



# Verteilte Künstliche Intelligenz – Formale Grundlagen

VAK 03-710.03

Universität Bremen

Ingo J. Timm, Jan D. Gehrke, Thorsten Scholz



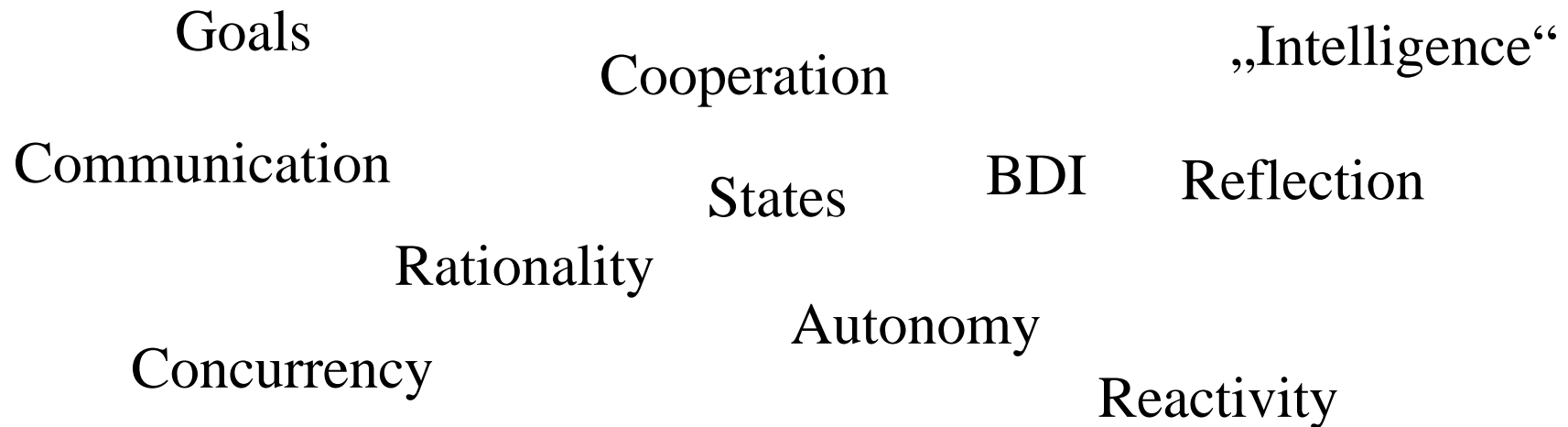
# Gliederung

- **Agentenarchitekturen**
  - Überblick
  - Personal Information Agents
  - Beliefs-Desires-Intention
- **Logik der Diskursagenten**
  - Sorten
  - Modalitäten
  - Operatoren
  - Alphabet
  - Grammatik



# Introduction to Intelligent Agents

*Agents are software entities that **assist** people and act on their behalf. (P. Maes, 1994)*



*Agents are situated in an environment, act autonomously, and are able to **sense** and to **react** to changes. (Knirsch & Timm, 1999)*



# Introduction to Intelligent Agents

*Agents are software entities that **assist** people and act on their behalf. (P. Maes, 1994)*

## Weak Definition of Agency:

- Reactivity
- Deliberativity
- Social Deliberation

[Jennings & Wooldridge, 1995]

*Agents are situated in an environment, act autonomously, and are able to **sense** and to **react** to changes.* (Knirsch & Timm, 1999)



# Is it an Agent or an Object ?

- Agents have an architecture
- Agents have „mental“ states
- Agents are goal-directed
- Agents are plan-based
- Agents have dialogs
- Objects have not
- Objects have attributes
- Objects are message-triggered
- Objects are „life-cycle bounded“
- Objects have send/receive behaviour

(As reported by Burmeister 1996)



# Requirements for MAS-Applications

Agent technologies are indicated if the following features can be fixed in the application. The application should

- show ***natural distributivity (ND)***,  
e.g. autonomous entities, geographical distribution,  
distributed data
- have a need for ***flexible interaction (FI)***,  
e.g. there is no a priori assignment of tasks to actors, there  
are no fixed processes
- be embedded in a ***dynamic environment (DE)***  
e.g. our physical world or artificial worlds like the internet.

