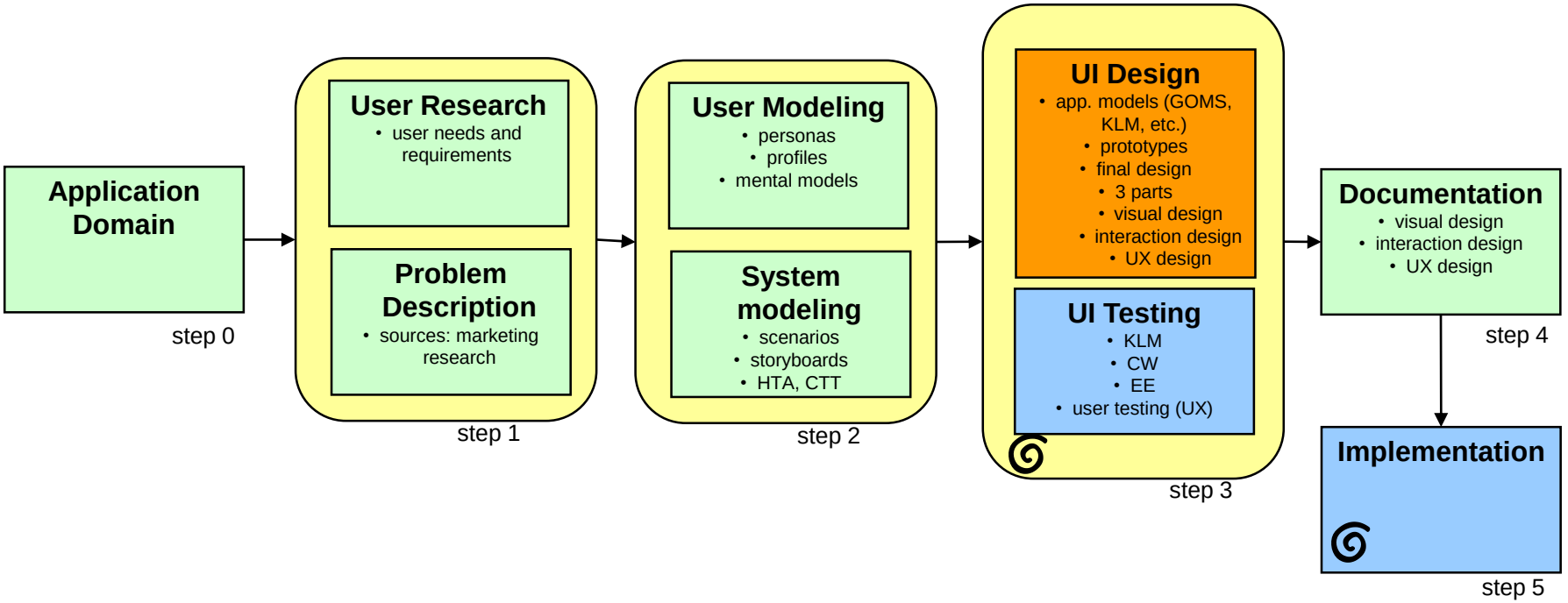
# NUR Intelligent UI

# User interface design - big picture



# Intelligent system

- Reasoning
- Learning
- Adaptability



# Intelligent system - reasoning

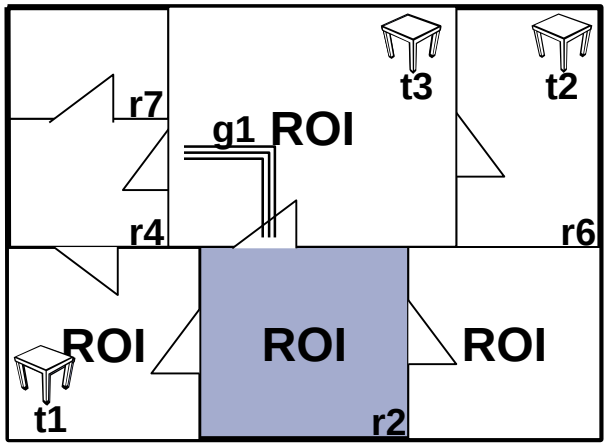
- Derivation of a new knowledge
  - not previously contained in the system

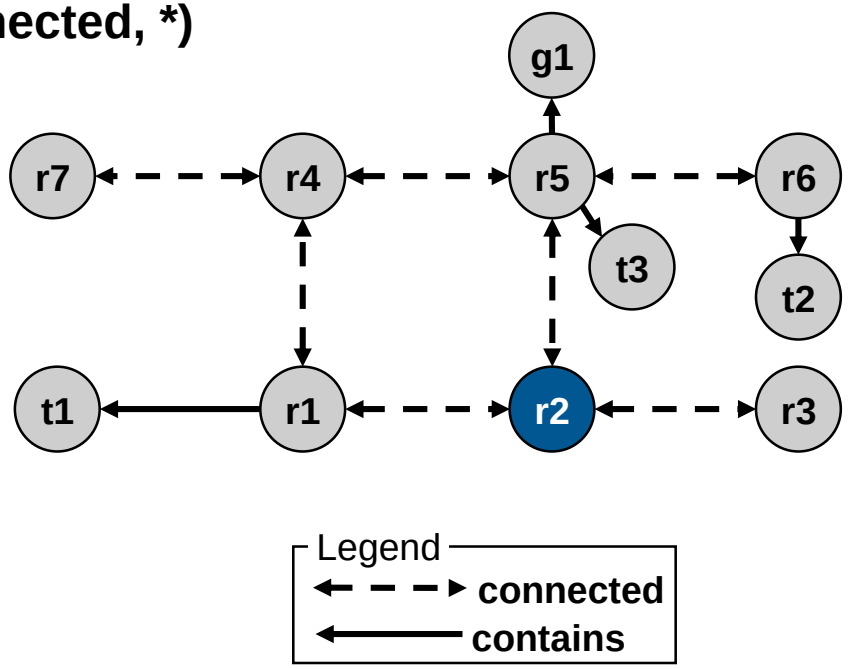
- New knowledge is based on the existing knowledge

