Reasoning with Ontologies for NPCs Artificial Intelligence in Games

Knowledge Representation and Reasoning 2021: Doctoral Consortium

Supervisors: Christophe REY, Bruno BACHELET & Loïc YON

PhD Student: **Sylvain LAPEYRADE**

05 November 2021

Thesis Context

AI decision making lacks credibility

Deus Ex Human Revolution (Eidos-Montréal)

Assassin's Creed: Revelations (Ubisoft)

Fallout 4 (Bethesda)

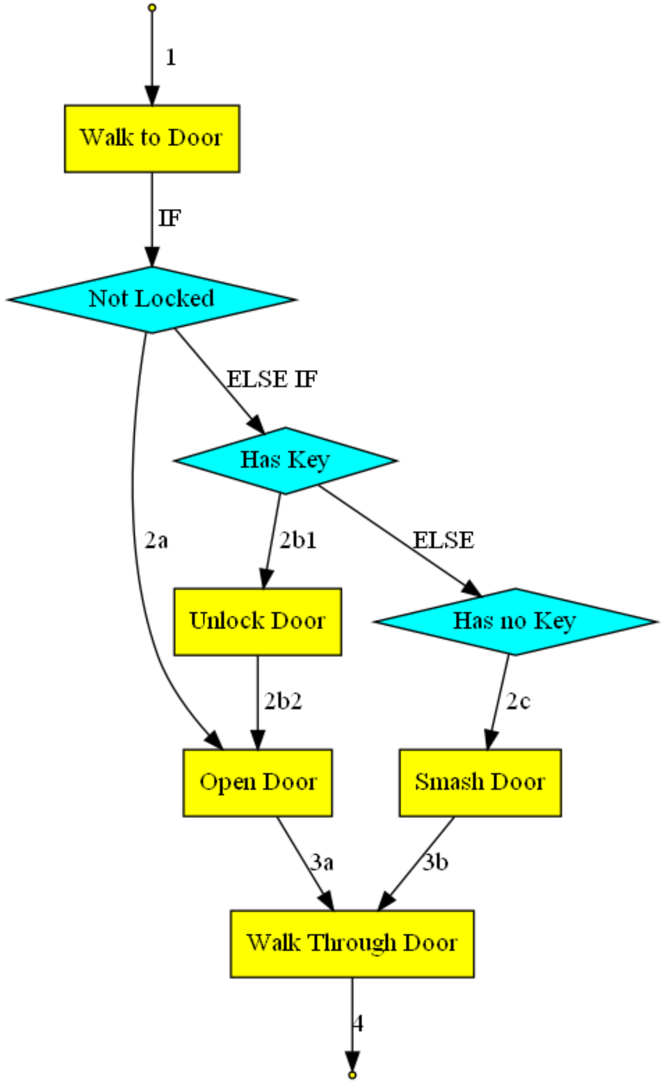
Grand Theft Auto IV (Rockstar Games)

Game AI State of the Art

Common techniques for decision making:

1. Ad-Hoc Algorithms

PROS	CONS
Easy and quick to set-up	AI behaviour is hard-coded
	Difficult to model complex behaviour
	Difficult to maintain
	Hardly reusable



