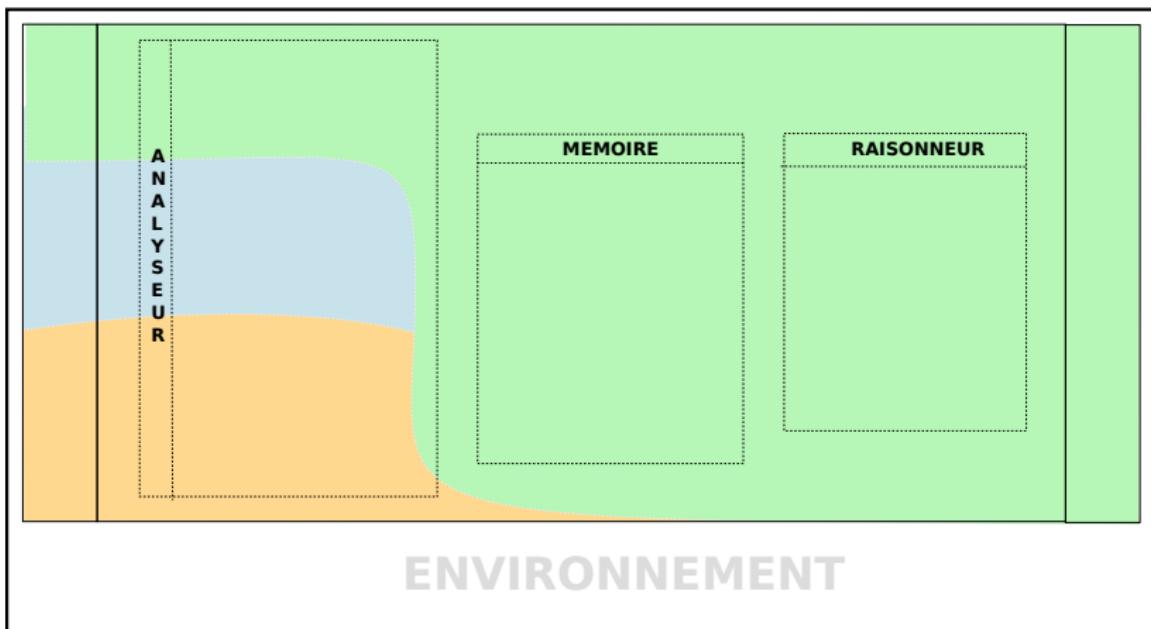


- Apprentissage
- Reconnaissance de formes
- Classification



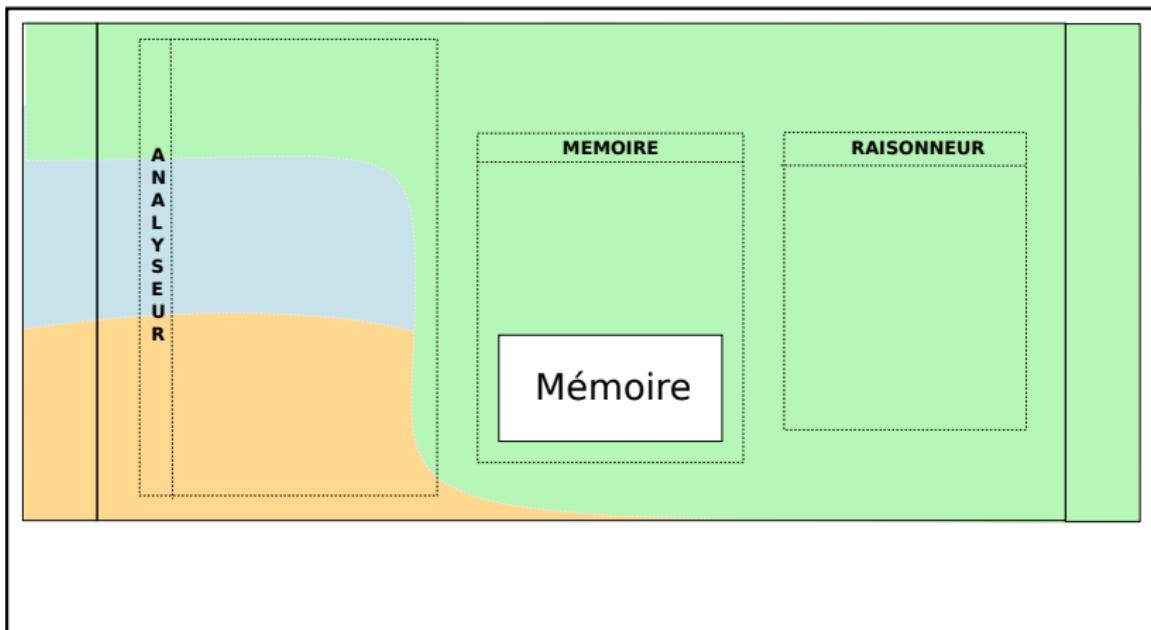
Vision globale du modèle opérationnel

Séquence



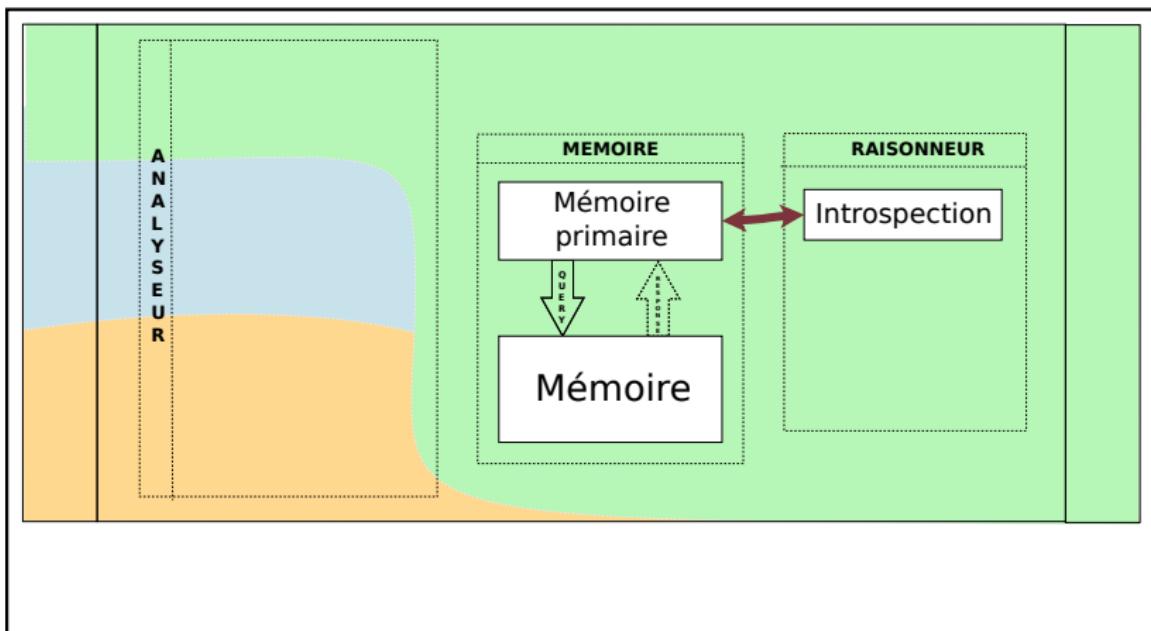
Vision globale du modèle opérationnel

Séquence



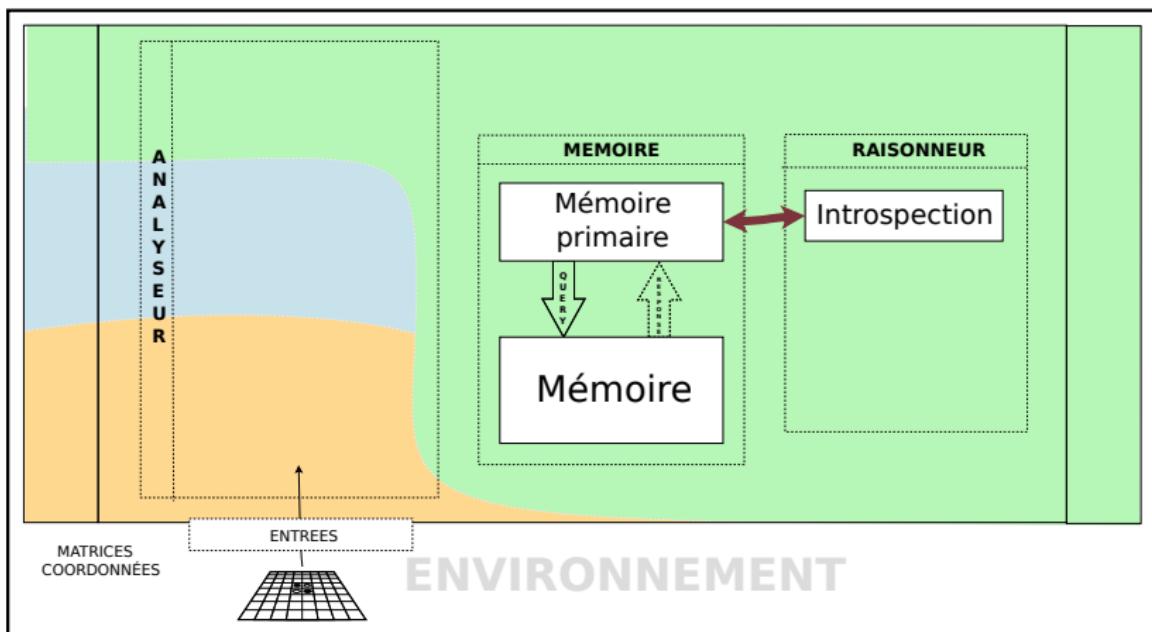
Vision globale du modèle opérationnel

Séquence



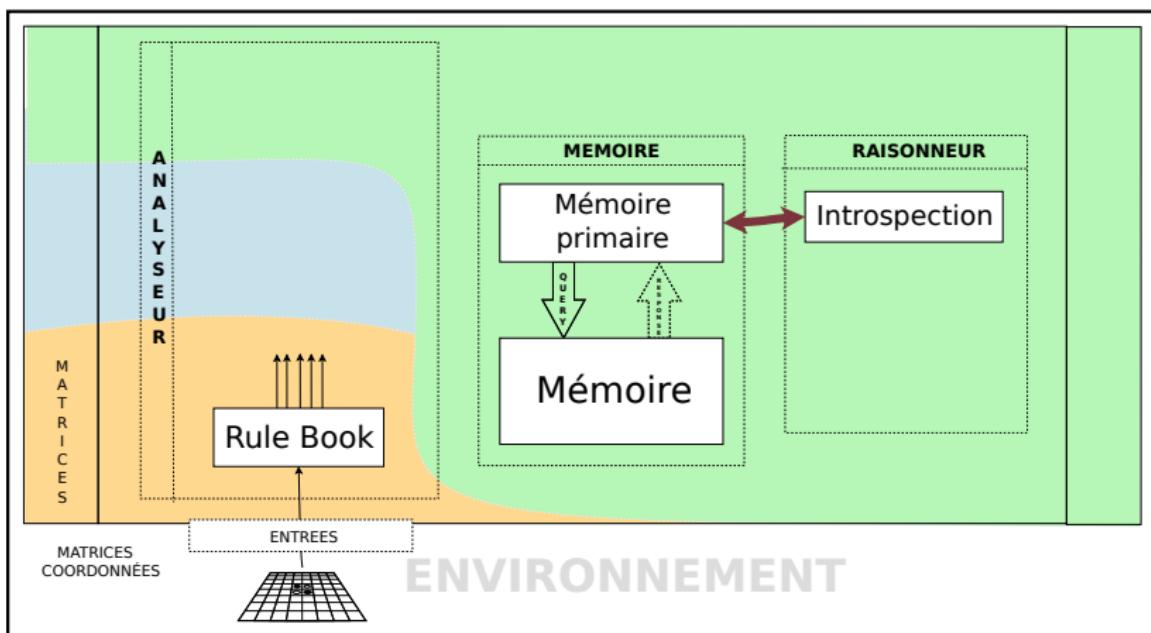
Vision globale du modèle opérationnel

Séquence



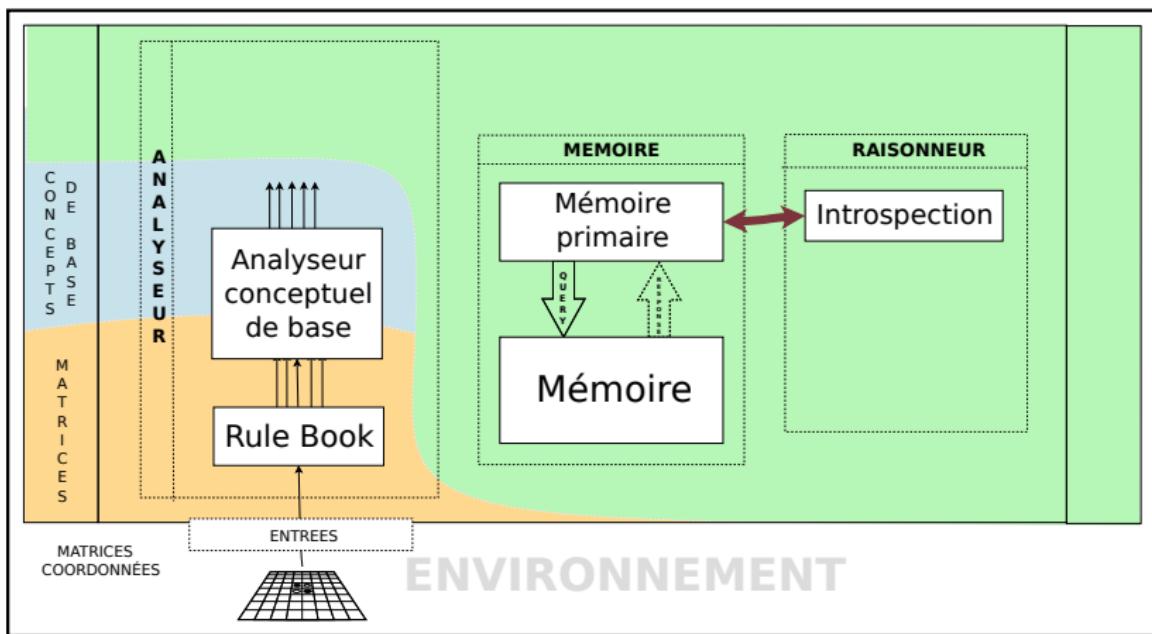
Vision globale du modèle opérationnel

Séquence



Vision globale du modèle opérationnel