# Reasoning Example

**ROI definition: (object$_1$ , semantic_relation, object$_2$)**

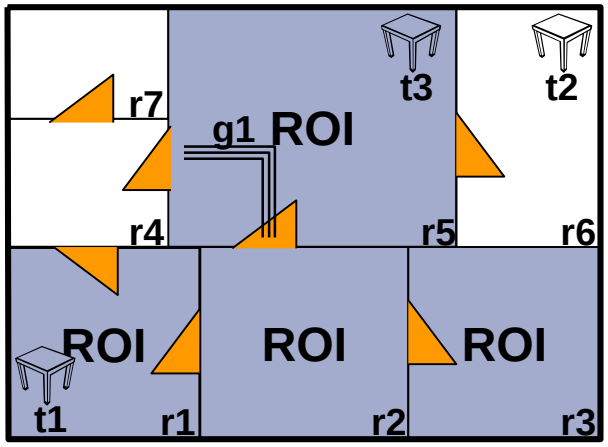
**ROI: (r2 , connected, *)**

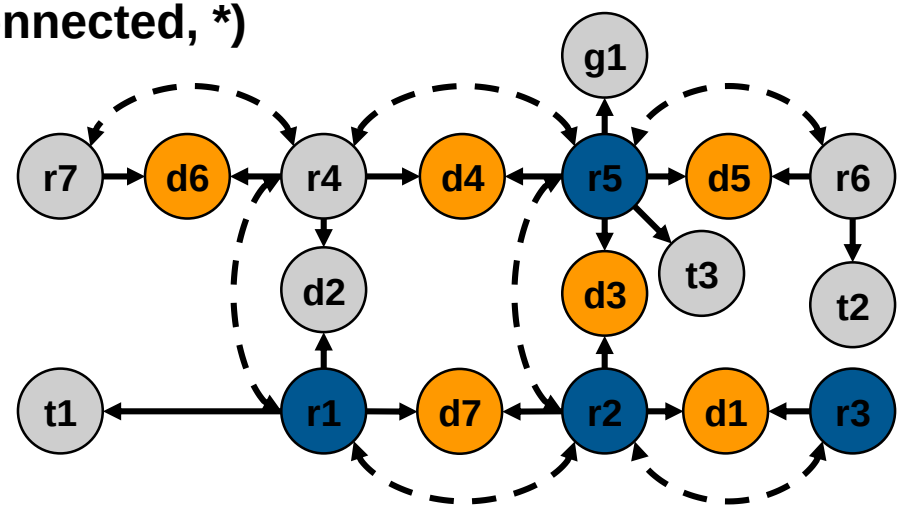


# Reasoning Example

**ROI: (r2 , connected, *)**



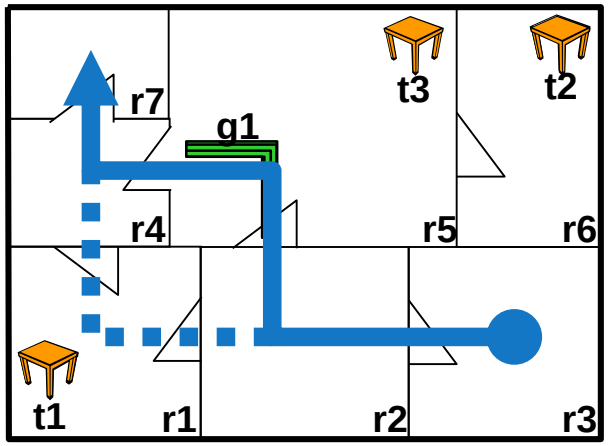




| | |
|---|---|
| rule template: | connected: **R**(**X**,**Y**) ⇐ door(**A**), **R**(**X**,**A**), **R**(**Y**,**A**). |
| query: | connected: contains(r1,r4)? |



# Reasoning Example

**ROI: (r3 , safeconn, r7)**




rule template:

| | |
|---|---|
| saferoom(**X**) | ⇐ room(**X**), safe(**X**). |
| safe(**X**) | ⇐ contains(**X**,g). |
| safe(**X**) | ⇐ not:dangerous(**X**). |
| dangerous(**X**) | ⇐ contains(**X**,t). |
| safeconn(**X**,**Y**) | ⇐ saferoom(**X**), saferoom(**Y**), connected: contains(**X**,**Y**). |

query:

safepath:safeconn(r3,r7)?

safepath:safeconn(r3,r7,**P**)?