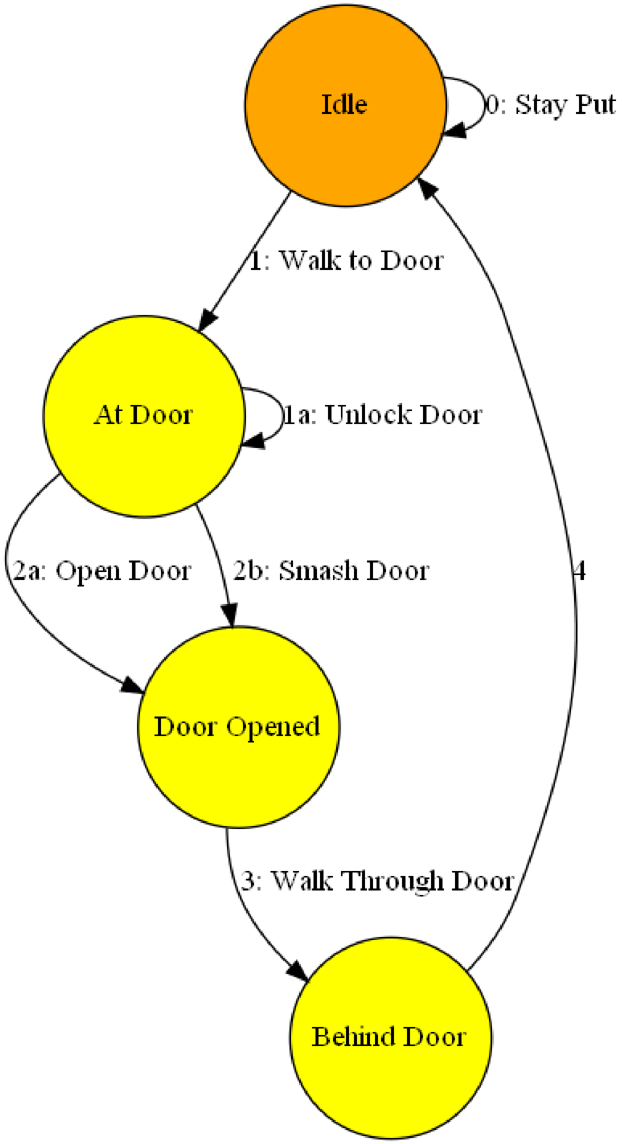
(Basic IF-THEN Algorithm)

Yannakakis, G. N. & Togelius, J. *Artificial Intelligence and Games.* (Springer, 2018)
Millington, I. *AI for games*. (Taylor & Francis, 2019)
Mark, D. *AI Architectures – What's on the Menu?* (Game Developer Magazine p.7-12, 2012)

Common techniques for decision making:

2. State Machines

PROS	CONS
Simple and effective	AI behaviour is hard-coded
Can be evolved into many variants (e.g., Hierarchical Finite State Machine)	Can become very complex to manage lots of behaviours



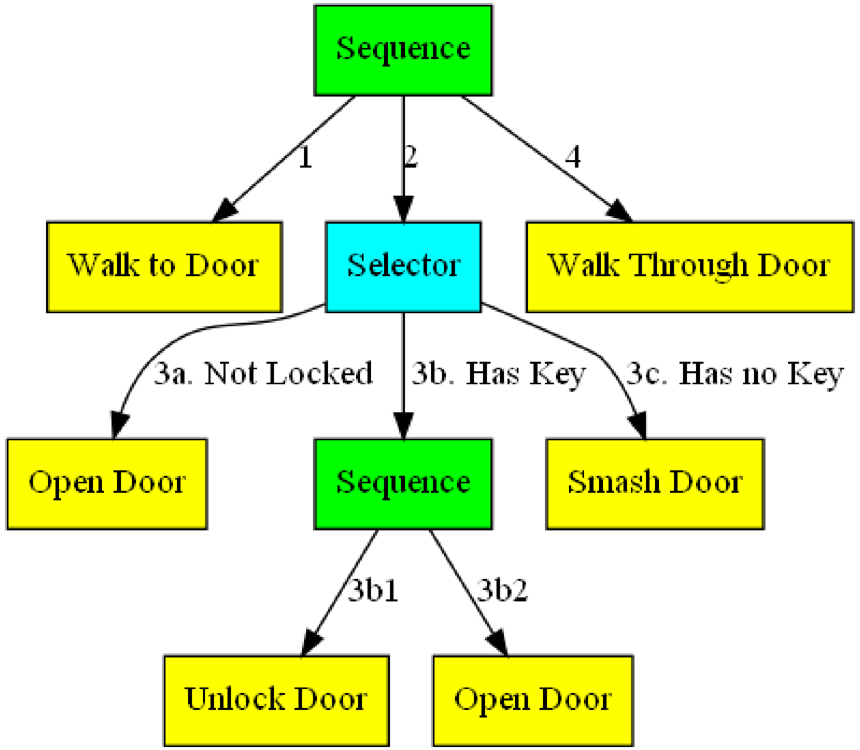
(FSM: Finite State Machine)

Yannakakis, G. N. & Togelius, J. *Artificial Intelligence and Games.* (Springer, 2018)
Millington, I. *AI for games*. (Taylor & Francis, 2019)
Mark, D. *AI Architectures – What's on the Menu?* (Game Developer Magazine p.7-12, 2012)

