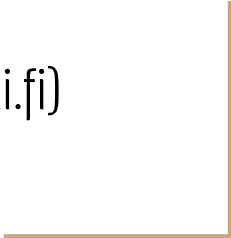




Agents And Interactions

Simo Linkola
([slinkola at cs.helsinki.fi](mailto:slinkola@cs.helsinki.fi))
7.11.16



Example: Robotic Installation

[Video 1 \(3:32\)](#)

[Video 2 \(4:18\) \(no speech\)](#)

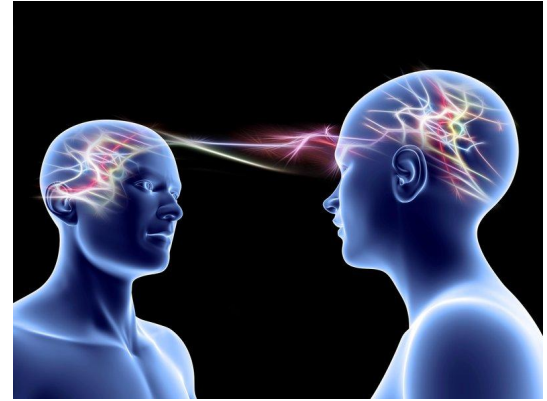


Outline

1. Characteristics of MAS for this course
2. Modeling the environment
3. Systems Model of Creativity
4. Interaction
5. Examples of Creative Agents and Societies

Recap of MAS Characteristics

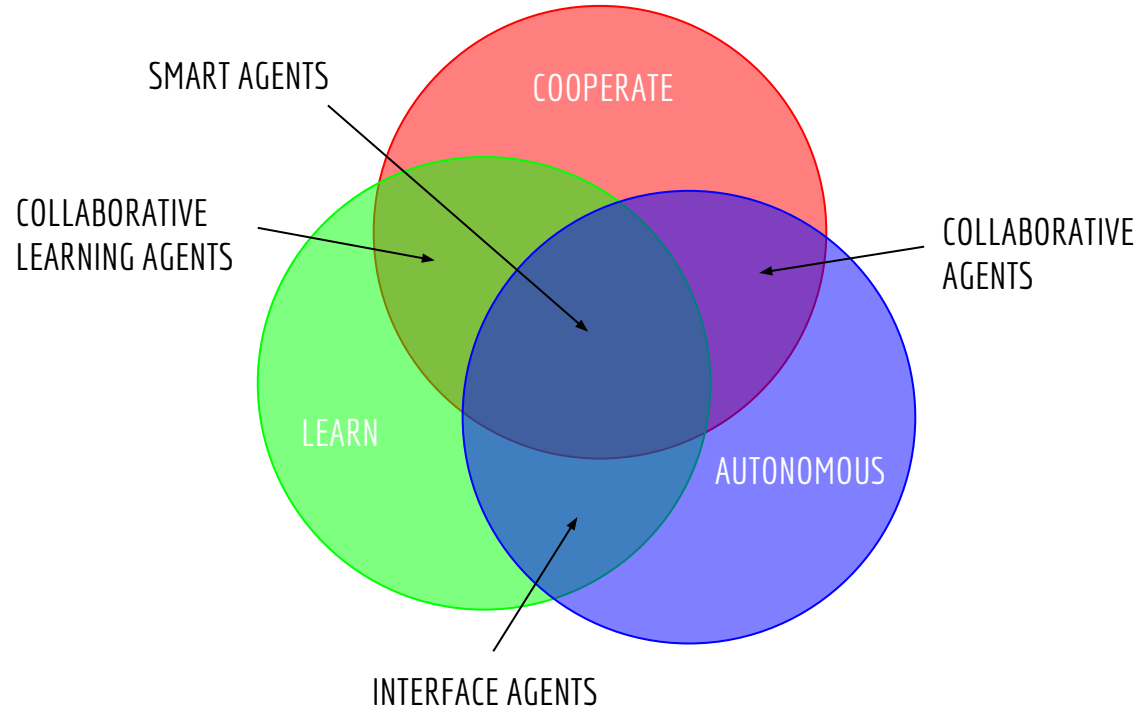
- Agents are autonomous and have local views
 - Agents take some individual responsibilities in the system
 - Agents can only “see” part of the system at any given time
- No central controlling agent
- There is always an environment, however, its role might be negligible
- Interaction between agents (either direct or indirect) is a central concept



Characteristics of MAS for This Course

- Mostly software agents
- Stationary societies
- Agents create artifacts in some domain(s)
- Agents have a way to evaluate artifacts
- Interaction is mostly direct