# Intelligent Interactive System

- Systems that provide interactive support based on embedded AI mechanisms
- Provides interface to AI functionality and knowledge representations

Examples

- Adaptive systems
- Cooperative problem-solving systems



# Intelligent User Interfaces (IUI)

- A bridge between user and machine that aims to improve the efficiency, effectiveness and naturalness of interaction by representing, reasoning, and acting on models of the user,  domain, task, discourse, and media.



# Reasons for introduction of IUI

- Make the communication more natural
  - natural language conversation
  - talking head
- Lower cognitive load
  - personalization of UI and the content to user needs
  - efficient UI control and content presentation
- Accessibility
  - adaptation of UI and content to user capabilities
- Increase the flexibility
  - automatic UI generation based on the context (tasks, environment, user model)
- Allow autonomous work



# Intelligent versus Intuitive Interfaces

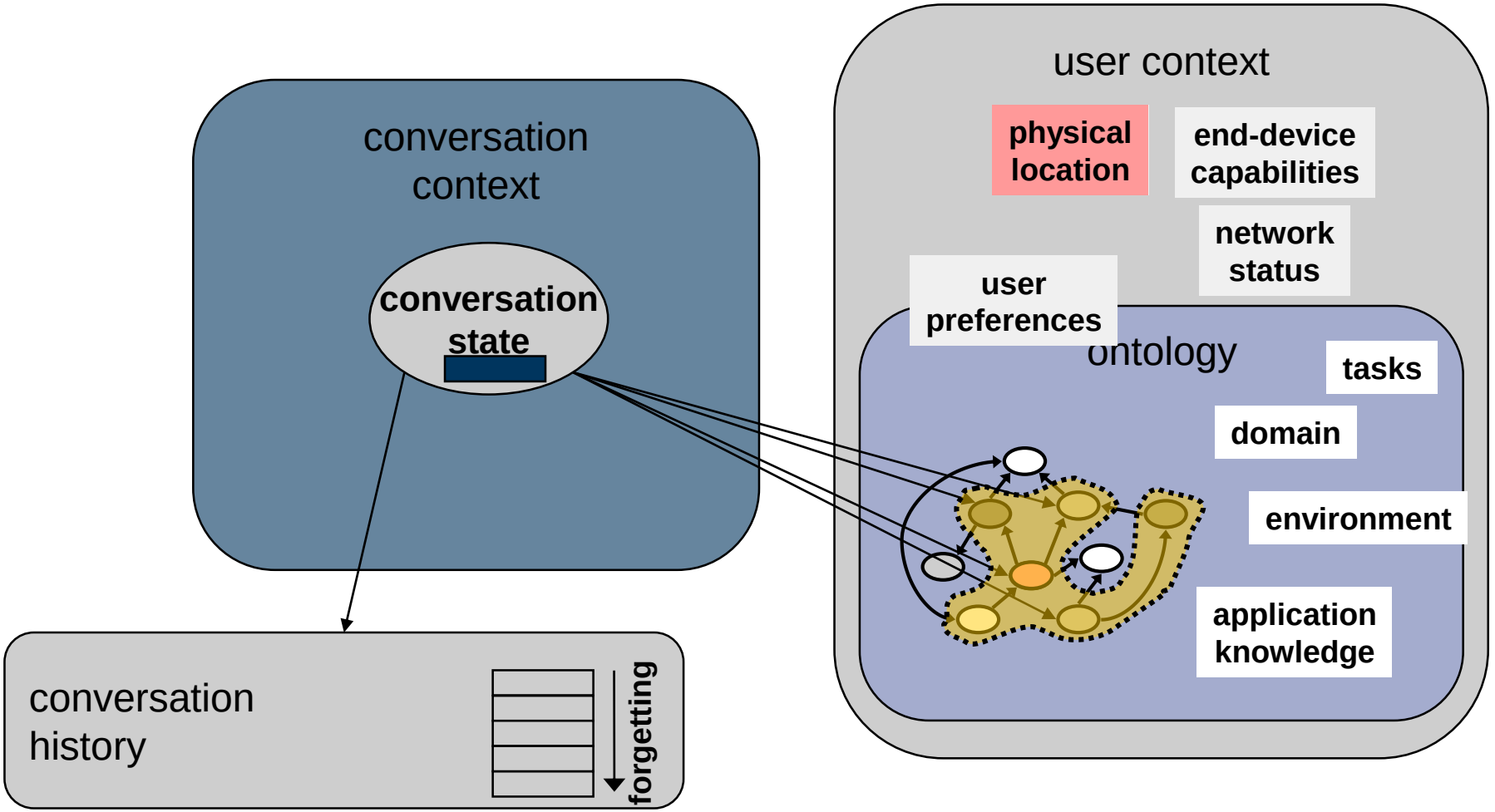
- Interface - a device that bridges different systems, people, ideas, etc. (interpretation and generation)
- Intuitive - having immediate mental perception or understanding (natural, no training)
- Intelligent - capable of communicating and reasoning (user, task, dialogue, information, media)

