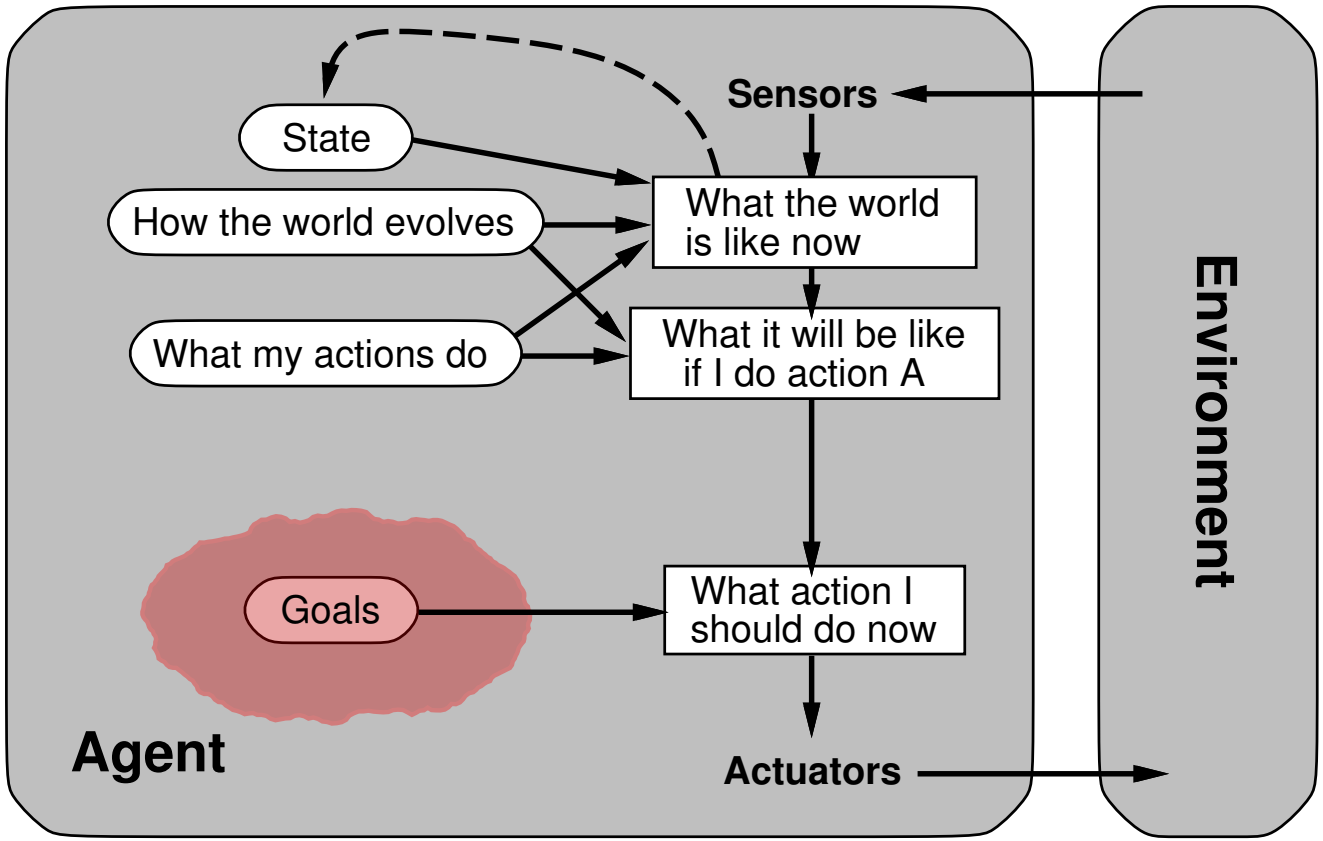
OPPA European Social Fund
Prague & EU: We invest in your future.