Nwana's Software Agent Topology



Stationary Agent Societies

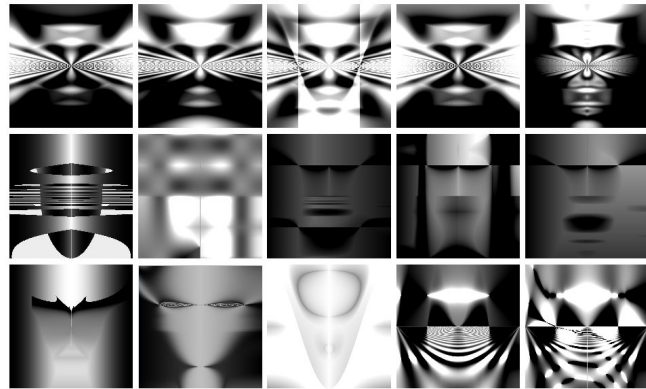
- Agents are (somewhat) homogenous
- Agents do not leave or enter after initialization
- Agents are generally assumed to be “well behaving”
- Agents may have connections to other agents at the initialization time
 - Even knowledge of all other agents
- In many cases, we will settle to iterative simulations
 - Useful for researching specific interests by simplifying the model

Artifacts

- Artifacts can be thought as solutions to a (search) problem
- Examples:
 - A piece of text generated with a Markov chain
 - An image created with genetic programming
 - A mathematical theory created via concept blending
 - Non-photorealistic rendering of an image

```
spec SPEC =  
  sort N  
  op  _+_ : N × N → N  
  op  p : N → N  
  op  s : N → N  
  op  zero : N  
  ∀ x, y : N • s(x) = s(y) ⇒ x = y  
  ∀ x, y : N • s(x) + y = s(x + y)  
  ∀ y : N • zero + y = y  
  ∀ x : N • s(p(x)) = x  
  ∀ x : N • p(s(x)) = x  
end
```

Listing 4: A consistent partial approach to the integers
(without order)



Evaluation

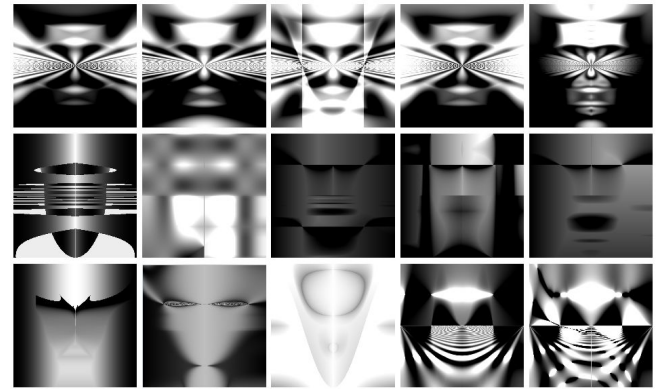
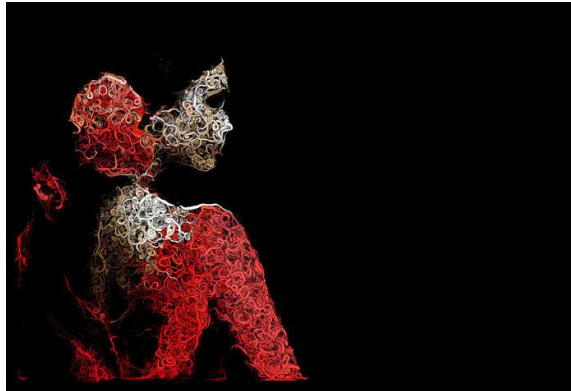
- Agents have some means to evaluate the artifacts they (and others) create
- Extract features from the artifacts (and combine them for a single value)
- In ideal case, evaluation takes all the following into account:
 - Is artifact novel?
 - Is artifact of high value?
 - Is artifact surprising?
- Designing good evaluation methods is difficult!



Evaluation: Examples

```
spec SPEC =  
  sort N  
  op  ___+_ : N × N → N  
  op  p : N → N  
  op  s : N → N  
  op  zero : N  
  ∀ x, y : N • s(x) = s(y) ⇒ x = y  
  ∀ x, y : N • s(x) + y = s(x + y)  
  ∀ y : N • zero + y = y  
  ∀ x : N • s(p(x)) = x  
  ∀ x : N • p(s(x)) = x  
end
```

Listing 4: A consistent partial approach to the integers
(without order)



Goal of the Agents

- Cooperative
 - All agents have the same goal, or
 - Agents benefit from fulfilling other agents' goals
 - Agents can also be called *altruistic*
- Competitive
 - All agents have their own goal they try to satisfy
 - Fulfilling a goal may render some other goals impossible
 - Agents can also be called *selfish*
- Mixture
 - Both cooperative and competitive agents
 - No considerable research in CC
- Intrinsic Motivation
 - See e.g. Schmidhuber



Goal of the Agents

Focus on this!