# Models used in IUI design

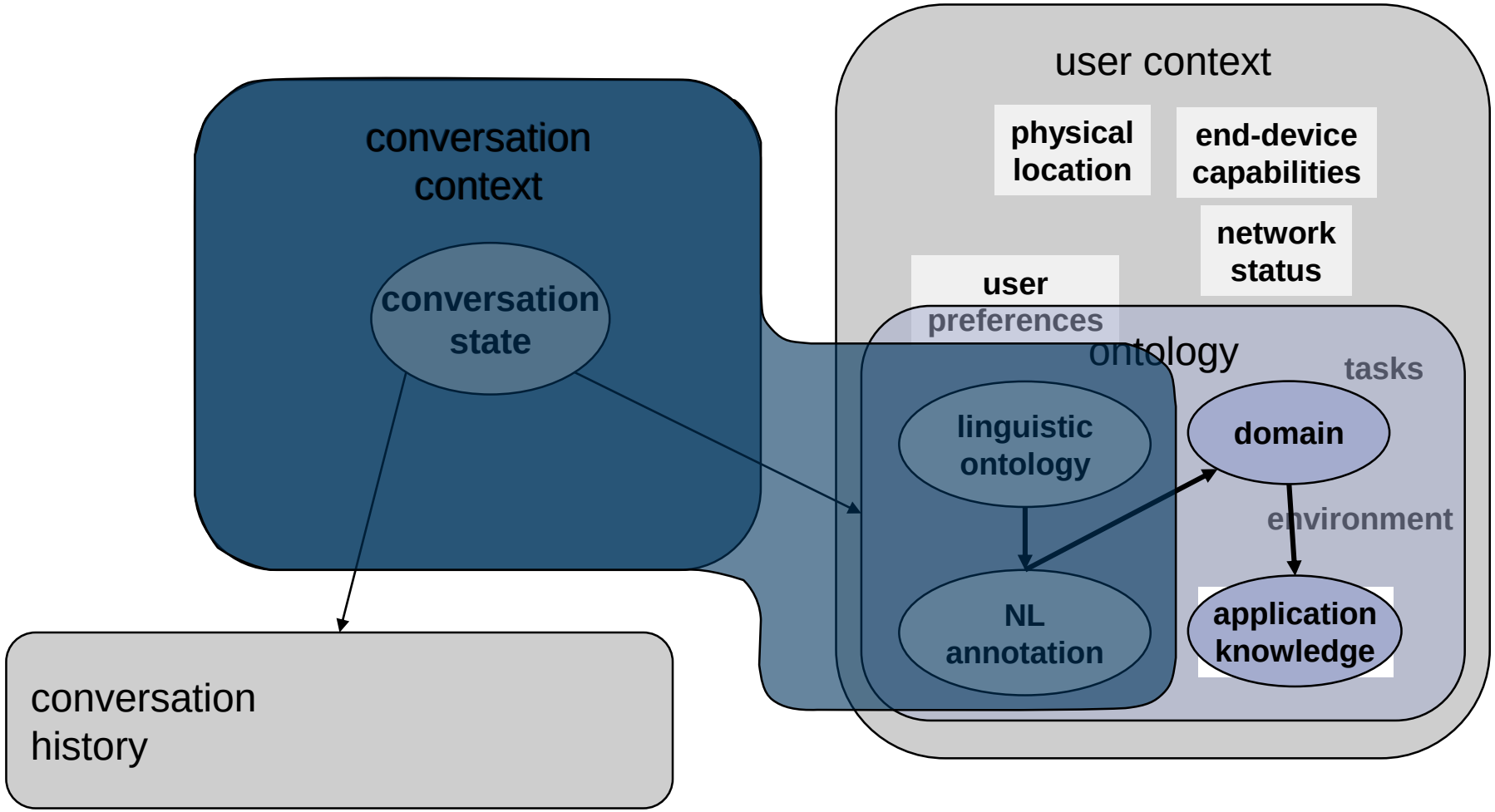
- Environment model
- Conversation model
- Domain model
- User model
- Behavior model
- Physical model