Common techniques for decision making:

3. Behaviour Trees

PROS	CONS
Great game engine integration (i.e. documented and easy to use)	AI behaviour is hard-coded
Complex behaviours can be managed with an interruption mechanism	Can become very complex to manage lots of behaviours
	Priorities of behaviours are hard-coded



(BT: Behaviour Trees)

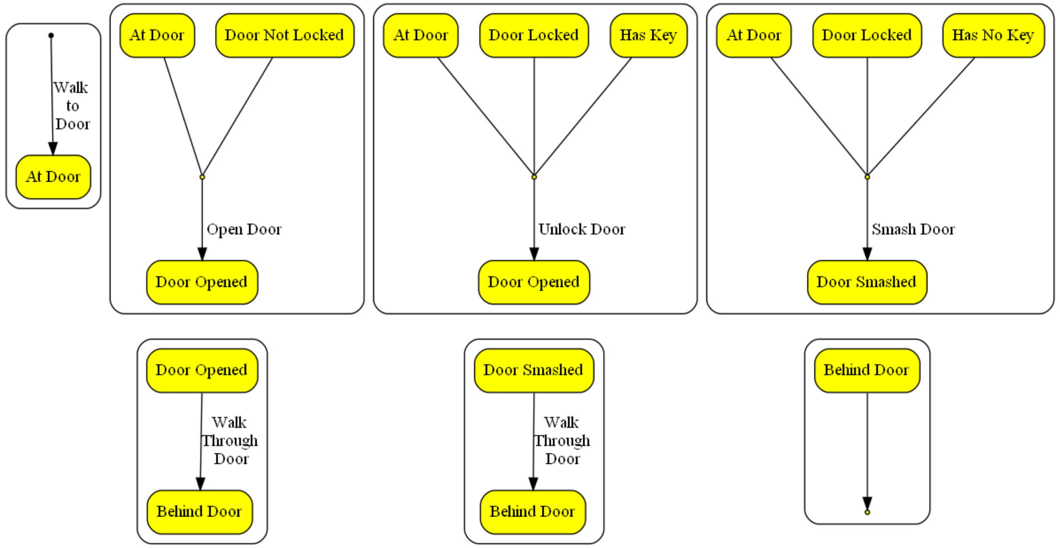
Yannakakis, G. N. & Togelius, J. *Artificial Intelligence and Games.* (Springer, 2018)
Millington, I. *AI for games*. (Taylor & Francis, 2019)
Mark, D. *AI Architectures – What's on the Menu?* (Game Developer Magazine p.7-12, 2012)

Game AI State of the Art

Common techniques for decision making:

4. Action Planning

PROS	CONS
AI behaviour is not hard-coded	Final behaviours are harder to predict for game designers
Modular for adding or removing actions	Planning can be costly with lots of goals and actions
	Need to define goals and how to prioritise them



(Goal-Oriented Action Planning)

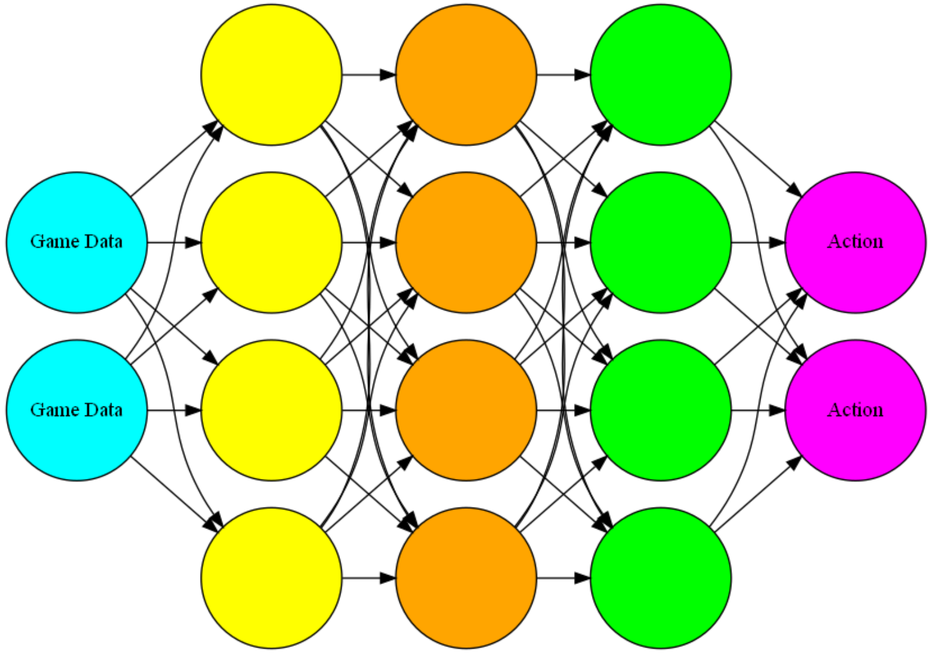
Yannakakis, G. N. & Togelius, J. *Artificial Intelligence and Games.* (Springer, 2018)
Millington, I. *AI for games*. (Taylor & Francis, 2019)
Mark, D. *AI Architectures – What's on the Menu?* (Game Developer Magazine p.7-12, 2012)