- Cooperative
 - All agents have the same goal, or
 - Agents benefit from fulfilling other agents' goals
 - Agents can also be called *altruistic*
- Competitive
 - All agents have their own goal they try to satisfy
 - Fulfilling a goal may render some other goals impossible
 - Agents can also be called *selfish*
- Mixture
 - Both cooperative and competitive agents
 - No considerable research in CC
- Intrinsic Motivation
 - See e.g. Schmidhuber



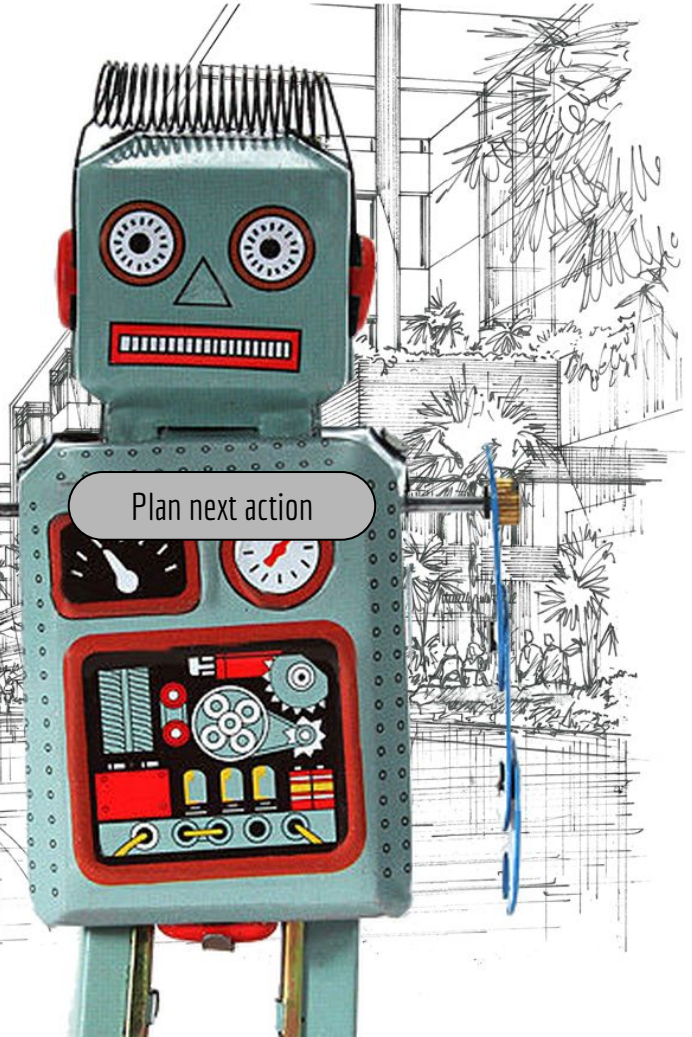
Modeling the Environment

- Way for an agent to reason about its choices
- Enhances adaptation (if done correctly)
- For example, use reinforcement learning
 - Tries to maximize expected utility (reward) in the long run
 - Selects “optimal” action in each situation (state)
 - E.g. Q-learning