# Example: Conversation system

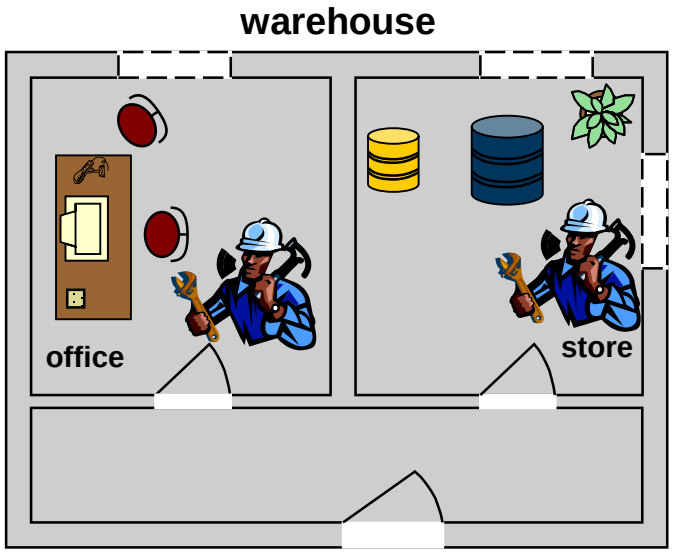


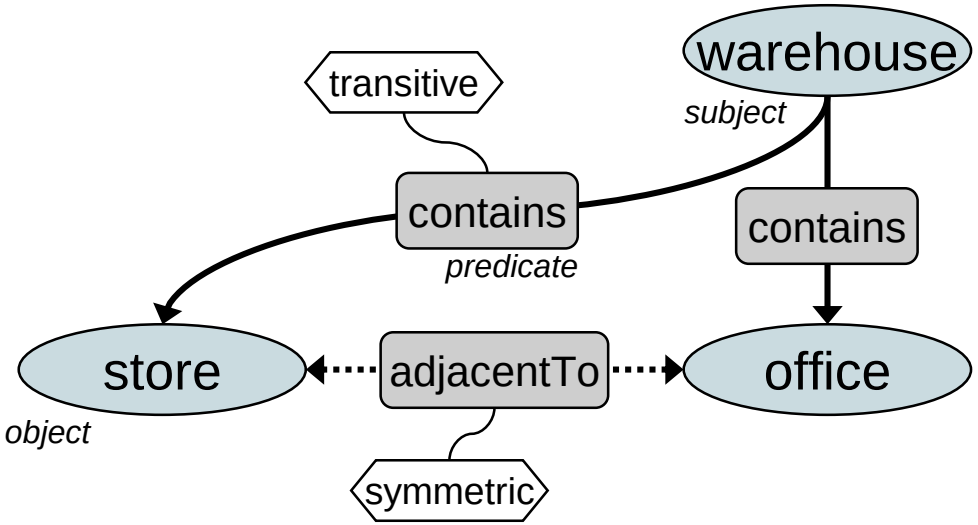
# Example: Conversation system

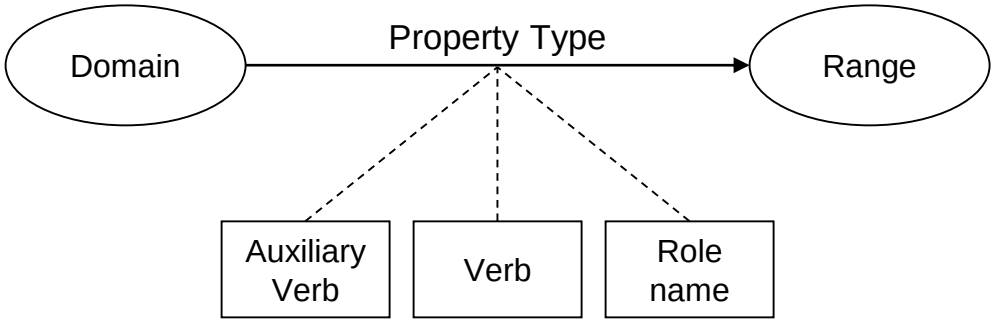


# Example: Conversation system

# Example: Domain model enriched by NL



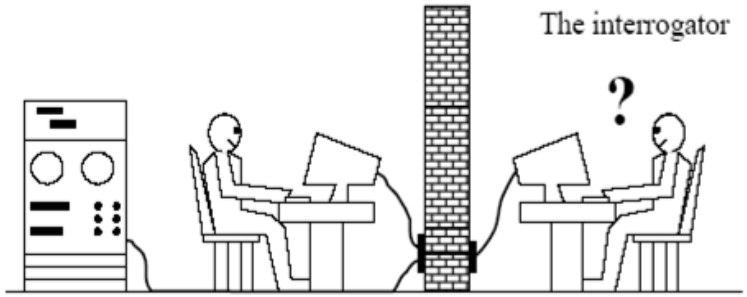

Natural language attributes of a property type

# How to discover Intelligence in an interactive system



# Turing test

- Test whether we discover that our partner is a computer
- Behavioral view of intelligence
  - A behavior capable of fooling a human interrogator
- Acting humanly is sufficient to pass the test
  - Is it necessary the AI system also thinks humanly?



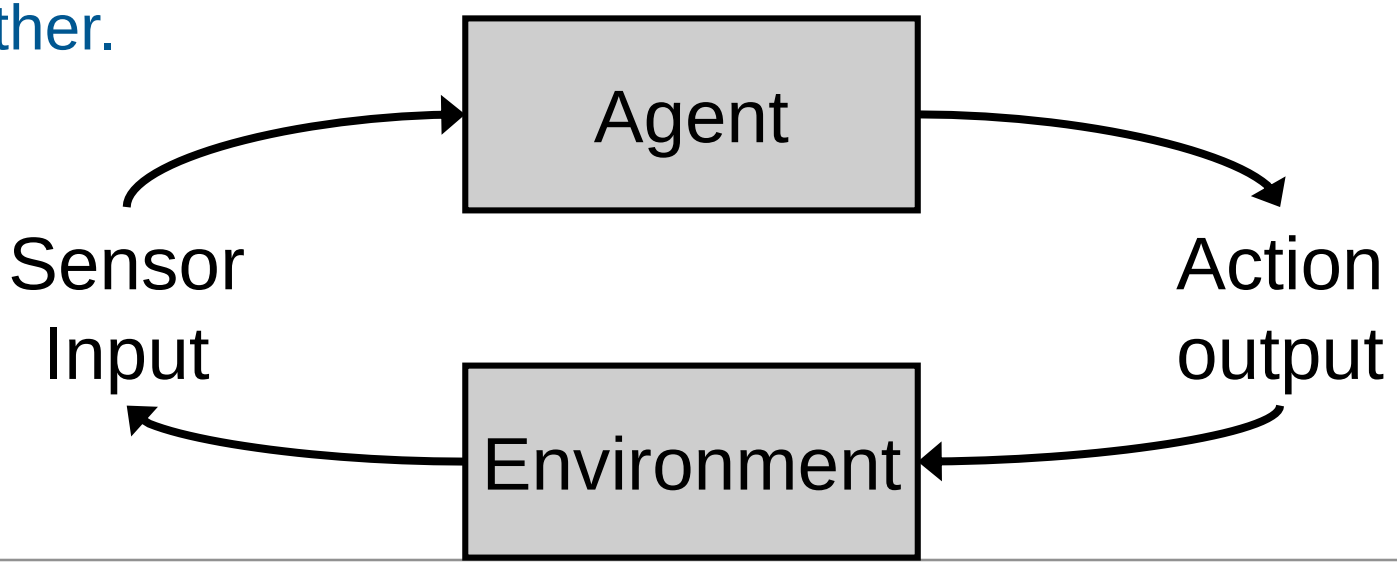
# Agents and intelligence in user interfaces