Game AI State of the Art

Uncommon techniques for decision making:

1. Neural Networks

PROS	CONS
AI Behaviour is not hard-coded	Final behaviour is harder to predict for game designers
	Need to be trained for each different NPC
	Need to be trained each time rules are changed
	Need lots of data for training



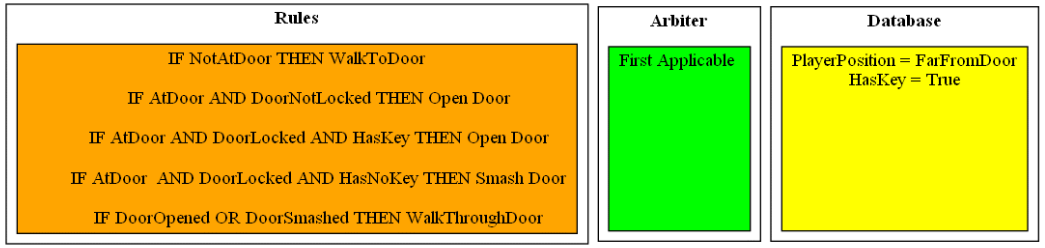
(Neural Network)

Yannakakis, G. N. & Togelius, J. *Artificial Intelligence and Games.* (Springer, 2018)
Millington, I. *AI for games*. (Taylor & Francis, 2019)
Mark, D. *AI Architectures – What's on the Menu?* (Game Developer Magazine p.7-12, 2012)

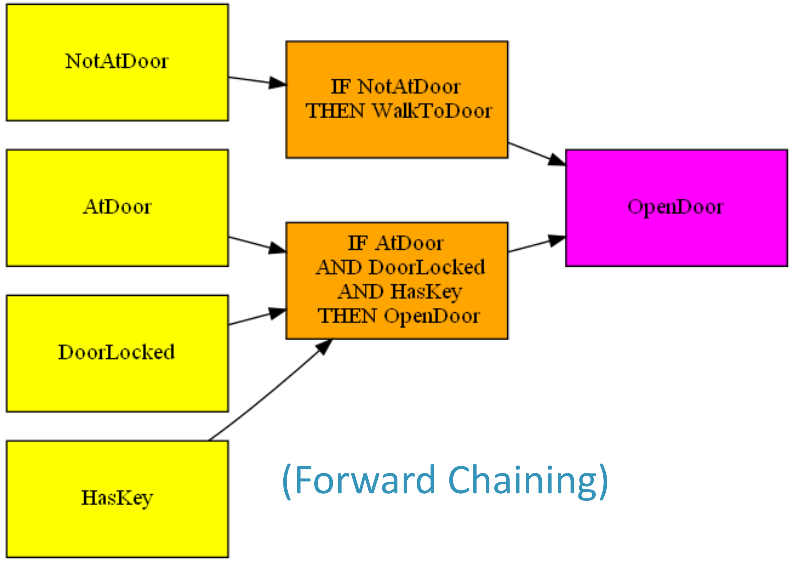
Uncommon techniques for decision making:

2. Rule-Based Systems

PROS	CONS
AI Behaviour is not hard-coded	Final behaviour is harder to predict for game designers
Justification for decision	Can be difficult to write good rules (a.k.a. knowledge acquisition)
Modular when adding or removing facts	Not popular, thus hardly supported
Forward chaining to generate new data from existing data	Slower with lots of rules (solutions exists: rule sets, rule pruning…)


Rules
IF NotAtDoor THEN WalkToDoor
IF AtDoor AND DoorNotLocked THEN Open Door
IF AtDoor AND DoorLocked AND HasKey THEN Open Door
IF AtDoor AND DoorLocked AND HasNoKey THEN Smash Door
IF DoorOpened OR DoorSmashed THEN WalkThroughDoor
Arbiter
First Applicable
Database
PlayerPosition = FarFromDoor
HasKey = True

(Ruled Based System)


NotAtDoor
IF NotAtDoor THEN WalkToDoor
AtDoor
IF AtDoor AND DoorLocked AND HasKey THEN OpenDoor
DoorLocked
HasKey
OpenDoor

(Forward Chaining)

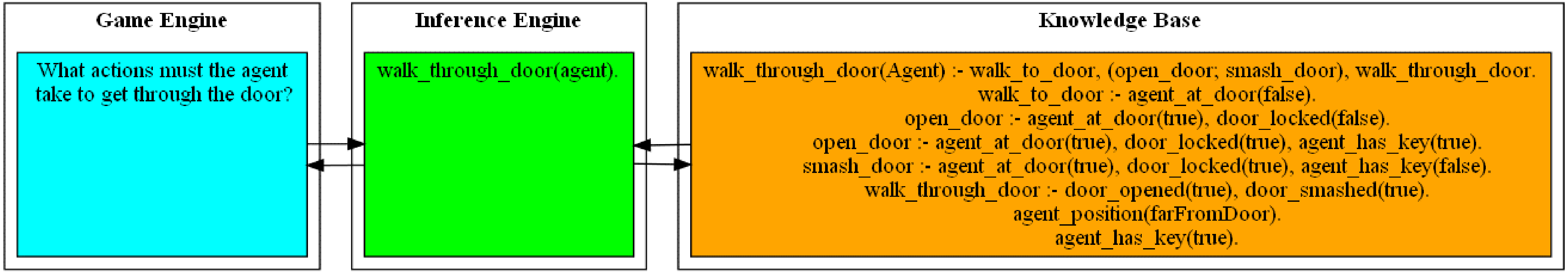
Yannakakis, G. N. & Togelius, J. *Artificial Intelligence and Games.* (Springer, 2018)
Millington, I. *AI for games*. (Taylor & Francis, 2019)
Mark, D. *AI Architectures – What's on the Menu?* (Game Developer Magazine p.7-12, 2012)

Ontology Reasoning to improve Decision Making

1. **Keep the dynamic aspect (as opposed to hard-coded) of GOAP & Rule Based Systems**

2. **Complexify the behaviour by adding ontologies and reasoning**
   (e.g. by inserting emotions, characteristics, environments, rules in society, etc.)

3. **Why Logic Programming?**
   1. **Usability in many domains:** (e.g. in Semantic web, Multi-Agent Systems (MAS), Planning, Databases, etc.)
   2. **Mature implementations:** (e.g. SWI-Prolog, XSB, SICstus, Mercury, etc.)



Game Engine

What actions must the agent take to get through the door?

Inference Engine

```
walk_through_door(agent).
```

Knowledge Base

```
walk_through_door(Agent) :- walk_to_door, (open_door; smash_door), walk_through_door.
walk_to_door :- agent_at_door(false).
open_door :- agent_at_door(true), door_locked(false).
open_door :- agent_at_door(true), door_locked(true), agent_has_key(true).
smash_door :- agent_at_door(true), door_locked(true), agent_has_key(false).
walk_through_door :- door_opened(true), door_smashed(true).
agent_position(farFromDoor).
agent_has_key(true).
```

(Ontology Reasoning)

Opportunities and Challenges

1. **Make Logic Programming easily usable for game developers & designers**
    a. Methodology: Modelling the communication between **game engine** & **inference engine**
    b. Friendly interface for Declarative Programming (GUI/Plugin)