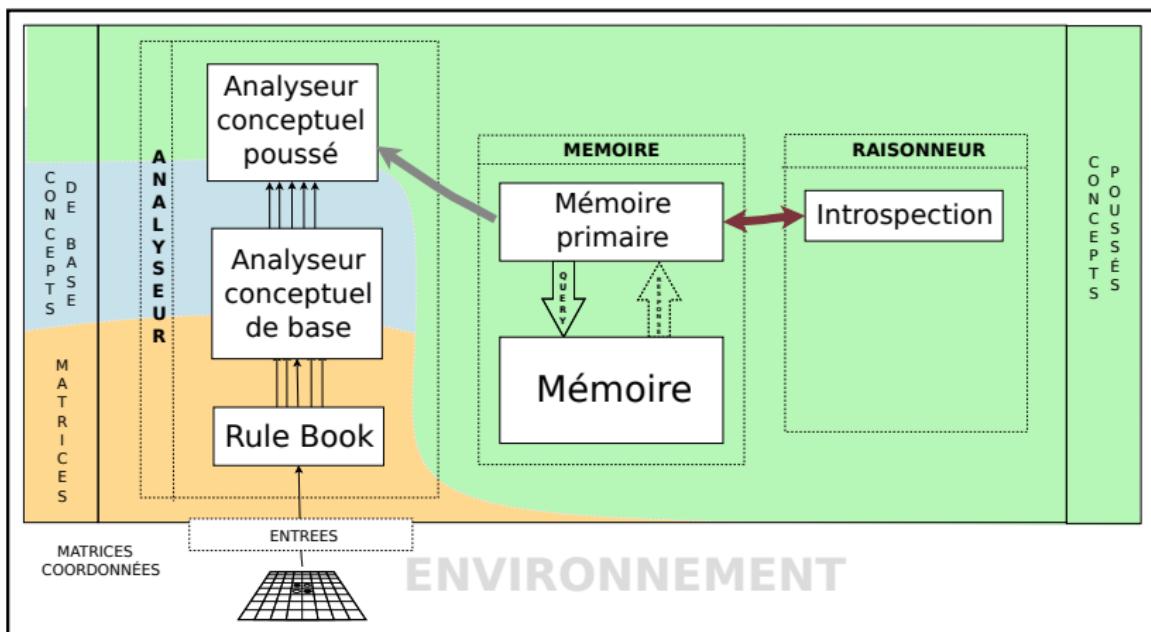
Séquence





Vision globale du modèle opérationnel

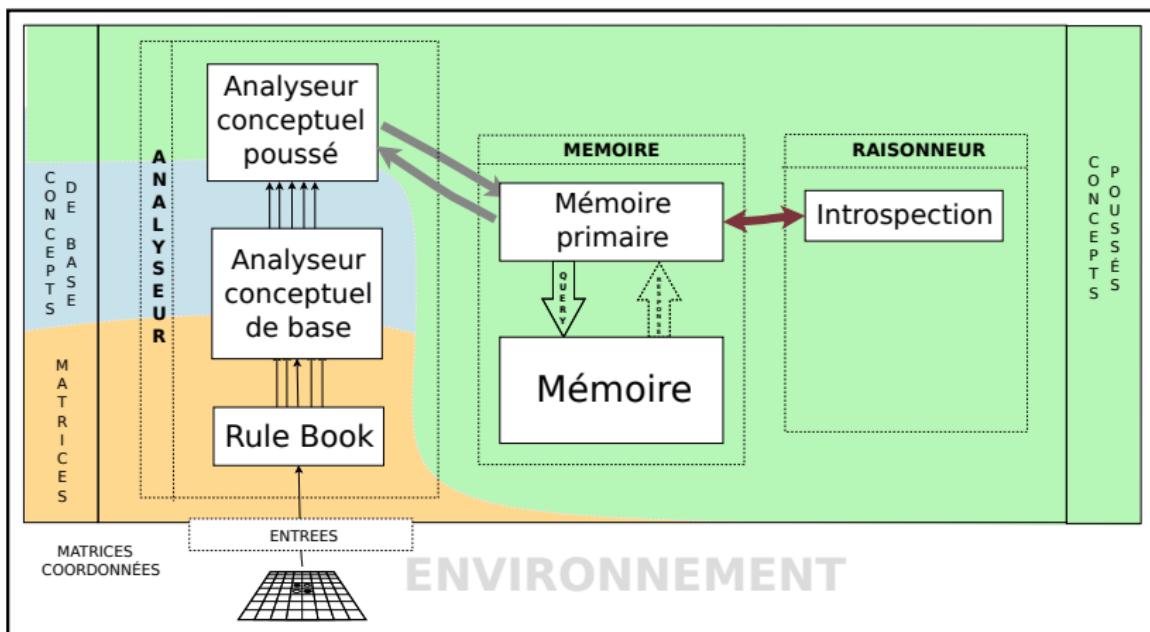
Séquence





Vision globale du modèle opérationnel

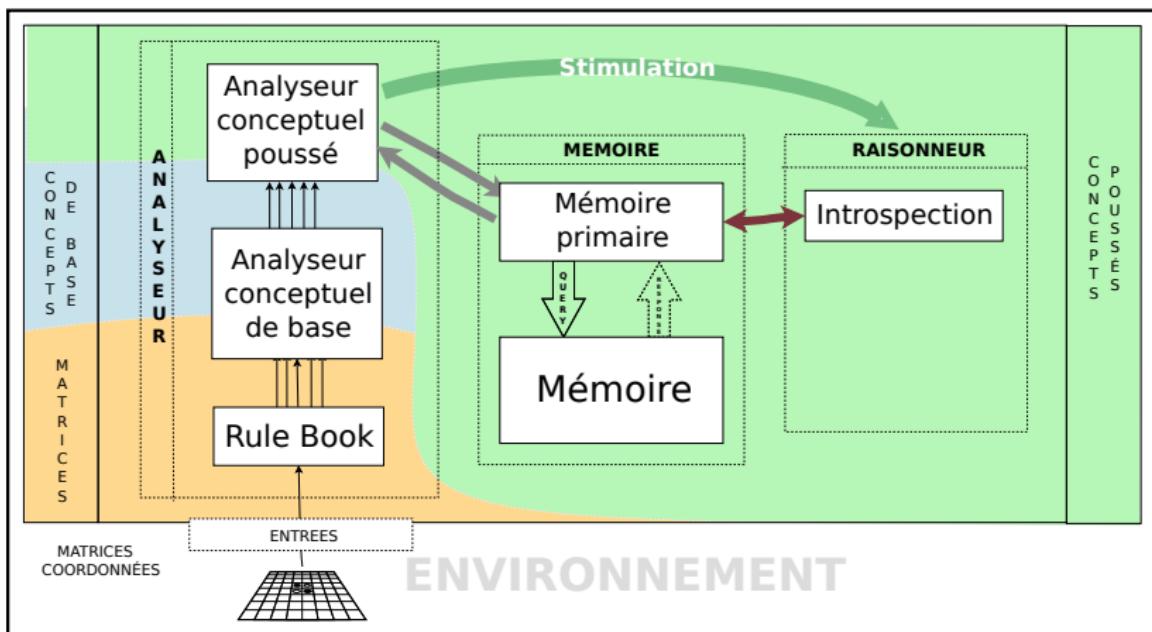
Séquence





Vision globale du modèle opérationnel

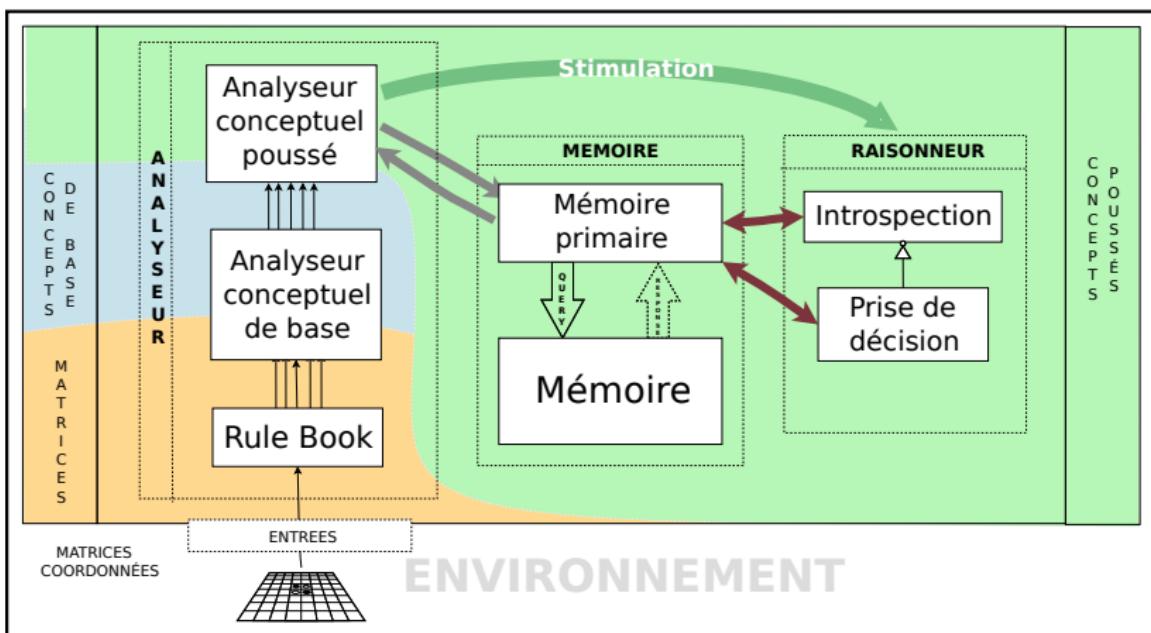
Séquence





Vision globale du modèle opérationnel

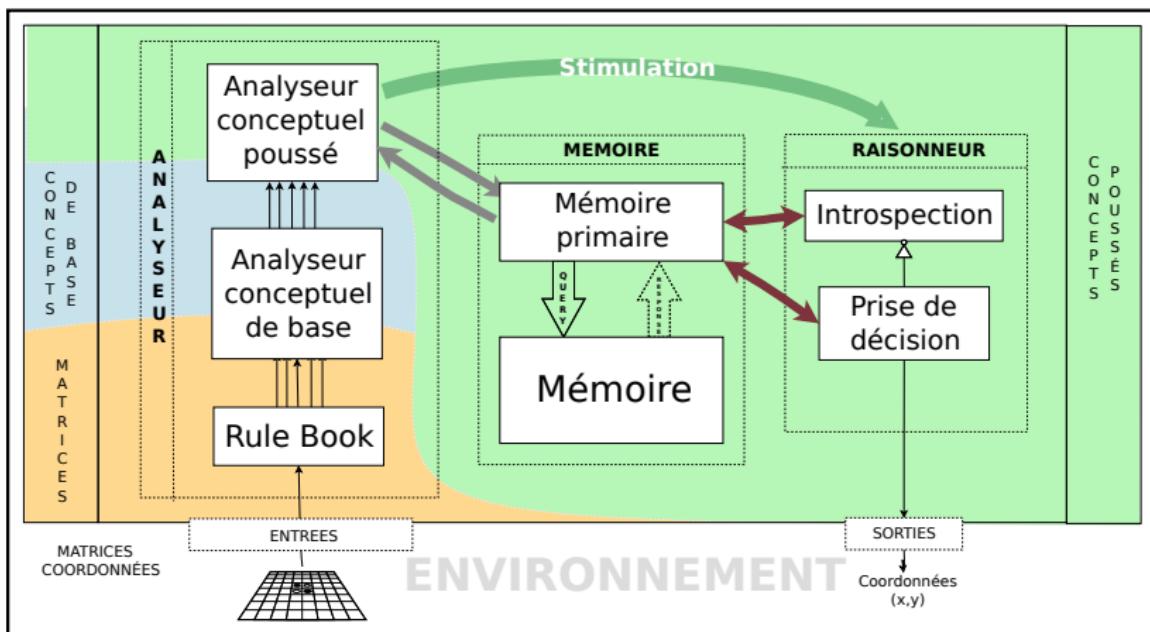
Séquence





Vision globale du modèle opérationnel

Séquence





Intelligence Artificielle : une approche cognitive

Introduction

Outils de travail