A4M33MAS/Lecture #4

Agent-oriented programming

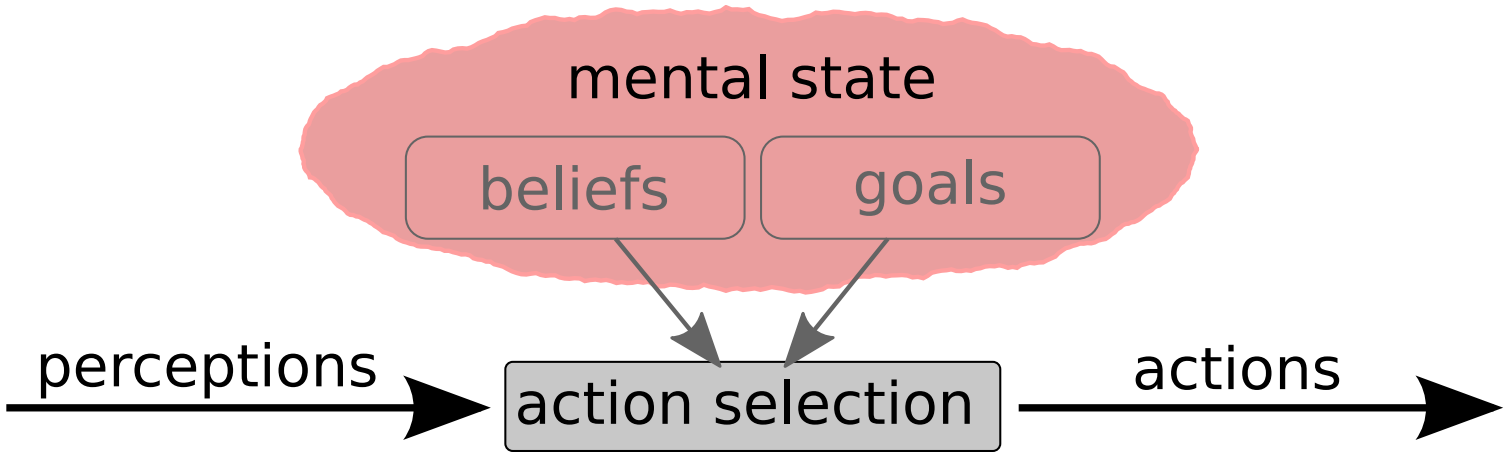
Goal-oriented agents



goals + state + actions' consequences $\rightsquigarrow$

action selection

Structure of cognitive agents



beliefs a database of agent's information about itself, the world (environment), other agents, etc.

⇝ NOW

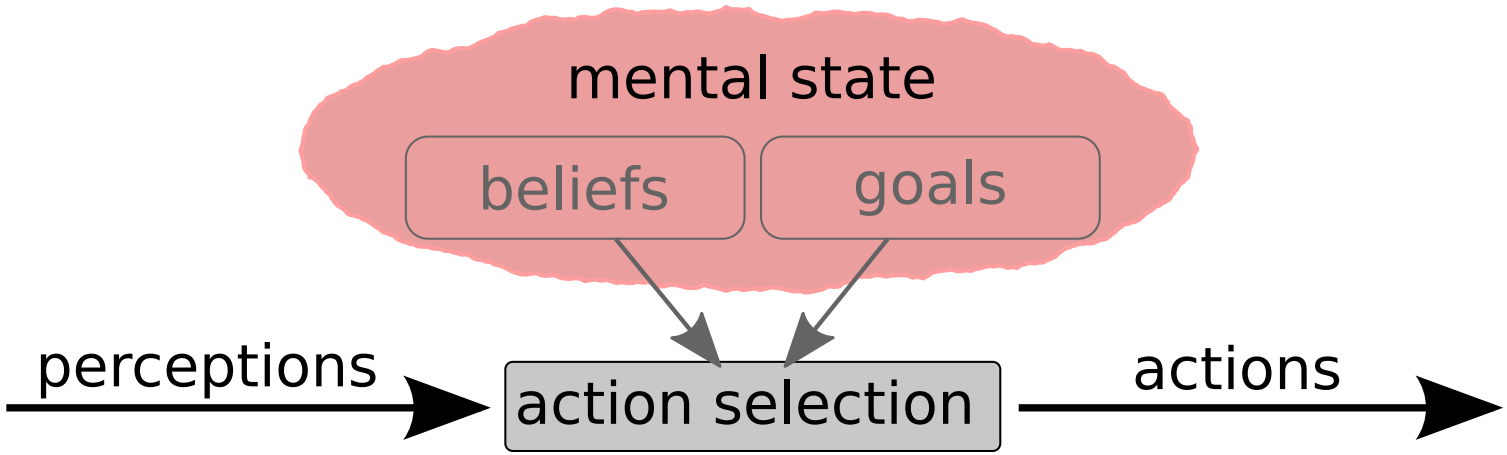
goals description of states the agent "wants" to bring about