    => Link with Knowledge Engineering

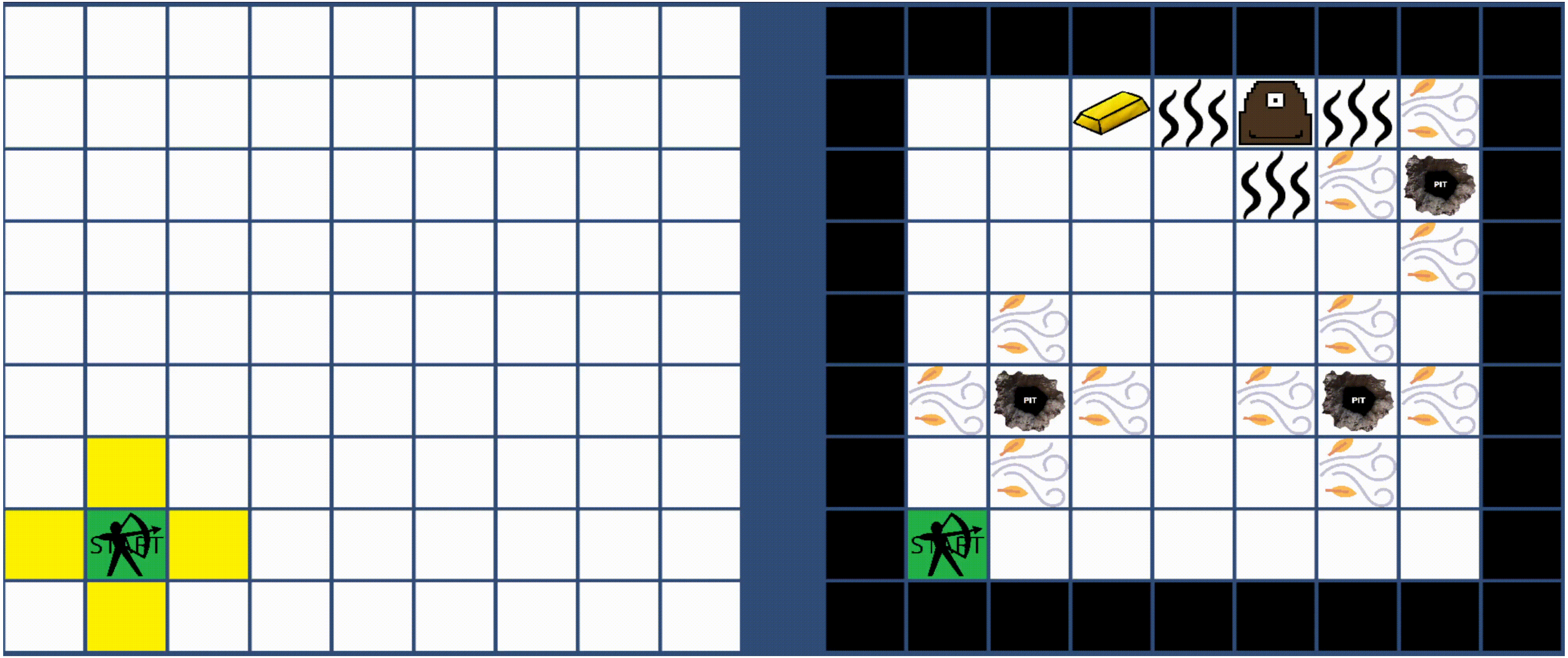
2. **Knowledge Representation and Reasoning extensions**
    a. Well-Founded Semantics (WFS) for negation and uncertainty
    b. Belief-Desire-Intention (BDI) paradigm usable for Multi-Agent Systems

3. **Transition from one agent to thousands is not trivial!**
    a. Optimization: Parallelization & Metaprogramming
    b. Dynamic reasoning resource allocation for each agent