Analyse Générale

Conclusion & Perspectives

Analyse & Implémentation

Environnement

Analyseur conceptuel

Raisonneur

Mémoire

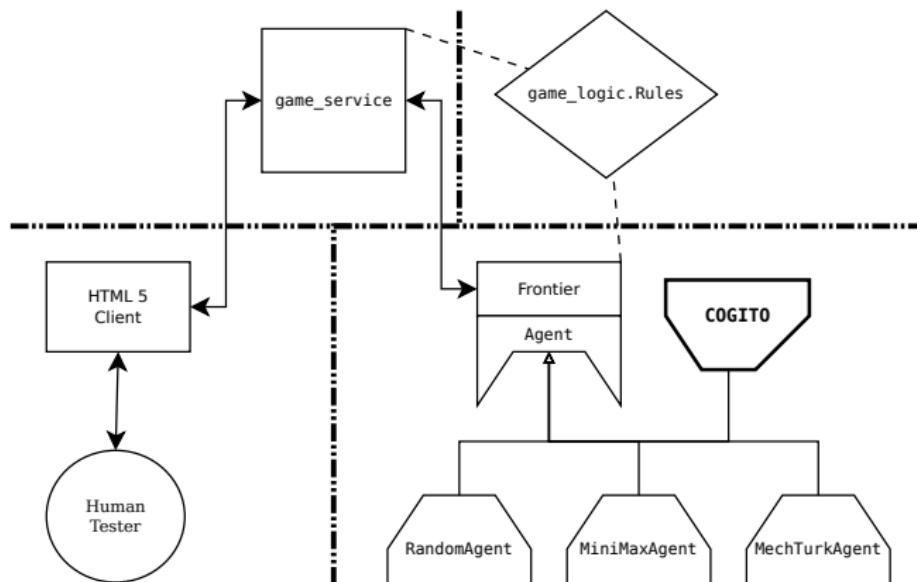
Conclusion partielle





Environnement

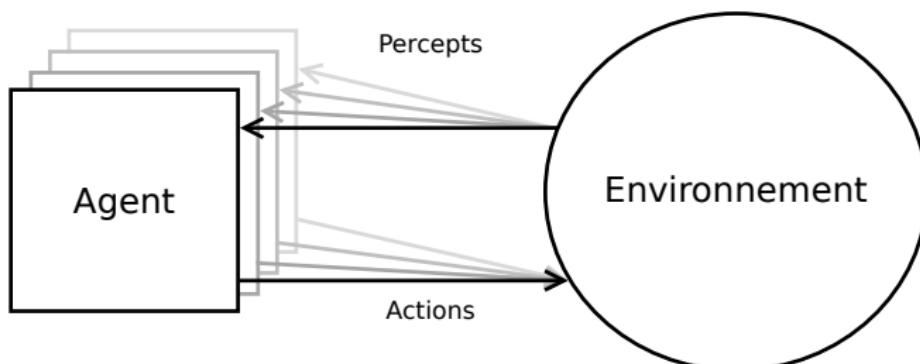
Architecture Globale



Environnement

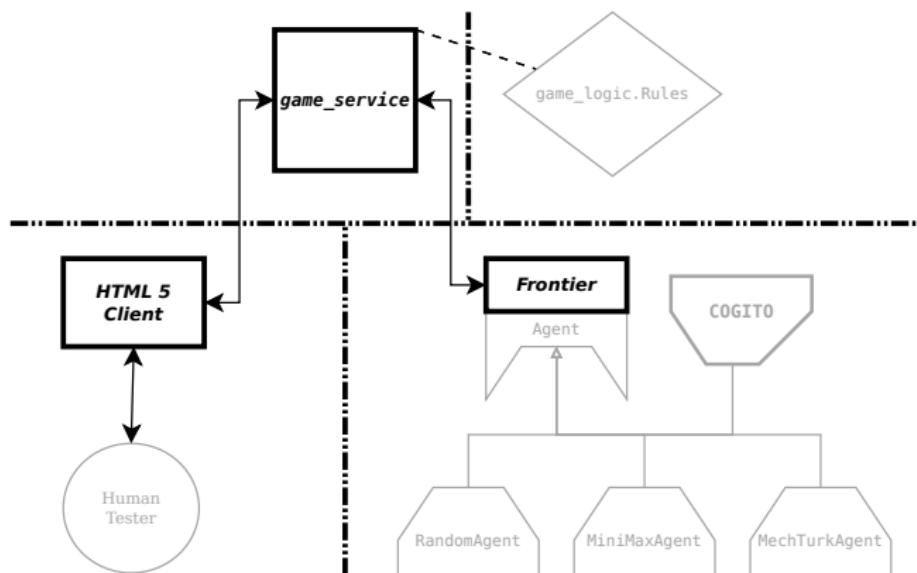
Découplage

- Modèle « Black Board »
- Communication « Stigmergique »



Environnement

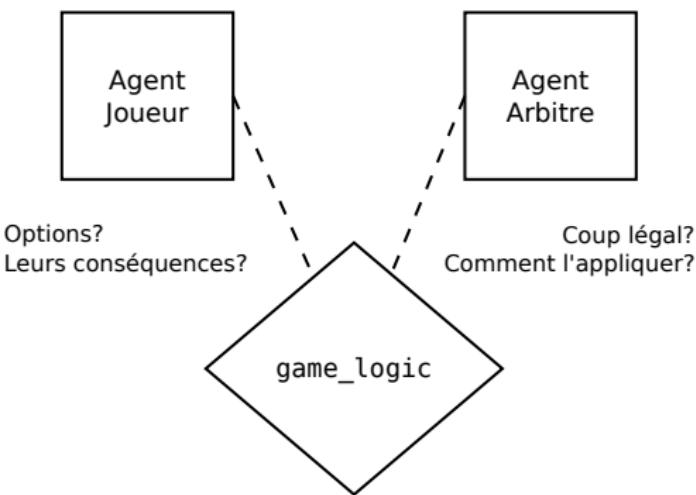
Architecture client-serveur



Environnement

Redondance

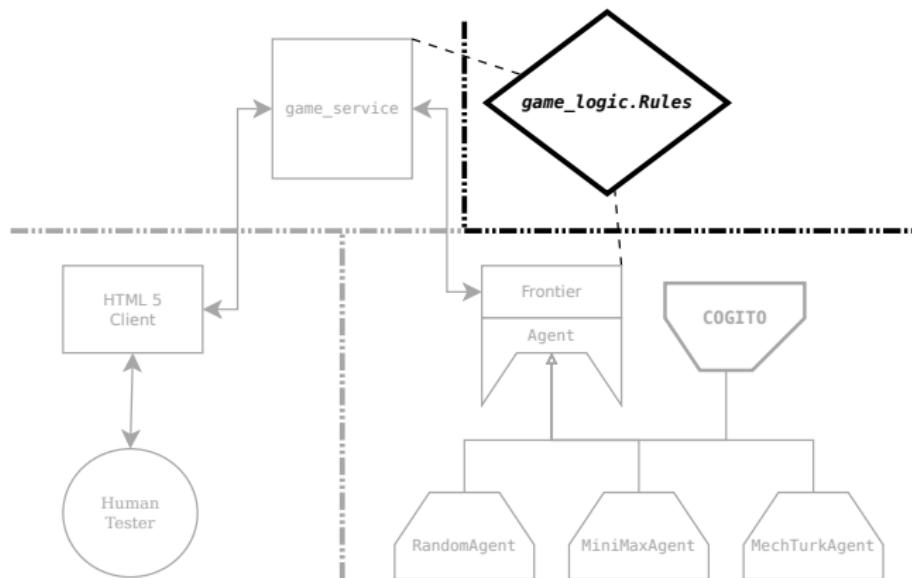
Besoin d'une ressource commune :