⇝ FUTURE

How to select actions leading from NOW to the FUTURE ?

⇝ Planning!!!

Structure of cognitive agents



beliefs a database of agent's information about itself, the world (environment), other agents, etc.

⇝ NOW

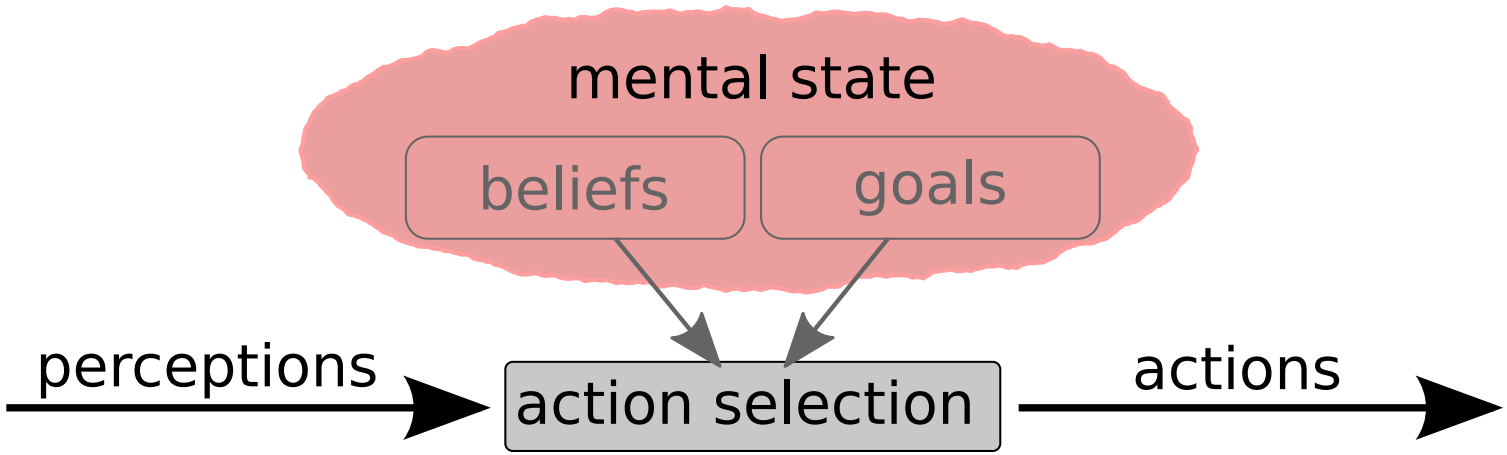
goals description of states the agent "wants" to bring about

⇝ FUTURE

How to select actions leading from NOW to the FUTURE ?

⇝ Planning!!!

Structure of cognitive agents



beliefs a database of agent's information about itself, the world (environment), other agents, etc.

⇝ NOW

goals description of states the agent "wants" to bring about

⇝ FUTURE

How to select actions leading from NOW to the FUTURE ?

⇝ **Planning!!!**

Planning

Definition (planning)

... is the process of generating (possibly partial) representations of future behavior prior to the use of such plans to constrain or control that behavior. The outcome is usually a set of actions, with temporal and other constraints on them, for execution by some agent or agents.