Modeling the Environment

SELECT ACTION WITH THE
HIGHEST EXPECTED UTILITY

Plan next action

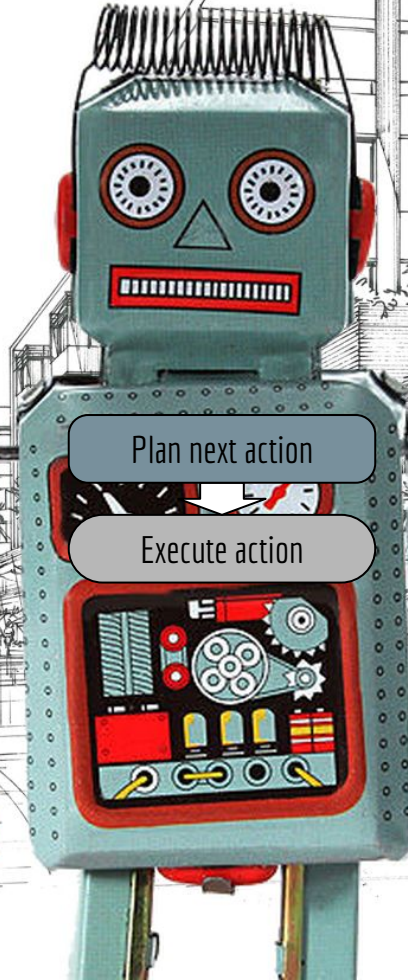


Modeling the Environment

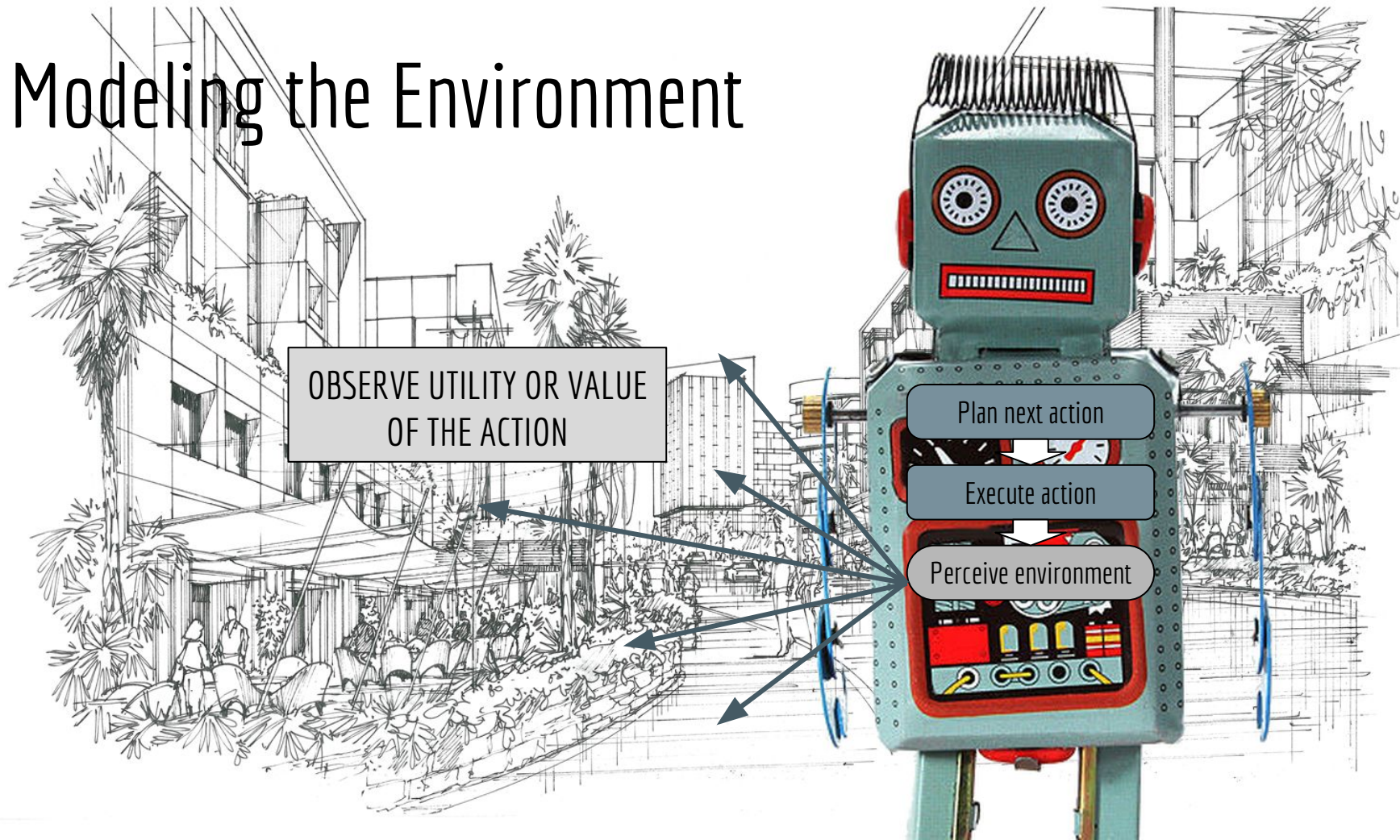
EXECUTE THE PLANNED
ACTION

Plan next action

Execute action



Modeling the Environment