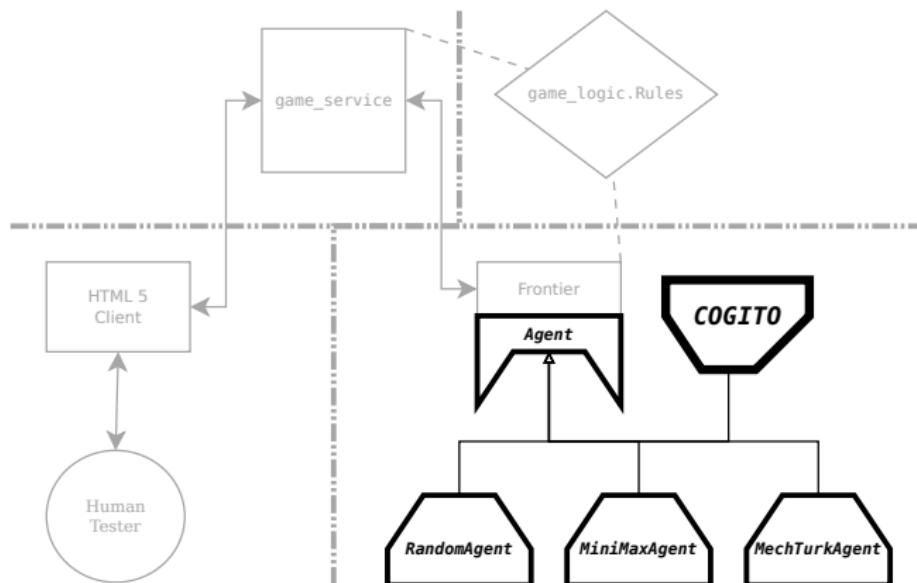
Environnement

Bibliothèque game_logic



Environnement

Agents



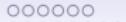
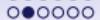


Analyseur conceptuel

Généralités

- Représente les connaissances tirées de l'environnement
- Analyse des connaissances afin d'en extraire de nouvelles



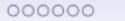
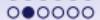


Analyseur conceptuel

Analyse détaillée

- Représentation des connaissances : **vocabulaire**
 - Formules logiques (logique du premier ordre)





Analyseur conceptuel

Analyse détaillée

- Représentation des connaissances : **vocabulaire**
 - Formules logiques (logique du premier ordre)
- Analyse des connaissances : **mécanisme**
 - Recherche d'homomorphismes





Analyseur conceptuel

Implémentation : rôles du module

- **Convertisseur :**

- Rend les données de l'environnement « lisibles » par l'IA

- **Moteur d'inférence :**

- Applique les règles générées par l'IA afin d'en extraire de nouvelles informations





Analyseur conceptuel

Implémentation : Classes principales

- **Choices (environnement)** : représente un plateau courant et l'ensemble des plateaux résultants des coups possibles





Analyseur conceptuel

Implémentation : Classes principales

- **Choices (environnement)** : représente un plateau courant et l'ensemble des plateaux résultants des coups possibles
- **BoardMatrix (environnement)** : représente un plateau sous forme matricielle



Analyseur conceptuel

Implémentation : Classes principales

- **Choices (environnement)** : représente un plateau courant et l'ensemble des plateaux résultants des coups possibles
- **BoardMatrix (environnement)** : représente un plateau sous forme matricielle
- **Choices_FOL (IA)** : version logique du premier ordre de Choices (même structure, attributs décrits par des formules logiques)



Analyseur conceptuel

Implémentation : Classes principales

- **Choices (environnement)** : représente un plateau courant et l'ensemble des plateaux résultants des coups possibles
- **BoardMatrix (environnement)** : représente un plateau sous forme matricielle
- **Choices_FOL (IA)** : version logique du premier ordre de Choices (même structure, attributs décrits par des formules logiques)
- **CompleteBoardState (IA)** : version logique du premier ordre de BoardMatrix (classe qui décrit la configuration d'un plateau complet comme une liste de faits logiques)



Analyseur conceptuel

Implémentation : Classes principales

- **Choices (environnement)** : représente un plateau courant et l'ensemble des plateaux résultants des coups possibles
- **BoardMatrix (environnement)** : représente un plateau sous forme matricielle
- **Choices_FOL (IA)** : version logique du premier ordre de Choices (même structure, attributs décrits par des formules logiques)
- **CompleteBoardState (IA)** : version logique du premier ordre de BoardMatrix (classe qui décrit la configuration d'un plateau complet comme une liste de faits logiques)
- **RelevantPartialBoardState (IA)** : classe qui décrit la configuration d'une sous-partie pertinente d'un plateau comme une règle logique





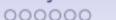
Analyseur conceptuel

Implémentation détaillée

- **Convertisseur :**

- Entrée (de l'environnement) : instance de Choices





Analyseur conceptuel

Implémentation détaillée

- **Convertisseur :**

- Entrée (de l'environnement) : instance de Choices
- Génération de faits logiques représentant la configuration d'un plateau en forme matricielle (BoardMatrix)





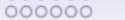
Analyseur conceptuel

Implémentation détaillée

- **Convertisseur :**

- Entrée (de l'environnement) : instance de Choices
- Génération de faits logiques représentant la configuration d'un plateau en forme matricielle (BoardMatrix)
- Base de faits correspondante stockée dans une instance de CompleteBoardState





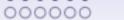
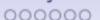
Analyseur conceptuel

Implémentation détaillée

- **Convertisseur :**

- Entrée (de l'environnement) : instance de Choices
- Génération de faits logiques représentant la configuration d'un plateau en forme matricielle (BoardMatrix)
- Base de faits correspondante stockée dans une instance de CompleteBoardState
- Sortie : instance de Choices_FOL





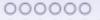
Analyseur conceptuel

Implémentation détaillée

- **Moteur d'inférence :**

- Entrée (de la mémoire) : liste de RelevantPartialBoardState, règles du type :





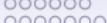
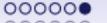
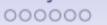
• Moteur d'inférence :

- Entrée (de la mémoire) : liste de RelevantPartialBoardState, règles du type :
 $isCorner(x) \wedge isMine(x) \implies _{rPBS} 034(x)$

Analyseur conceptuel

Implémentation détaillée





Analyseur conceptuel

Implémentation détaillée

- **Moteur d'inférence :**

- Entrée (de la mémoire) : liste de RelevantPartialBoardState, règles du type :
 $isCorner(x) \wedge isMine(x) \implies _rpbs034(x)$
- Saturation de la base de faits des CompleteBoardState en appliquant l'ensemble de ces règles



Analyseur conceptuel

Implémentation détaillée

- **Moteur d'inférence :**

- Entrée (de la mémoire) : liste de RelevantPartialBoardState, règles du type :
 $isCorner(x) \wedge isMine(x) \implies _rpbs034(x)$
- Saturation de la base de faits des CompleteBoardState en appliquant l'ensemble de ces règles
- Reconnaissance de formes : recherche d'homomorphisme entre l'atome de conclusion de la règle (RelevantPartialBoardState) et la base de faits de chaque plateau



Analyseur conceptuel

Implémentation détaillée

- **Moteur d'inférence :**

- Entrée (de la mémoire) : liste de RelevantPartialBoardState, règles du type :
 $isCorner(x) \wedge isMine(x) \implies _rpbs034(x)$
- Saturation de la base de faits des CompleteBoardState en appliquant l'ensemble de ces règles
- Reconnaissance de formes : recherche d'homomorphisme entre l'atome de conclusion de la règle (RelevantPartialBoardState) et la base de faits de chaque plateau
- Ajout de la liste de RelevantPartialBoardState présents dans chaque CompleteBoardState du paquet Choices_FOL



Analyseur conceptuel

Implémentation détaillée

- **Moteur d'inférence :**

- Entrée (de la mémoire) : liste de RelevantPartialBoardState, règles du type :
 $isCorner(x) \wedge isMine(x) \implies _rpbs034(x)$
- Saturation de la base de faits des CompleteBoardState en appliquant l'ensemble de ces règles
- Reconnaissance de formes : recherche d'homomorphisme entre l'atome de conclusion de la règle (RelevantPartialBoardState) et la base de faits de chaque plateau
- Ajout de la liste de RelevantPartialBoardState présents dans chaque CompleteBoardState du paquet Choices_FOL
- Appel à la méthode stimulate du module de raisonnement
- Sortie (passée à la mémoire) : instance de Choices_FOL



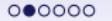
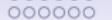


Raisonneur

Généralités

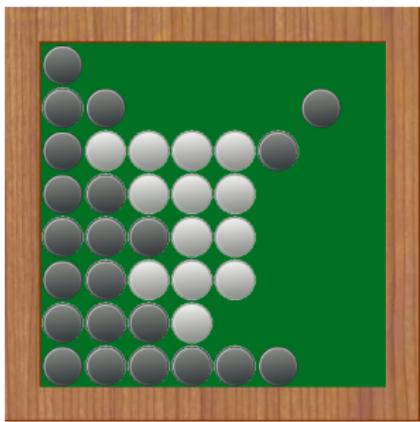
- Prise de décision
- Valuation des formes connues
- Découverte de nouvelles formes





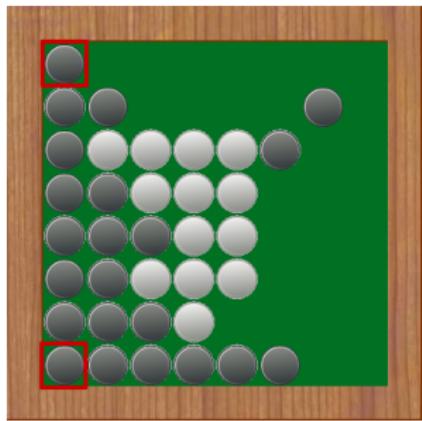
Raisonneur

Moteur de choix



Raisonneur

Moteur de choix



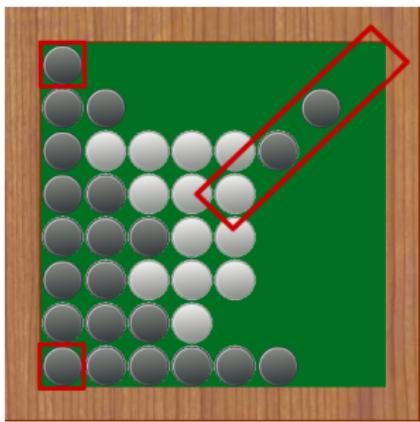
- $isCorner(x) \wedge isMine(x)$





Raisonneur

Moteur de choix



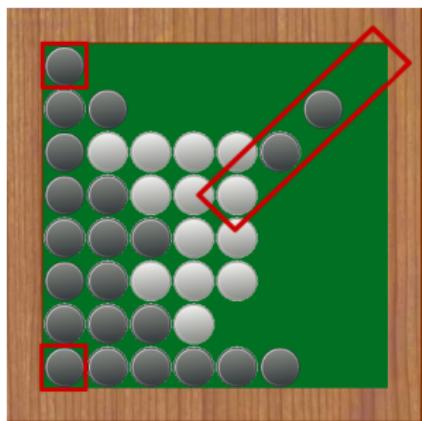
- $isCorner(x) \wedge isMine(x)$
- $isMine(w) \wedge isOpp(x) \wedge isOpp(y) \wedge aligned(w, x, y) \wedge isEmpty(z) \wedge aligned(x, y, z)$





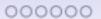
Raisonneur

Moteur de choix



- Moyenne de la probabilité de gain de chaque forme reconnue : $\overline{P(Gain|Forme)}$





Raisonneur

Valuation des formes

- Valuation d'une forme :

$$P(\text{Gain}|\text{Forme}) = \frac{P(\text{Forme}|\text{Gain}) \times P(\text{Gain})}{P(\text{Forme})}$$