(The MIT Encyclopedia of the Cognitive Sciences)

plan - execute - monitor cycle

1. plan from the current state to a goal state(s)
2. sequentially execute actions from the plan
3. monitor success of action execution
   - in the case of action failure, (re-)plan again (goto 1)

The issue with planning

to arrive to a valid plan, in the worst case, the planner has to explore all the possible action sequences!!!

$\rightsquigarrow$ high computational complexity ($\approx$PSPACE) ☹

speed of planning vs. environment dynamics

planning $\overset{speed}{\succ}$ environment can perform relatively well

planning $\overset{speed}{\prec}$ environment can lead to fatal inefficiencies

$\rightsquigarrow$ the system “suffocates” in (re-)planning

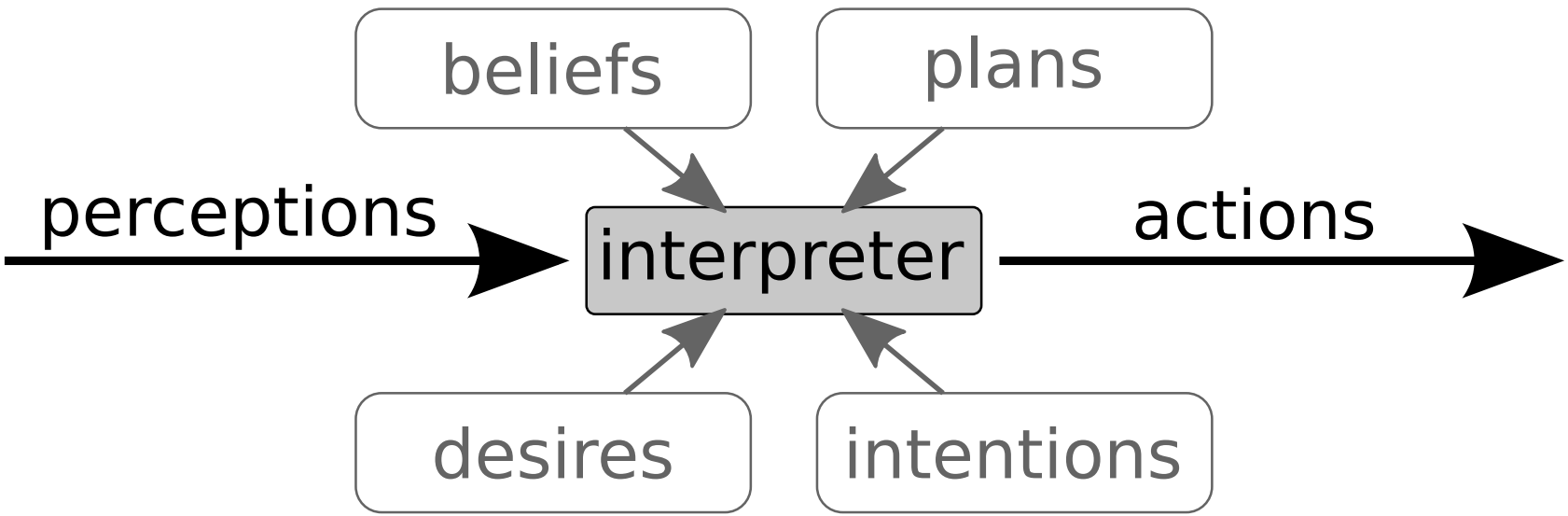
A way out: reactive planning & BDI

Structural decomposition:

- ***(B)eliefs:*** agent's static information about the world
- ***(D)esires:*** situations the agent wants to bring about
- ***(I)ntentions:*** courses of action, plans

System dynamics:

- ***reactive planning***: instead of plan-execute-monitor cycle, select partial plans reactively on the ground of the current state of the world, beliefs and goals



Agent-oriented programming

Agent-oriented programming

Promotes programming with mentalistic notions and intentional stance as an abstraction. Provides a realization of the BDI agent architecture in pragmatic programming languages.