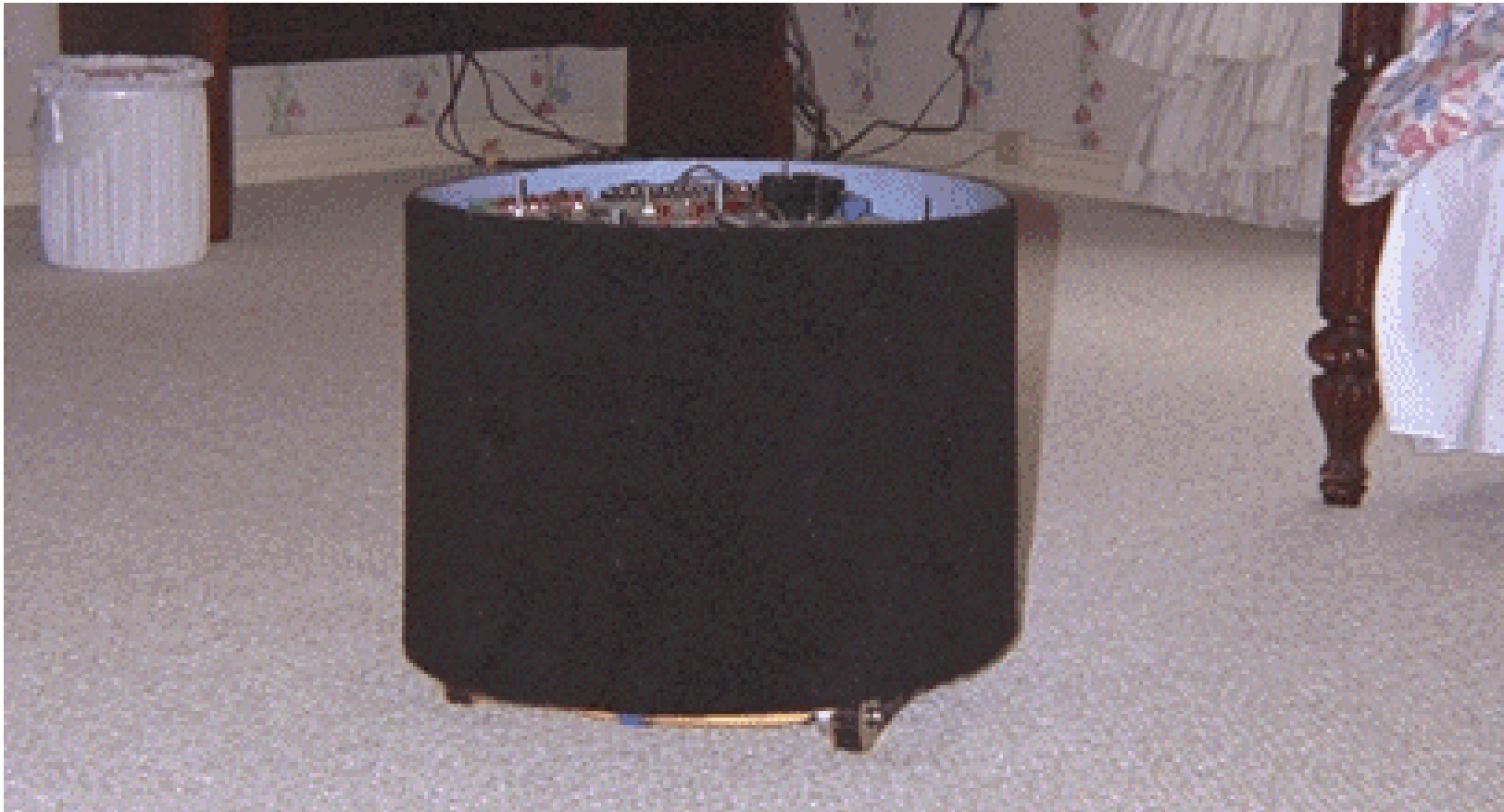
*Applications which do not match all three criteria should not be approached with the agent technology!*