# Agent definition

- Agent is a theoretical concept from AI
- There is no single universal definition of an agent

- Agent in Webster's New World Dictionary: A person or thing that acts or is capable of acting or is empowered to act, for another.

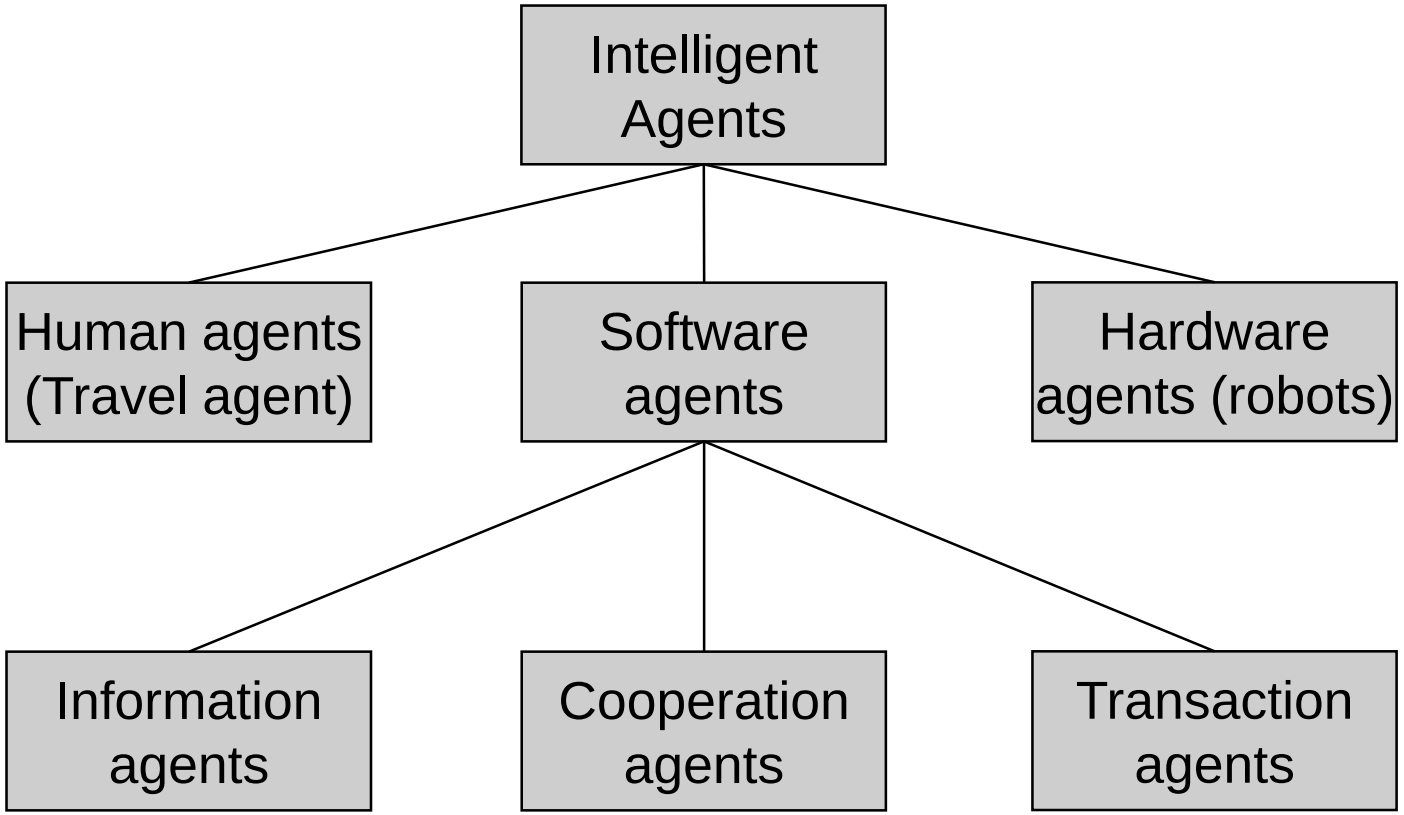


# What is an intelligent agent?

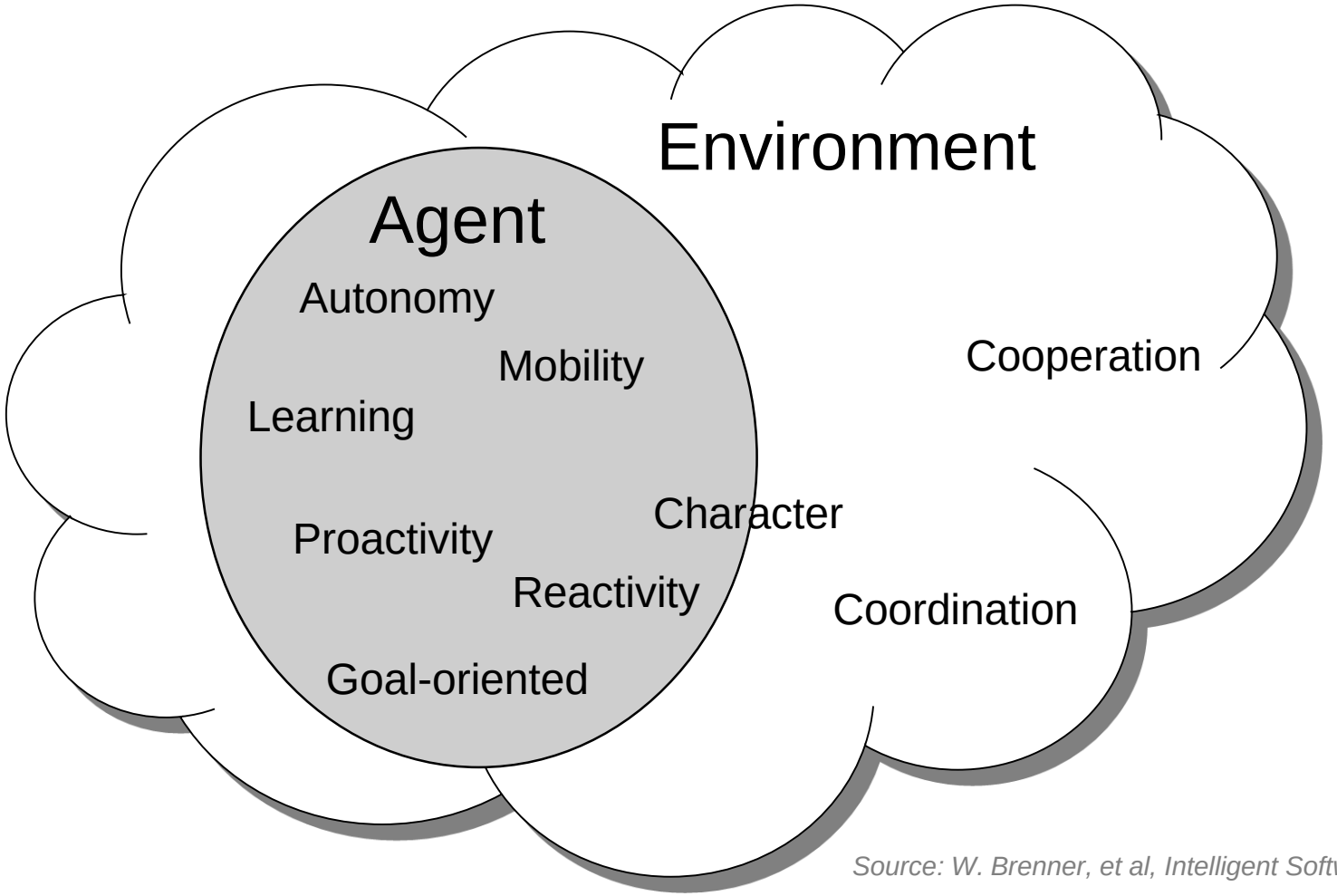
- Intelligent agent is a computer system located in certain environment and is capable to react in a flexible way on events in its environment



# Agent Types


Intelligent Agents
Human agents (Travel agent)
Software agents
Hardware agents (robots)
Information agents
Cooperation agents
Transaction agents

# Characteristics of SW agents




*Source: W. Brenner, et al, Intelligent Software Agents*

# Software Agents

- One view:

  Software processes that have non-trivial tasks delegated to them which require independent action and a report on the results.



# SW Agents: Autonomy

- Agent's activities are autonomous (no statements from the user)
- Properties of an agent: mobility, ability to communicate, ability to learn, …



# SW Agents: Intelligence (necessary attributes)

- Reasoning: agent monitors environment and takes decisions (based on changes in the environment)
- Learning: agent's behavior is improving (based on previous experience)
- Adaptability: agent is able of adaptation to changes in its environment (robustness)



# SW Agents: Mobility

- Agent mobility = “traveling” from one computer to another one



# SW Agents: Cooperation with other agents

- Cooperation between agents makes possible  to solve the problems much faster (usually the solution is better)
- Language for cooperation description



# SW Agents: Emotions

- Agents can communicate with people. It is desirable to humanize the agents



# Issues for Software Agents

- Personification
  - Should agents be represented as a living or animated character?
  - Does it improve adoption of software?
  - Does it create inflated expectations?
  - Is it just too annoying?