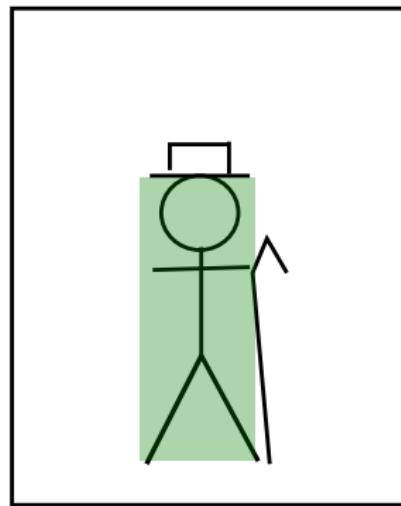
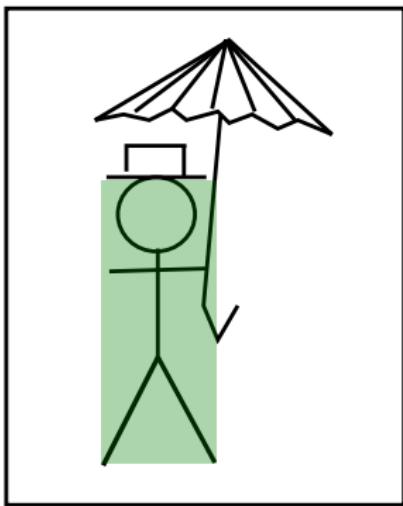
- Fin de partie → mise à jour des formes rencontrées



Raisonneur

Moteur d'introspection

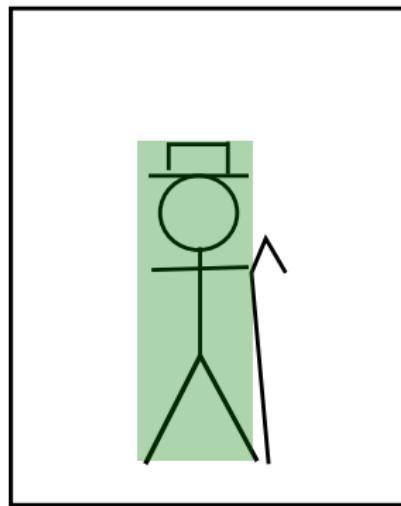
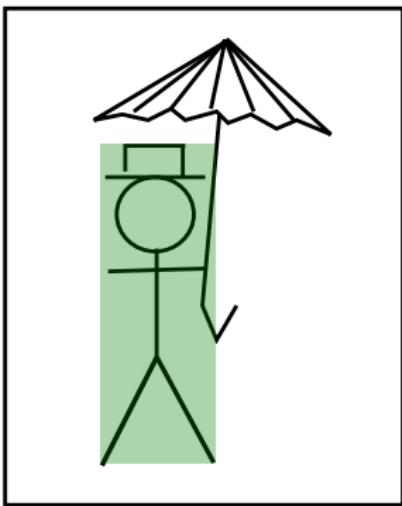
Extraction de nouvelles formes



Raisonneur

Moteur d'introspection

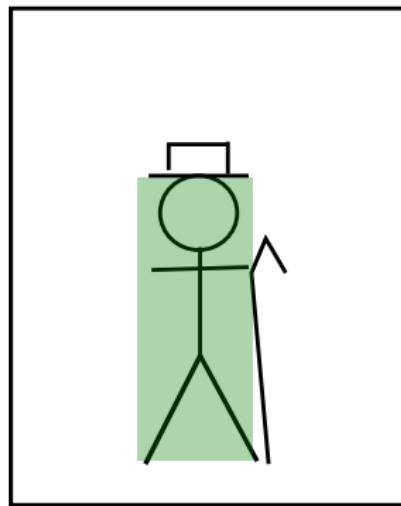
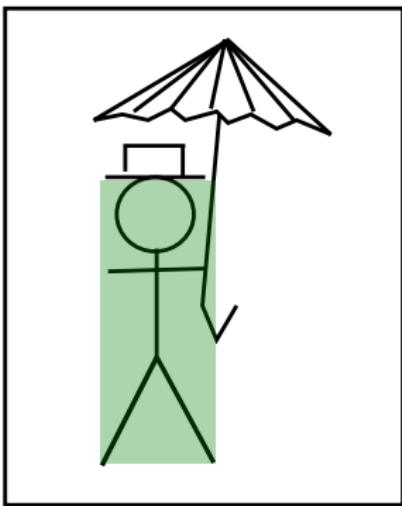
Extraction de nouvelles formes

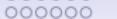


Raisonneur

Moteur d'introspection

Extraction de nouvelles formes

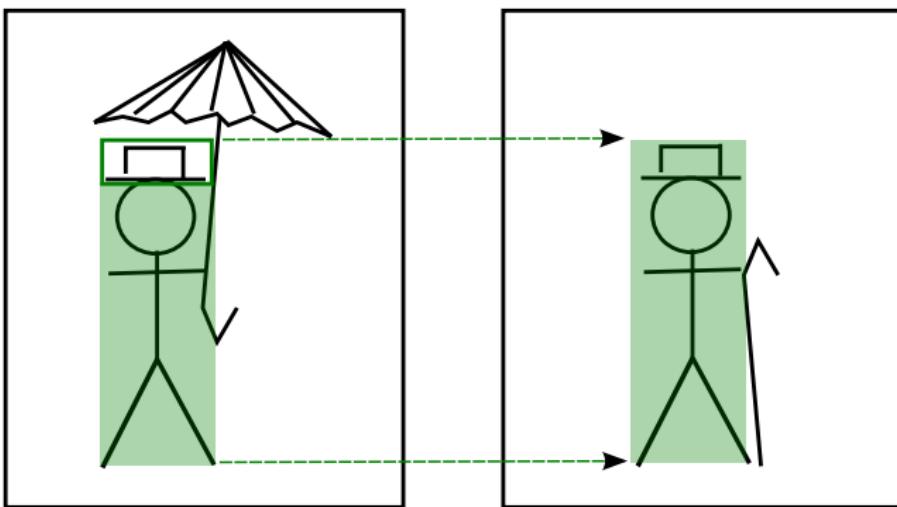




Raisonneur

Moteur d'introspection

Extraction de nouvelles formes



○
○○

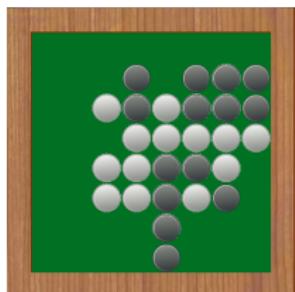
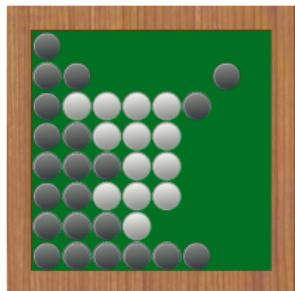
○
○
○○○
○○

○○○○○
○○○○○
○○○●○○○
○

○○○
○○

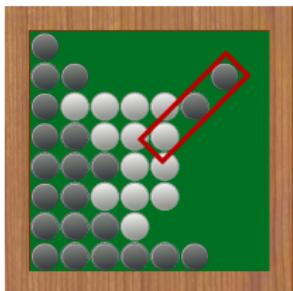
Raisonneur

Moteur d'introspection

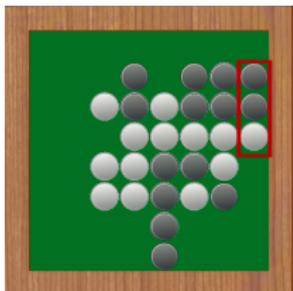


Raisonneur

Moteur d'introspection



$\dots \text{isMine}(x0) \wedge \text{isOpp}(x1) \wedge \text{isOpp}(x2) \wedge \text{aligned}(x0, x1, x2) \dots$

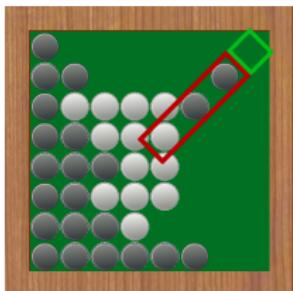


$\dots \text{isMine}(y0) \wedge \text{isOpp}(y1) \wedge \text{isOpp}(y2) \wedge \text{aligned}(y0, y1, y2) \dots$

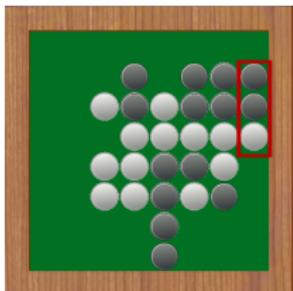


Raisonneur

Moteur d'introspection



$\dots \text{isMine}(x0) \wedge \text{isOpp}(x1) \wedge \text{isOpp}(x2) \wedge$
 $\text{aligned}(x0, x1, x2) \wedge \text{isEmpty}(x3) \wedge$
 $\text{aligned}(x1, x2, x3) \wedge \text{isCorner}(x3) \dots$

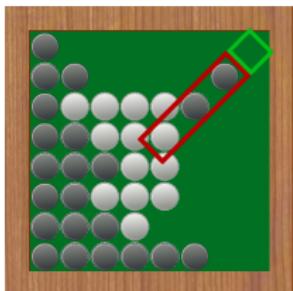


$\dots \text{isMine}(y0) \wedge \text{isOpp}(y1) \wedge \text{isOpp}(y2) \wedge$
 $\text{aligned}(y0, y1, y2) \dots$

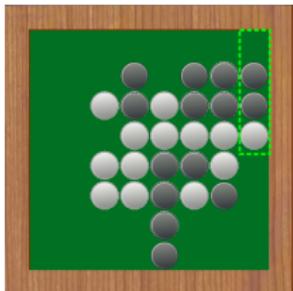


Raisonneur

Moteur d'introspection



$\dots \text{isMine}(x0) \wedge \text{isOpp}(x1) \wedge \text{isOpp}(x2) \wedge$
 $\text{aligned}(x0, x1, x2) \wedge \text{isEmpty}(x3) \wedge$
 $\text{aligned}(x1, x2, x3) \wedge \text{isCorner}(x3) \dots$



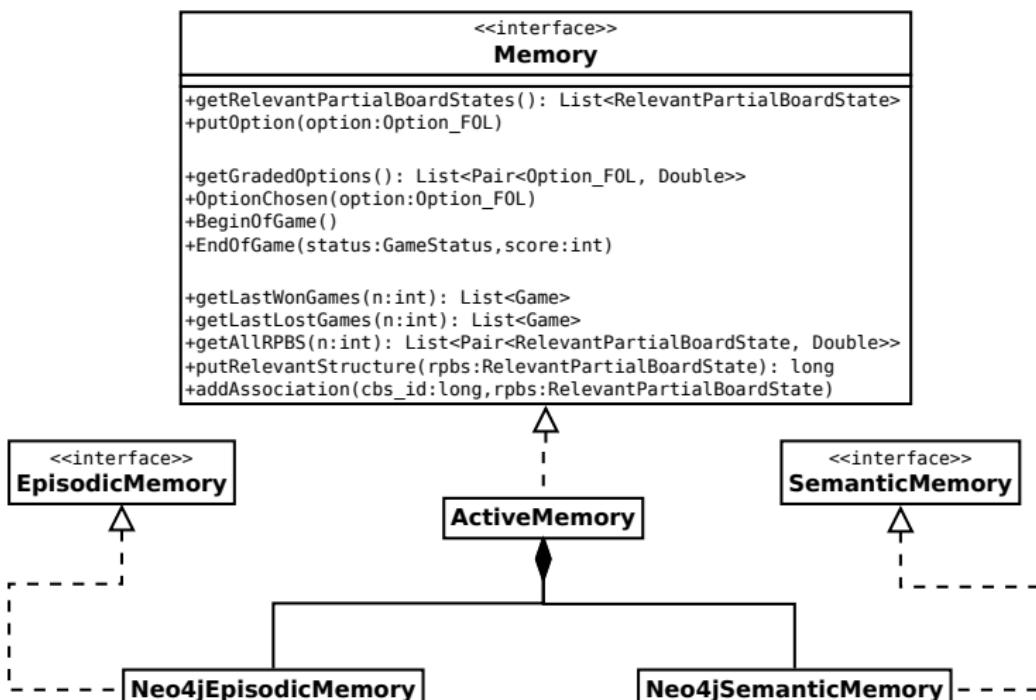
$\dots \text{isMine}(y0) \wedge \text{isOpp}(y1) \wedge \text{isOpp}(y2) \wedge$
 $\text{aligned}(y0, y1, y2) \wedge \text{isEmpty}(y3) \wedge$
 $\text{aligned}(y1, y2, y3) \wedge \text{isCorner}(y3) \dots$