Modeling the Environment

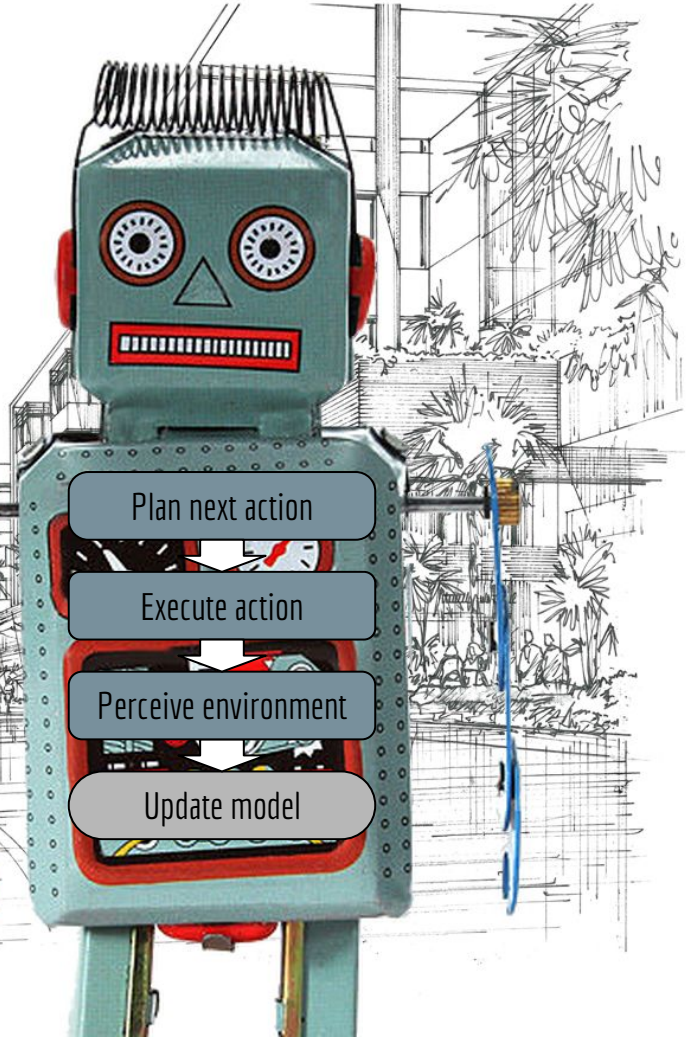
UPDATE MODEL BASED ON
THE OBSERVED UTILITY

Plan next action

Execute action

Perceive environment

Update model



Modeling the Environment

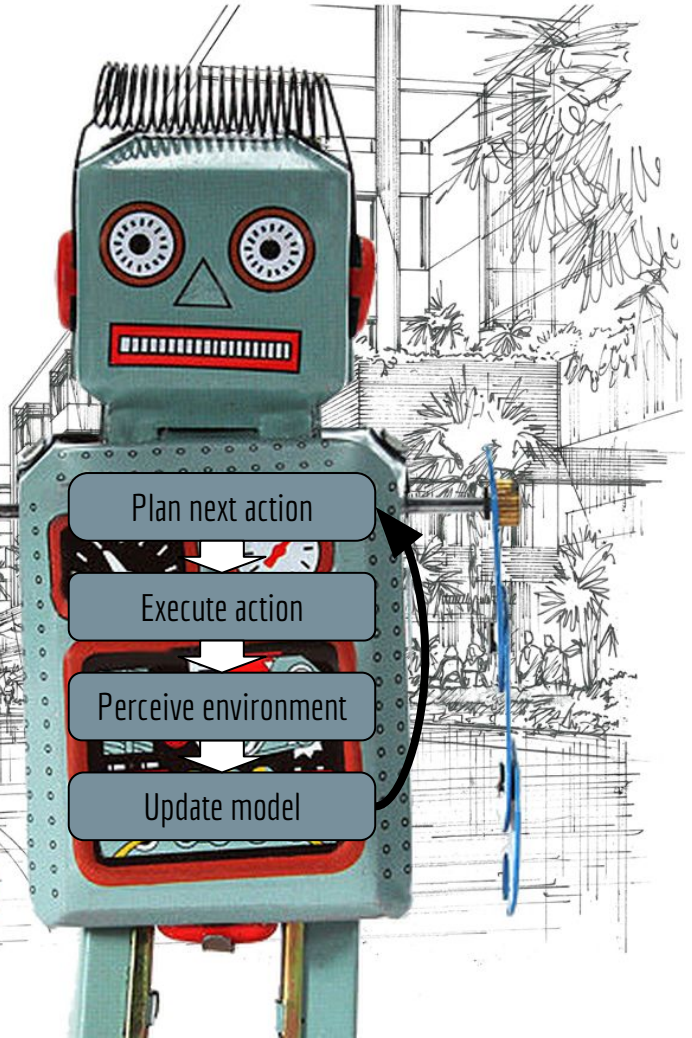
START FROM THE
BEGINNING

Plan next action

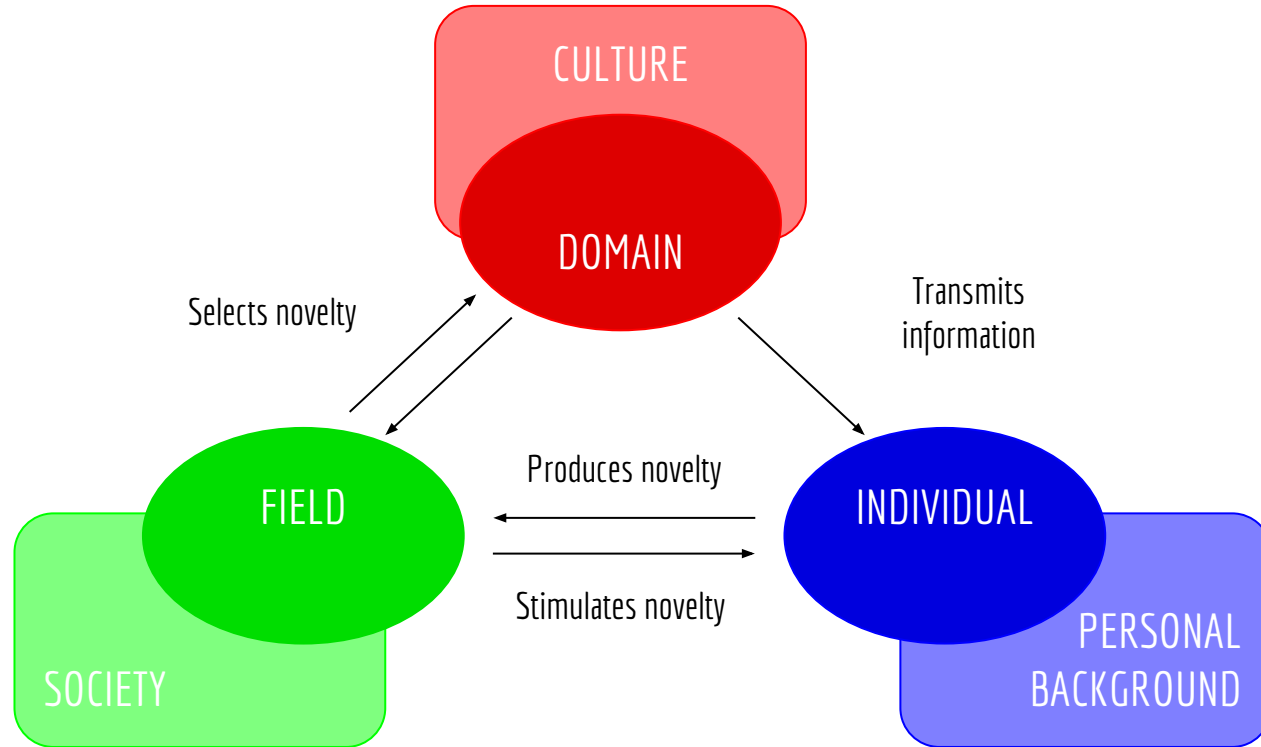