# Application of Autonomous Agents

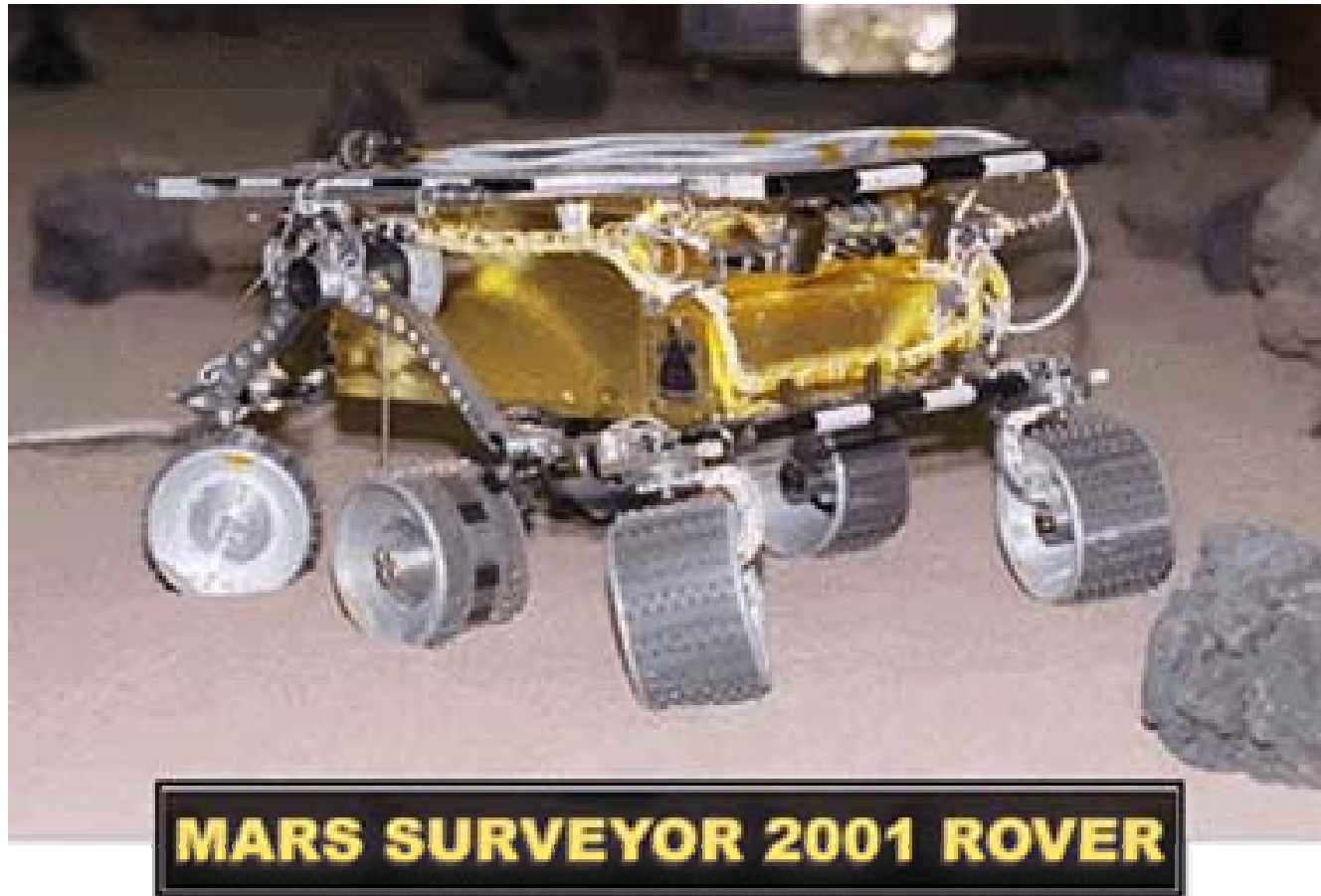
*Sun Jini – Agent Technology for the Microwave?*