Mémoire

Architecture de la mémoire

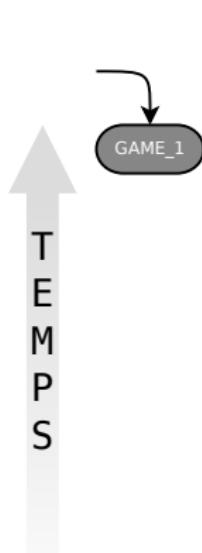




Mémoire

Mémoire épisodique

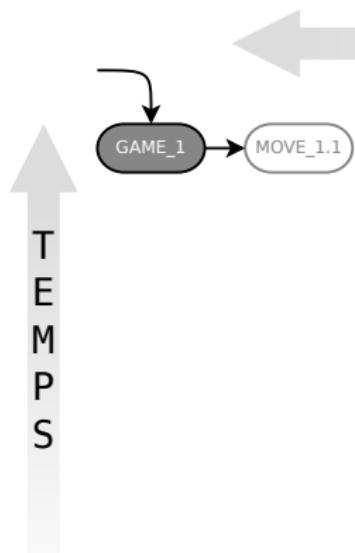
T E M P S



Mémoire

Mémoire épisodique

T E M P S

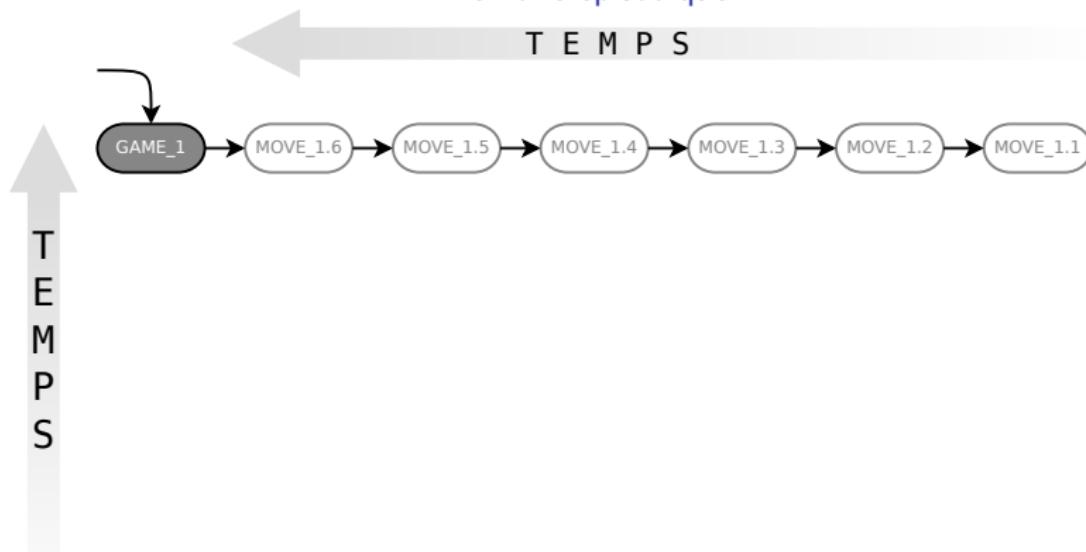




Mémoire

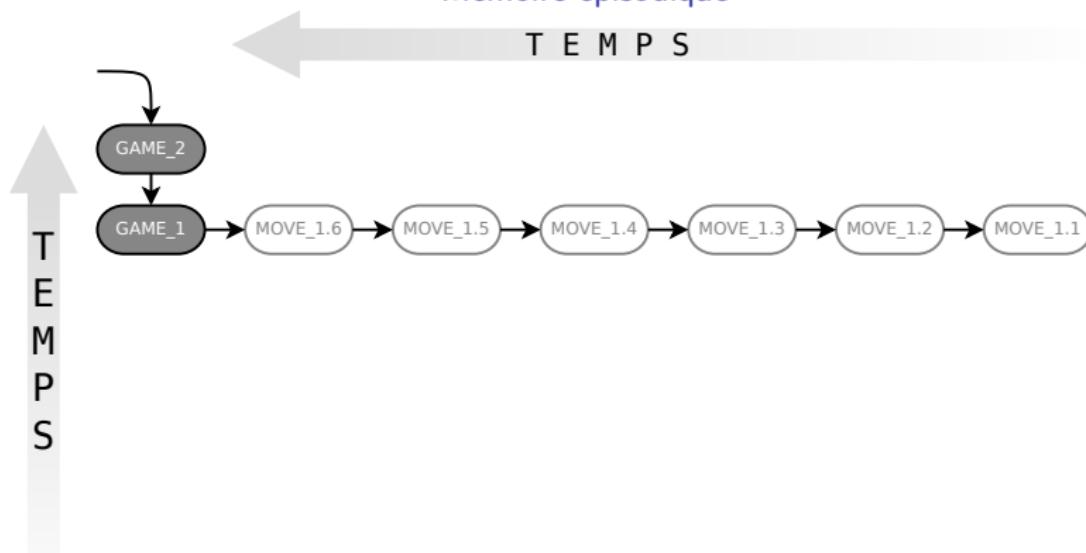
Mémoire épisodique

T E M P S



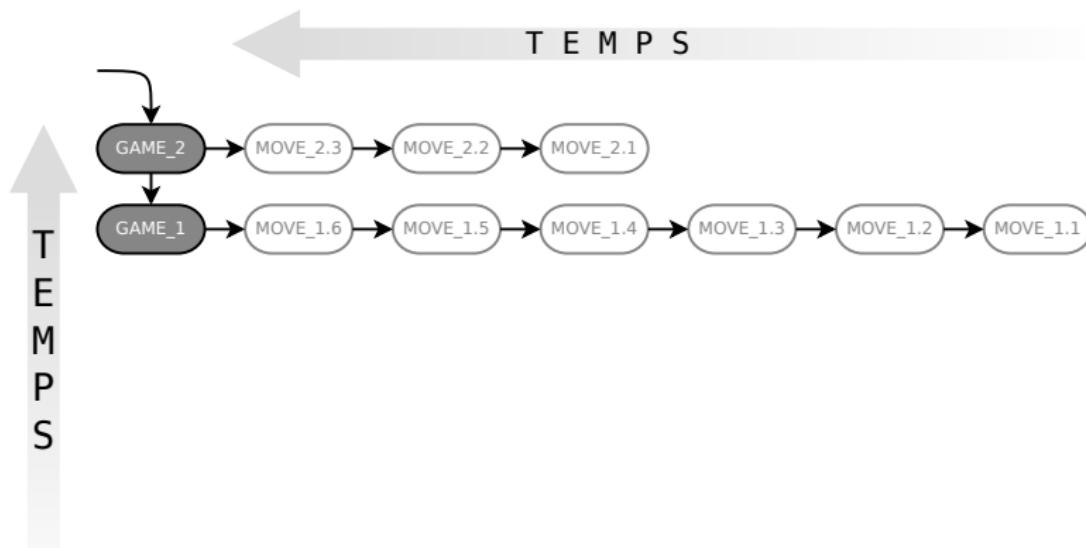
Mémoire

Mémoire épisodique



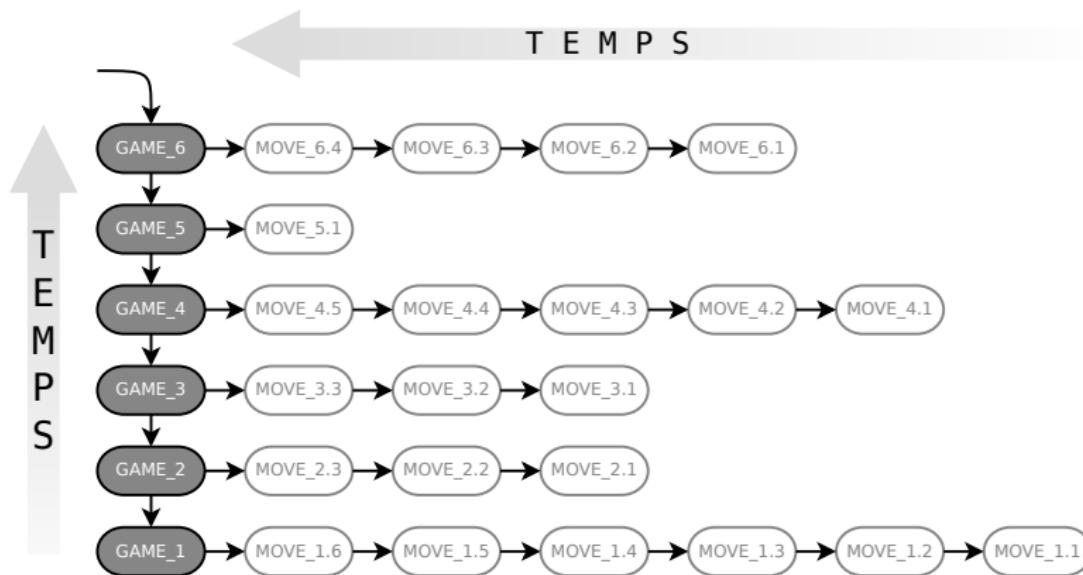
Mémoire

Mémoire épisodique



Mémoire

Mémoire épisodique



Mémoire

Mémoire sémantique

Vision matricielle

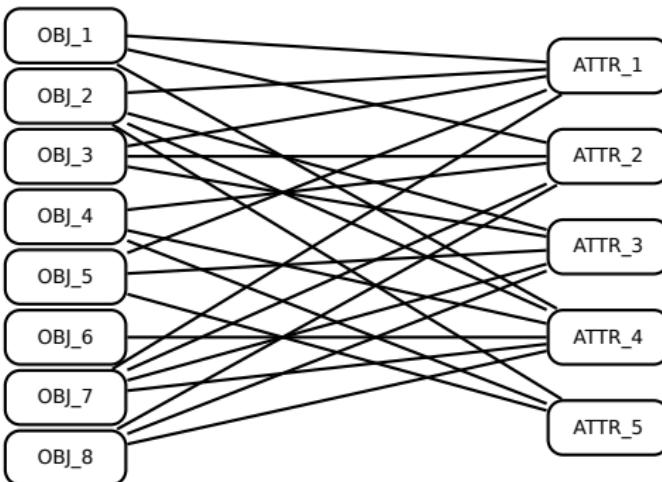
	ATTR_1	ATTR_2	ATTR_3	ATTR_4	ATTR_5
OBJ_1	●	●	●		
OBJ_2	●		●	●	●
OBJ_3	●	●	●		
OBJ_4		●		●	●
OBJ_5	●		●		●
OBJ_6				●	
OBJ_7	●	●	●	●	
OBJ_8		●	●	●	



Mémoire

Mémoire sémantique

Vision en graphe





Mémoire

Persistance



Neo4j

- Logiciel libre
(GPLv3 / AGPLv3)
- SGBD NoSQL orienté
graphe
- Respect des caractéristiques
ACID
- Multiples versions
(embedded in Java)





Mémoire

Persistance



Neo4j

- Logiciel libre (GPLv3 / AGPLv3)
- SGBD NoSQL orienté graphe
- Respect des caractéristiques ACID
- Multiples versions (embedded in Java)

Pourquoi ?

- Volonté de découvrir un nouvel outil
- Proche d'une vision cognitive
- Performances





Mémoire

Persistance

Éléments

- Nœud racine
- Nœuds
- Relations (orientées)
- Types de relations
- Attributs (Nœuds & Relations)

Comment typer les nœuds ?