AOP system:

1. a logical system for *mental states*
2. an interpreted *programming language*
3. an *'agentification'* process

Agent-oriented programming

Agent-oriented programming

Promotes programming with mentalistic notions and intentional stance as an abstraction. Provides a realization of the BDI agent architecture in pragmatic programming languages.

AOP system:

1. a logical system for *mental states*
2. **an interpreted *programming language***
3. an *'agentification'* process

What can APLs do for us?

1. mentalistic abstractions for agent system specification
   - beliefs, desires, intentions, plans, practical reasoning rules, etc.,
   - operationalization of the BDI architecture
   - *tools for encoding the system dynamics*
2. agent-oriented language semantics
   - syntax & model of execution
   - loosely corresponds to temporal modal logics
3. means to tackle the pro-activity vs. reactivity problem
   - deliberation/planning vs. handling events & interruptions $\rightsquigarrow$ hybrid architectures

Historical overview

Hybrid architectures: *– incomplete –*

1987: PRS (Georgeff and Lansky)
1988: IRMA (Bratman, Israel and Pollack)
1991: Abstract BDI architecture (Rao and Georgeff)
1994: INTERRAP (Müller and Pischel)

Agent-Oriented Programming Languages: *– incomplete –*

1990: AGENT-0 (Shoham)
1996: AgentSpeak(L) (Rao)
1996: Golog (Reiter, Levesque, Lesperance)
1997: 3APL (Hindriks et al.)
1998: ConGolog (Giacomo, Levesque, Lesperance)
2000: JACK (Busetta et al.)
2000: GOAL (Hindriks et al.)
2002: Jason (Bordini, Hubner)
2003: Jadex (Braubach, Pokahr et al.)
2008: BSM/Jazzyk (Novák)
2008: 2APL (Dastani)

BDI: the underlying principles

Structure of agent's internal state

- beliefs $\rightsquigarrow \mathcal{B}$
- goals $\rightsquigarrow \mathcal{G}$
- intentions/plans $\rightsquigarrow \mathcal{I}$ (optional)
- \+ an interface to the environment $\rightsquigarrow \mathcal{E}$

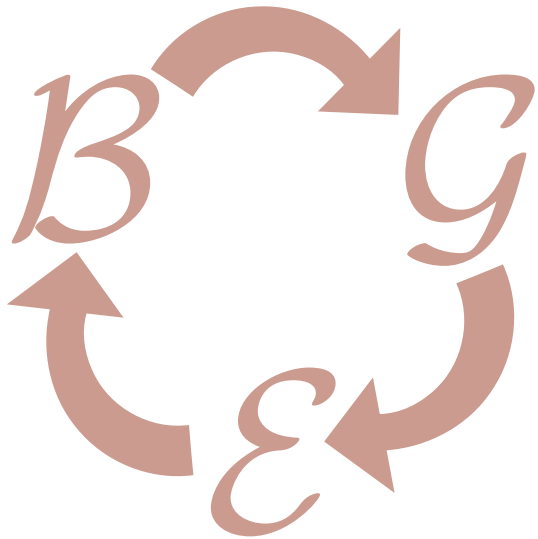
Minimal flow of information

1. agent perceives the environment and reflects it in the belief base
2. its beliefs about the world determine the goals it pursues
3. pursuing goals triggers behaviors aimed at fulfilling them

BDI: the underlying principles

Structure of agent's internal state

- beliefs $\rightsquigarrow \mathcal{B}$
- goals $\rightsquigarrow \mathcal{G}$
- intentions/plans $\rightsquigarrow \mathcal{I}$ (optional)
- \+ an interface to the environment $\rightsquigarrow \mathcal{E}$



Minimal flow of information

1. agent perceives the environment and reflects it in the belief base
2. its beliefs about the world determine the goals it pursues
3. pursuing goals triggers behaviors aimed at fulfilling them

OPPA European Social Fund
Prague & EU: We invest in your future.