# Application of Autonomous Agents





# Application of Autonomous Agents





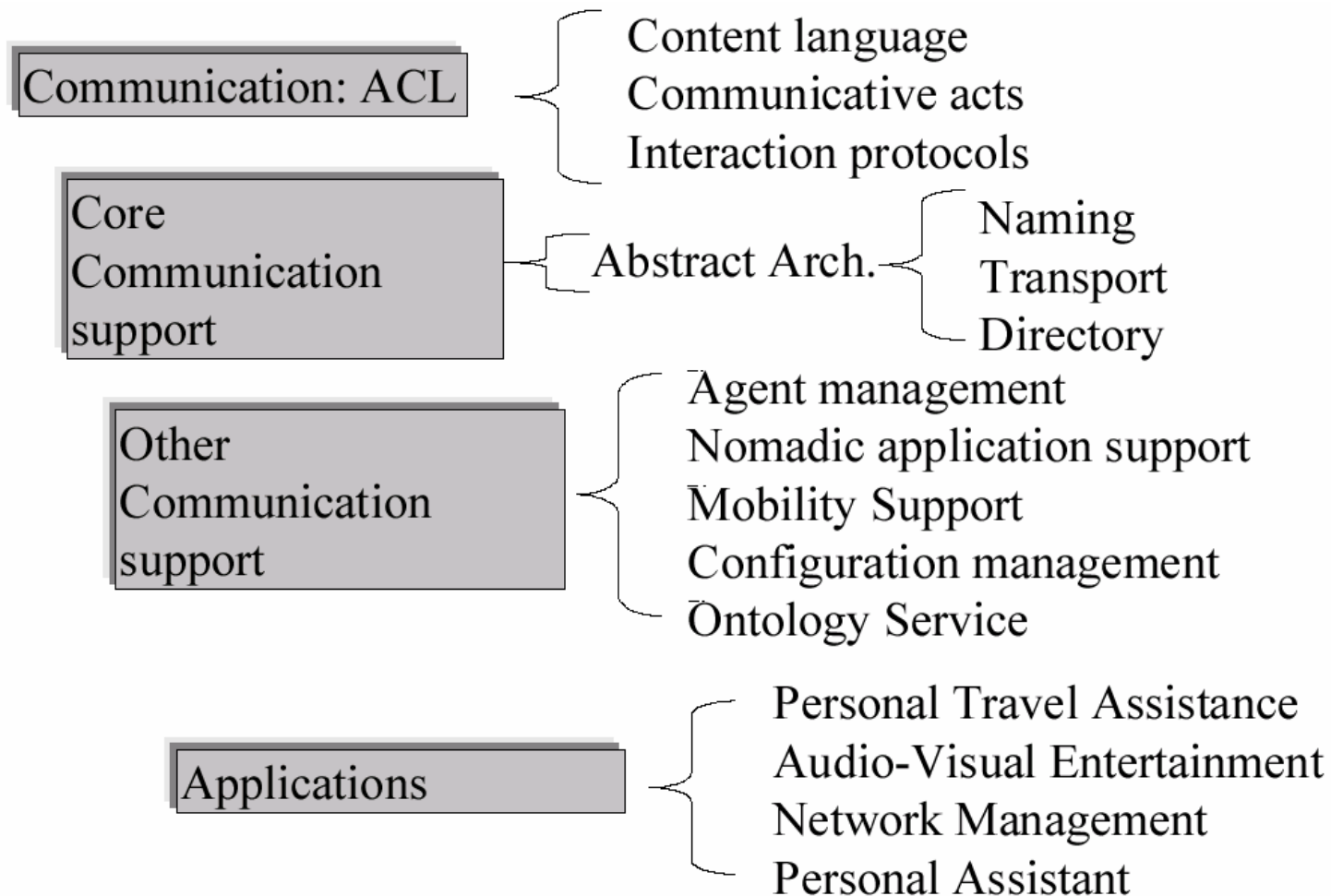
# FIPA: Foundation for Intelligent Physical Agents