Mémoire

Persistance

Éléments

- Nœud racine
- Nœuds
- Relations (orientées)
- Types de relations
- Attributs (Nœuds & Relations)

Comment typer les nœuds ?

1. Créer un nœud Maitre
2. Créer une relation typée Racine → Maitre
3. Créer des relations typées Maitre → Noeuds





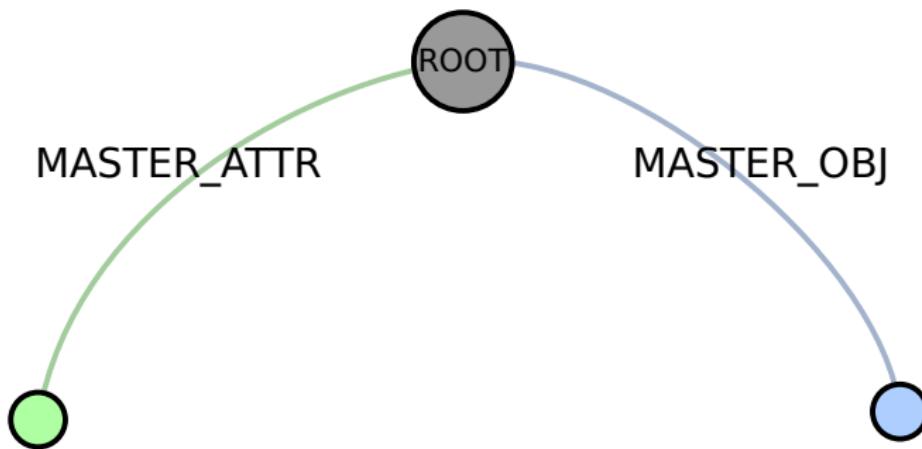
Mémoire

Graphe de la mémoire sémantique



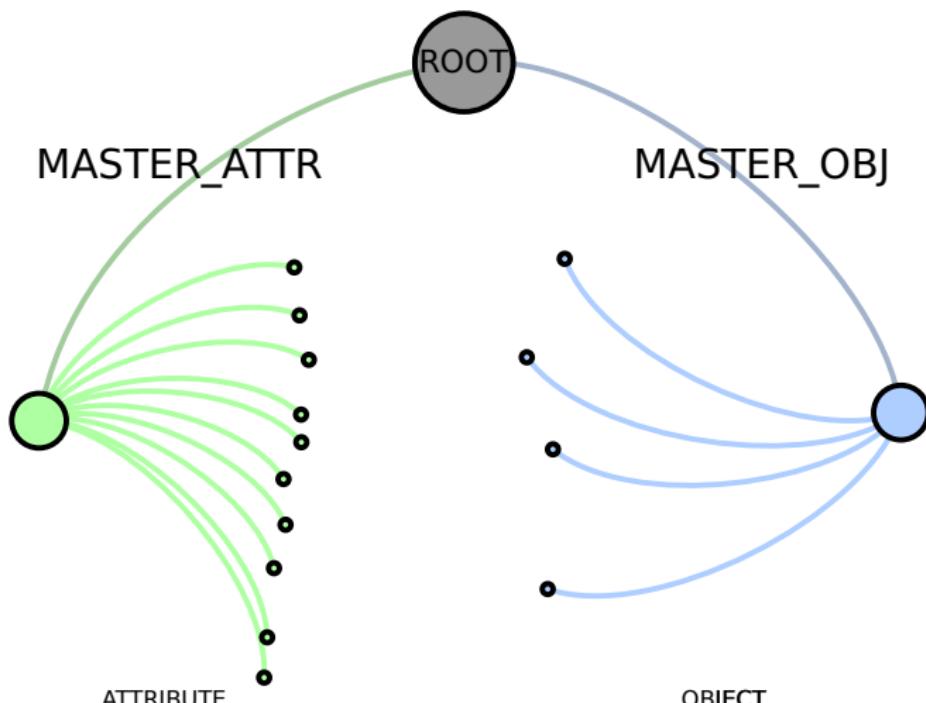
Mémoire

Graphe de la mémoire sémantique



Mémoire

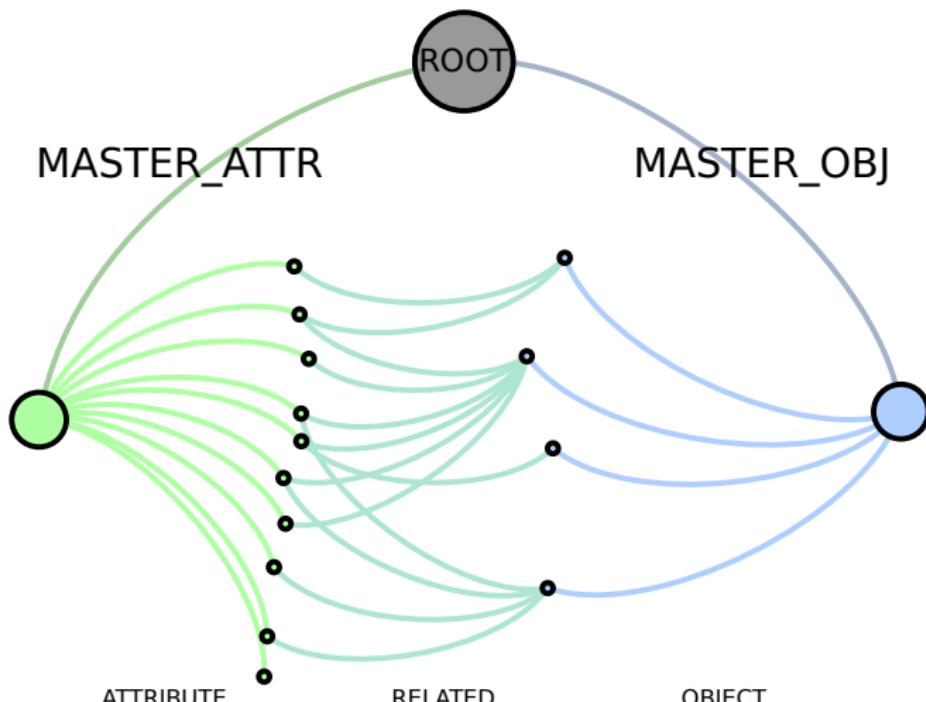
Graphe de la mémoire sémantique

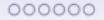
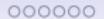