# Talking Head: Real-time generation

# Talking Head: Interpolation between expressions

Surprised

*Actual*

Sad

*Actual*

Worried

*Interpolated*

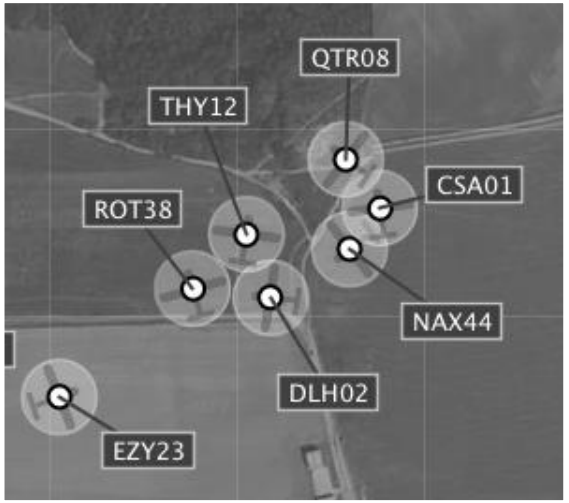
# Issues for Software Agents

- Trust and Competence
  - How does user develop an informed level of trust?
  - Can agent give self-assessment on likely outcome of task?
- Delegation
  - How can user delegate tasks?
  - How can user check on status of delegated tasks?

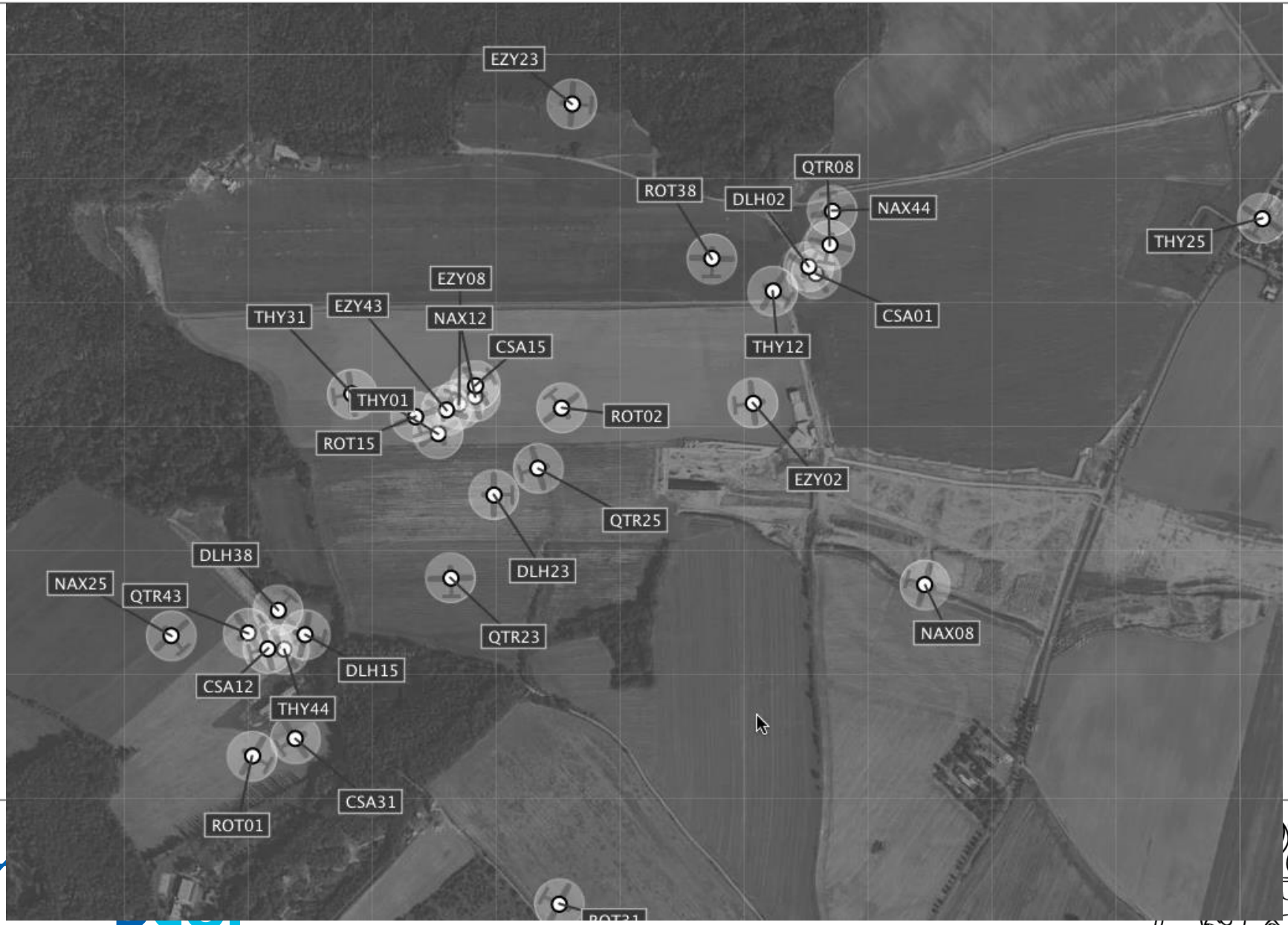


# Issues for Software Agents

- Control
  - How does user set limits on the agent's activity?
  - When does the agent get to interrupt the user (mixed-initiative dialog)?
- Dealing with multiple agents
  - How can the user manage many agents?
  - How can interactions between agents be predicted?



# Issues for Software Agents: Multiple agents



# Automation and human control

- Users can avoid:
  - Routine, tedious, and error prone tasks

- Users can concentrate on:
  - Making critical decisions, coping with unexpected situations, and planning future actions



# Agents in user interfaces

- Agents learn
  - monitoring user's behavior (and they use behavioral patterns)
  - feedback from the user
  - question to other agents

- Example
  - e-mail filter
  - purchase of ticket



# Thank you for attention