Execute action

Perceive environment

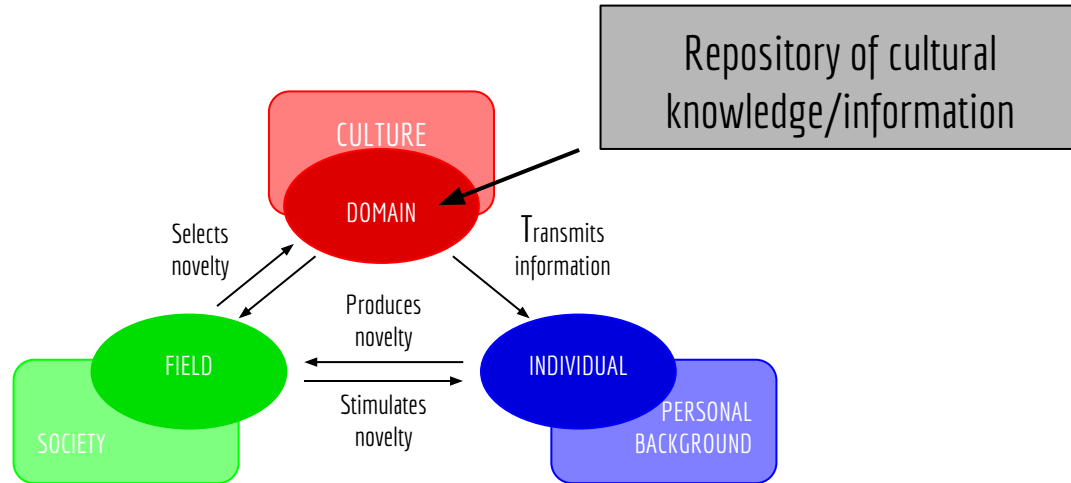
Update model



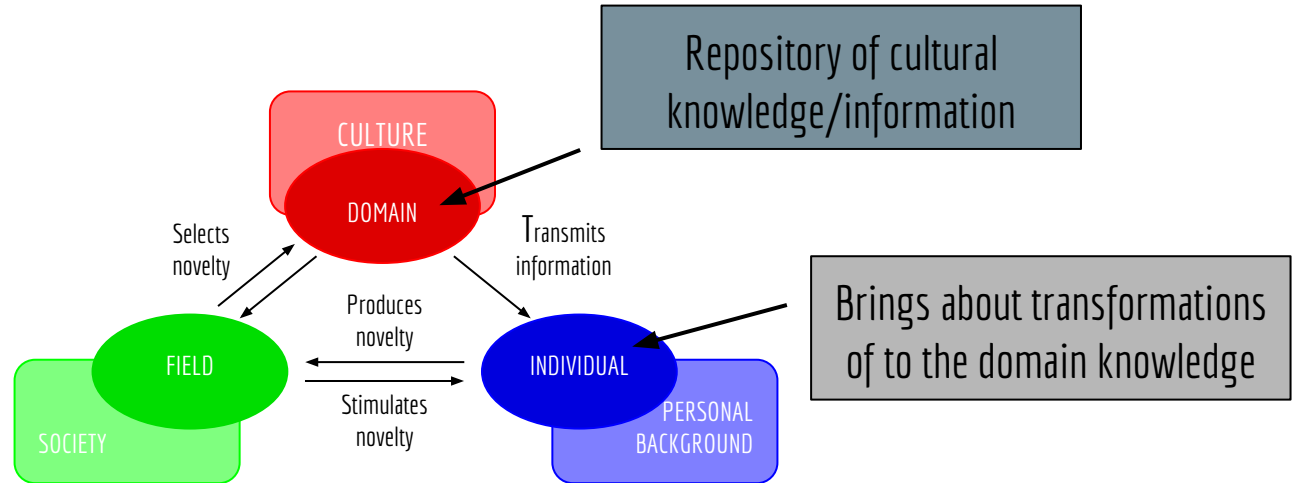
The Systems Model of Creativity



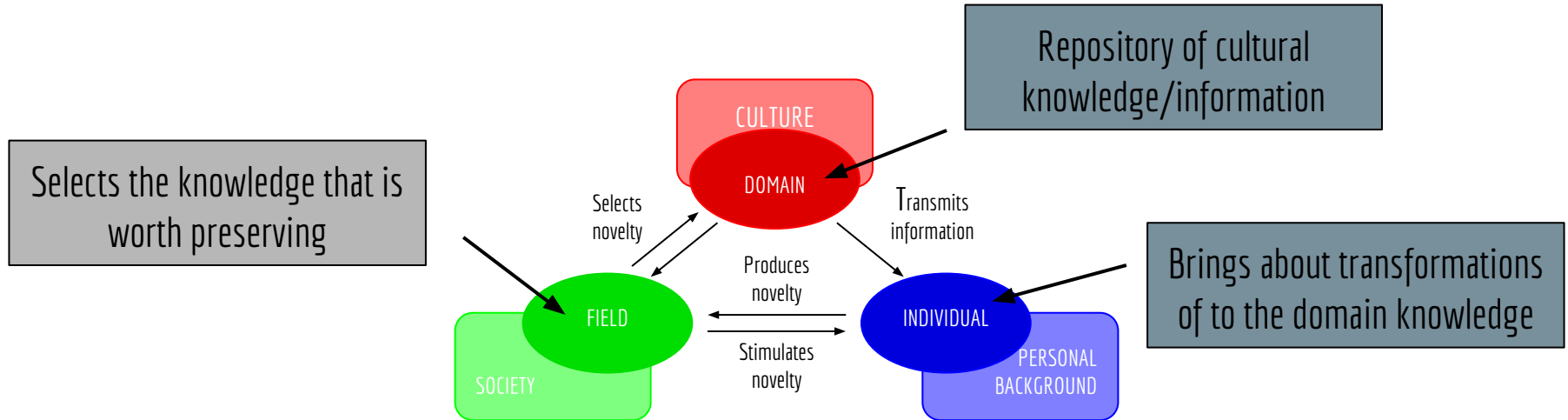
The Systems Model of Creativity



The Systems Model of Creativity



The Systems Model of Creativity



Interaction

- Agents interact with each other one way or another
- We are mostly interested in direct interaction
- Some specific interaction types:
 - Negotiation
 - Voting
- Emerging social phenomena:
 - “Best friends”
 - Cliques
 - Gatekeepers
 - Other social structures