Mémoire

Graphe de la mémoire sémantique





Mémoire

Graphe de la mémoire épisodique





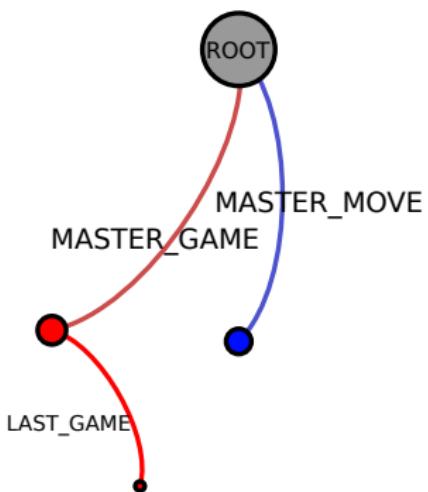
Mémoire

Graphe de la mémoire épisodique



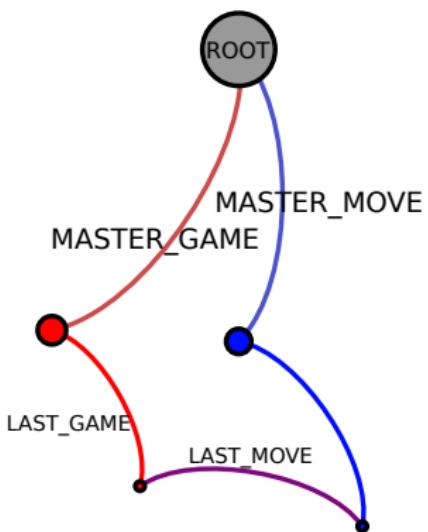
Mémoire

Graphe de la mémoire épisodique



Mémoire

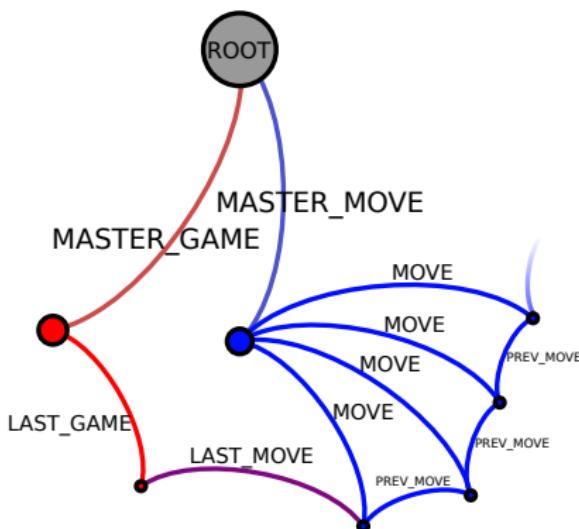
Graphe de la mémoire épisodique





Mémoire

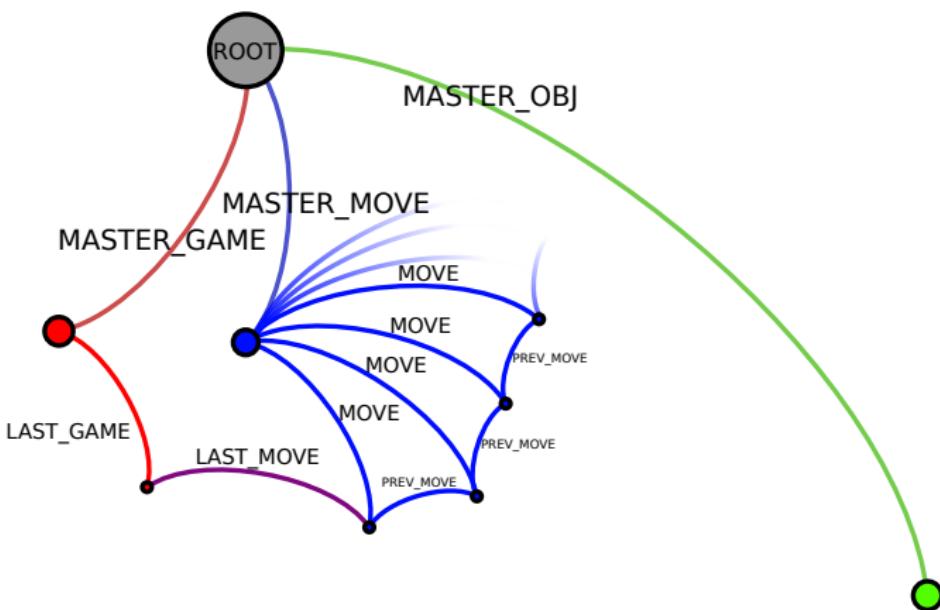
Graphe de la mémoire épisodique





Mémoire

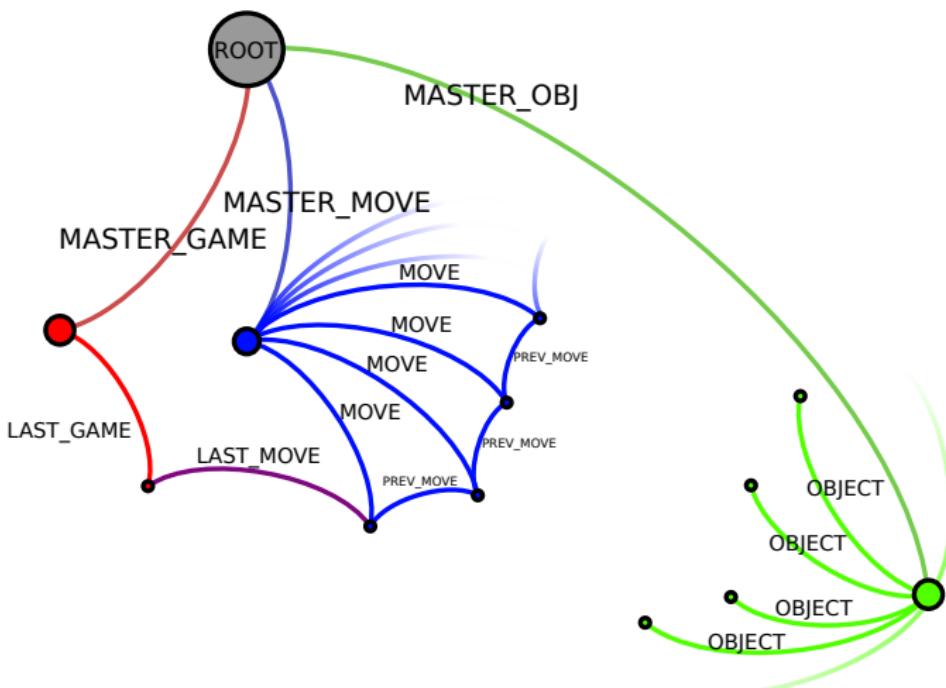
Graphe de la mémoire épisodique





Mémoire

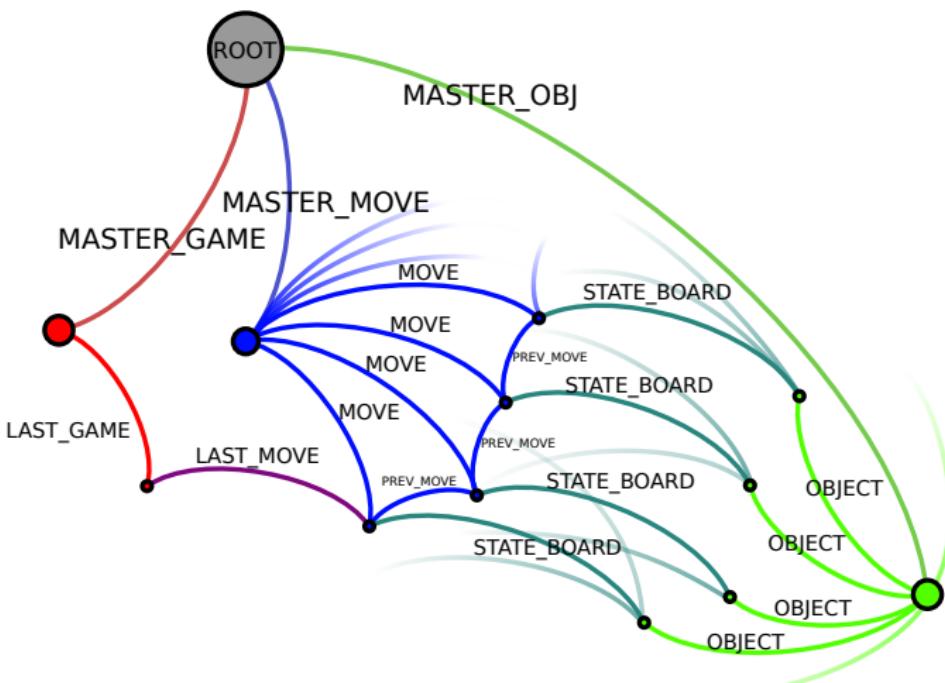
Graphe de la mémoire épisodique





Mémoire

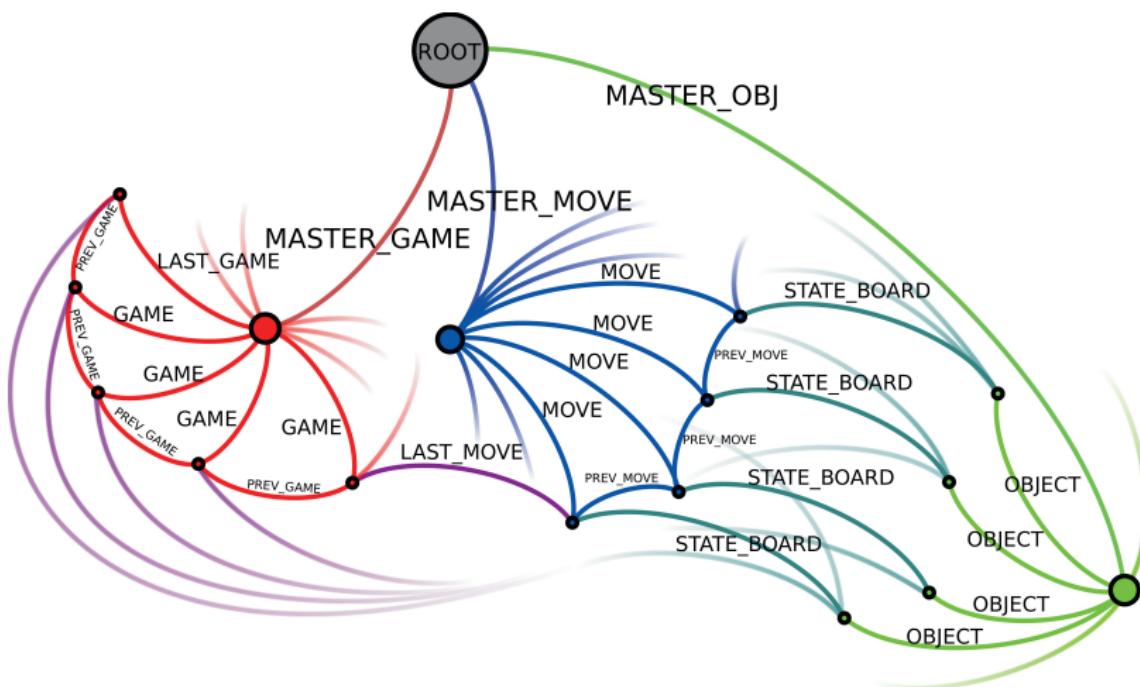
Graphe de la mémoire épisodique





Mémoire

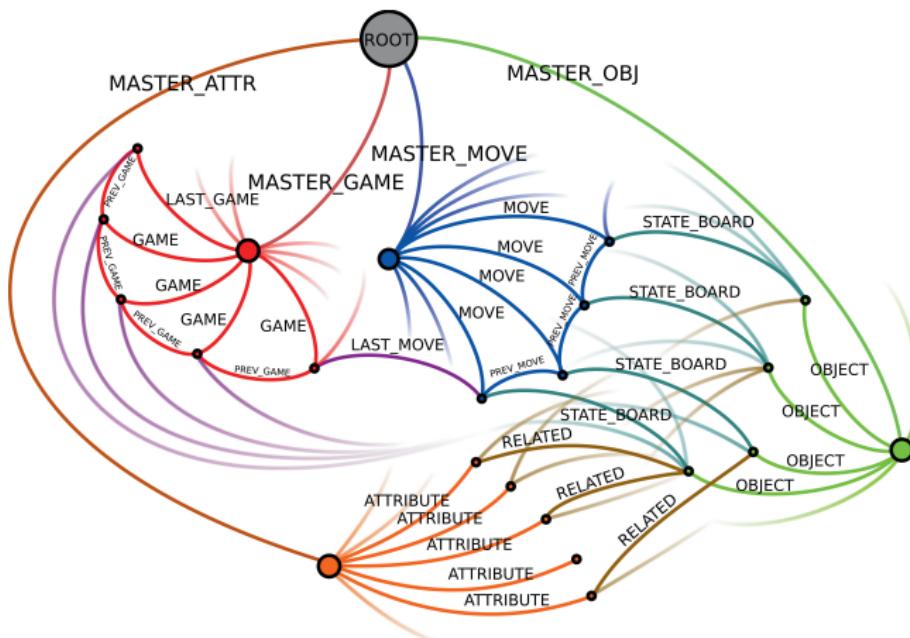
Graphe de la mémoire épisodique





Mémoire

Graphe complet de la mémoire





Analyse & implémentation

Optimisation

- Problème de passage à l'échelle du moteur d'homomorphisme
 - Structure de données inadaptée pour représenter la BF :
`ArrayList`
 - Choix d'une structure plus adaptée : `TreeSet`





Analyse & implémentation

Optimisation

- Problème de passage à l'échelle du moteur d'homomorphisme
 - Structure de données inadaptée pour représenter la BF : `ArrayList`
 - Choix d'une structure plus adaptée : `TreeSet`
- Mise à profit de l'architecture multi-cœurs des processeurs par l'implémentation de *threads*.



o
oo
o
ooo
oo

oooooo
oooooo
oooooo
oooooooo

o



Intelligence Artificielle : une approche cognitive

Introduction

Outils de travail

Analyse Générale

Conclusion & Perspectives

Analyse & Implémentation



Outils de travail

GIT - Un gestionnaire de version décentralisé



- Logiciel libre (GNUv2)
- Simple d'utilisation
- Hébergement via GitHub

https://github.com/cogitoTeam/artificial_consciousness/





Outils de travail

Etherpad - Éditeur de texte collaboratif

EtherPad

EtherPad is the only web-based word processor that allows people to work together in really real-time.

The screenshot shows the Etherpad interface with two users, Thibaut Marmin and Narrator Patel, editing a document titled "Lorem ipsum". The interface includes a toolbar with text styling options, a preview window, and a sidebar for user management. The main area displays a collaborative text document with various paragraphs of Latin placeholder text.

Aperçu d'un pad hébergé sur framapad.org

- Logiciel libre
Licence Apache v2
- Collaboratif en temps réel
- Complet
Chat, couleurs, etc.
- Hébergement via FramaPad
<http://framapad.org/>



Outils de travail

Développement



Logiciel Libre (EPL) / EDI renommé

Javadoc

- Standard industriel
- Réutilisation du code source

Log4j

- Outil libre (Apache v2)
- Journalisation



Pas libre... / Plugin Eclipse &
autres EDI





Intelligence Artificielle : une approche cognitive

Introduction

Outils de travail

Analyse Générale

Conclusion & Perspectives

Analyse & Implémentation

Perspectives
Conclusion



Perspectives

Un treillis de concepts en mémoire ?

Le contexte

Un contexte est un triplet (G, M, I) avec

- G l'ensemble des objets
- M l'ensemble des attributs
- $I \subseteq G \times M$ l'ensemble des relations

On pourrait donc utiliser la mémoire sémantique comme contexte pour la création d'un treillis de concepts.





Perspectives

Un treillis de concepts en mémoire ?

Utile ?

- Abstraction supplémentaire
- Travail sur des ensembles de formes (RPBS)





Perspectives

Un treillis de concepts en mémoire ?

Exemple (concret)



○

○
○
○○
○○○○○○○
○○○○○
○○○○○
○○○○○○○
○○○●
○○

Perspectives

Un treillis de concepts en mémoire ?

Exemple (concret)



Perspectives

Un treillis de concepts en mémoire ?

Exemple (concret)

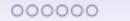
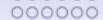
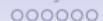


Perspectives

Un treillis de concepts en mémoire ?

Exemple (concret)



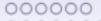


Perspectives

Un treillis de concepts en mémoire ?

Exemple (concret)





Conclusion

Une application fonctionnelle :

- Généralisable à la classification d'objets descriptibles en logique
- Données exploitables



Conclusion

Une expérience enrichissante :

- Mise en pratique de connaissances acquises lors du semestre



Conclusion

Une expérience enrichissante :

- Mise en pratique de connaissances acquises lors du semestre
- Découverte de nouveaux outils



Conclusion

Une expérience enrichissante :

- Mise en pratique de connaissances acquises lors du semestre
- Découverte de nouveaux outils
- Travail collaboratif





Merci pour votre attention

Questions & Démo

https://github.com/cogitoTeam/IA_a_cognitive_approach