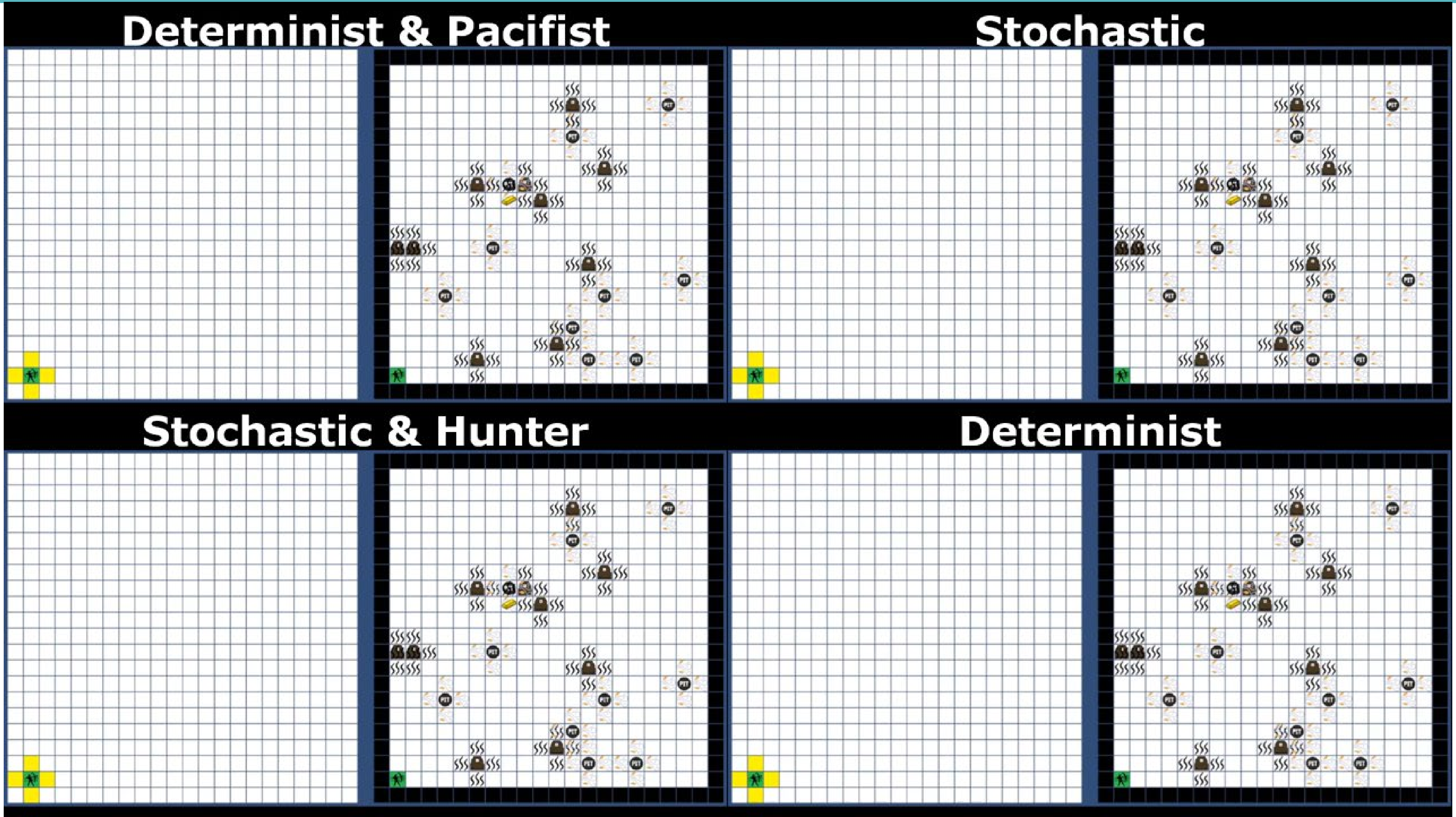
Bratman, M. *Intention, Plans, and Practical Reason*. (Cambridge, MA: Harvard University Press, 198–199, 1987).
Van Gelder, A., Ross, K. A. & Schlipf, J. S. *The well-founded semantics for general logic programs.* (J. ACM 38, 619–649, 1991)

Prototype

Video Game (Wumpus World) with Unity & SWI-Prolog:



Demo Wumpus World:



Research Collaboration

Collaboration with video game studio:

Wako Factory

Wako Factory Logo

Why this collaboration?

1. Make a true commercial game
   a) Industry developers feedbacks
   b) Actual players feedbacks
   c) Gameplay data to analyse
   d) Test the approach on a real game

2. This game could showcase the approach
   a. Wako wants the AI at the centre of the game
   b. AI decisions will be visible & justified

Visual from video game project Numini Stone (Wako Factory)