Interaction

- Communication model can be nearly anything:
 - Tell a friend
 - Report to a supervisor
 - Broadcast to peers
 - Write on a blackboard
- Interest in models which restrict communication
 - Broadcasting everything causes bandwidth problems
 - Agents are modeled to have limited resources

Negotiation

- Agents negotiate over:
 - a contract
 - the terms of transactions
 - etc.
- Typical in Consumer - Producer models, e.g. e-commerce
- May happen between only two agents or a set of agents
- Typically iterative



Negotiation: Example

Agent A: Give me service **X**.

Agent B: I'll give you service **X** if you give me service **Y**.

Agent A: Give me service **X** and I'll give you service **Z**.

... etc.

Voting

- A set of agents **A**
- A set of candidates **C**
- Each agent in **A** votes by returning **C** sorted by preference
- Selected voting rule defines which candidate(s) get selected
- Voting rules:
 - Majority, Condorcet
 - Dictatorship
 - Run-off rules
 - Etc.
- No silver bullet, have to pick the “least evil” option (voting rule)
- Example: Agents vote for the best artifact from the set of artifacts



Emerging Social Phenomena

- Cliques / coalition formation
 - Agents form tight social circles where they like artifacts produced by the agents in their circle more than agents outside the circle.
- Gatekeepers
 - Some agents form a role of a “gatekeeper”, where they have substantially more control on the artifacts that are accepted to the domain than other agents.
- Etc.

Computational Social Creativity

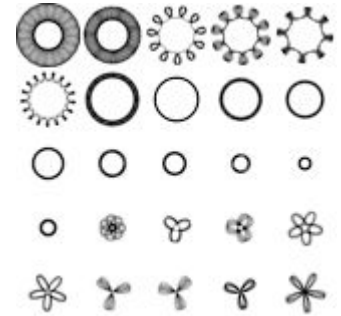
- Research interests mainly in super-individual properties (emergence)
- What are the individual properties and interaction types that cause certain phenomena? (proof-of-concepts)
- In this course we are interested in both:
 - Computational social creativity
 - More traditional research in CC with a twist of multiple interacting agents

Evolution of Culture (EVOC) by Liane Gabora

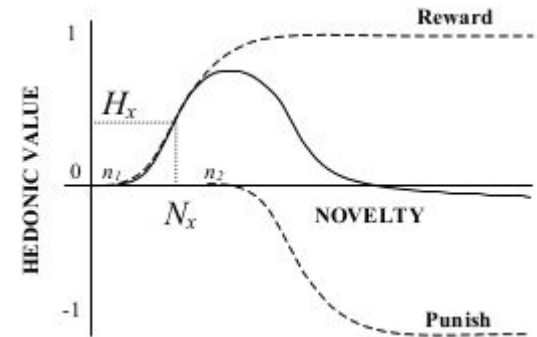
- Society of neural network based agents in a 2D grid-world:
 - Neural network encodes ideas for actions and detects trends which constitutes as fit actions
 - Body implements the actions
 - Agent's goal is to attract a mate and make tools (two fitness functions)
- Cultural phenomena:
 - Imitation: agents copy neighbours actions, allows sharing of effective actions
 - Invention: agents can modify their previous actions
 - Knowledge-based operations: new ideas are generated strategically (i.e. not random)
 - Mental stimulation: before implementing an idea, the agent simulates its usage
- Interest in diversity of actions in the population as a function of time

Curious Design Agents

by Rob Saunders and John Gero



- Agents aim to produce novel artifacts (spirographs)
- Agents have a model (SOM) of previously seen artifacts
- Agents evaluate the novelty of an artifact w.r.t the model
- Agents have different levels of preferred novelty
- Hedonic function follows the Wundt Curve
- Agents may give positive feedback of artifacts
- Emergence of cliques is observed
 - Agents that give positive feedback to each other
 - Agents in a clique have the same level of preferred novelty



Reflexive Looper by Sony CSL

- Part of [Flow Machines](#) project lead by François Pachet
- Musical companion which adapts to you
- Plays chords, drums, bass, etc.
- [Video \(4:09\)](#)

Social Models of Creativity

by Ricardo Sosa and John Gero

- Study creativity in the DIFI-setting
- Agents make designs in simple two-dimensional shapes
- Designers and adopters
- Findings about many individual and social aspects of creativity, e.g.
 - Change agents: trigger change cycles in a bottom-up direction
 - Gatekeeping: some adopters become gatekeepers by an aggregate result of social interaction