- FIPA was formed in 1996 and is dedicated to:

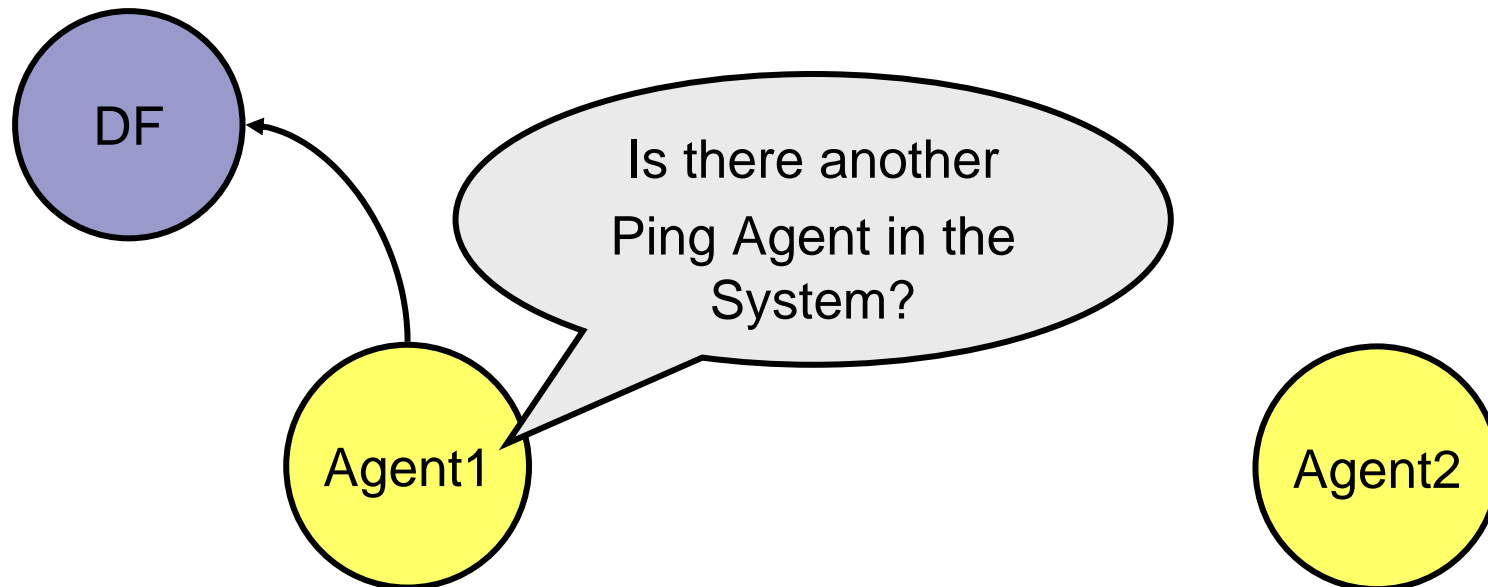
*The promotion of technologies and interoperability specifications that facilitate the end-to-end inter-working of intelligent agent systems in modern commercial and industrial settings*

- Important specifications: abstract architecture and
- Agent communication language (FIPA-ACL)
- 5 open source implementations (FIPA-OS, ZEUS, JADE)

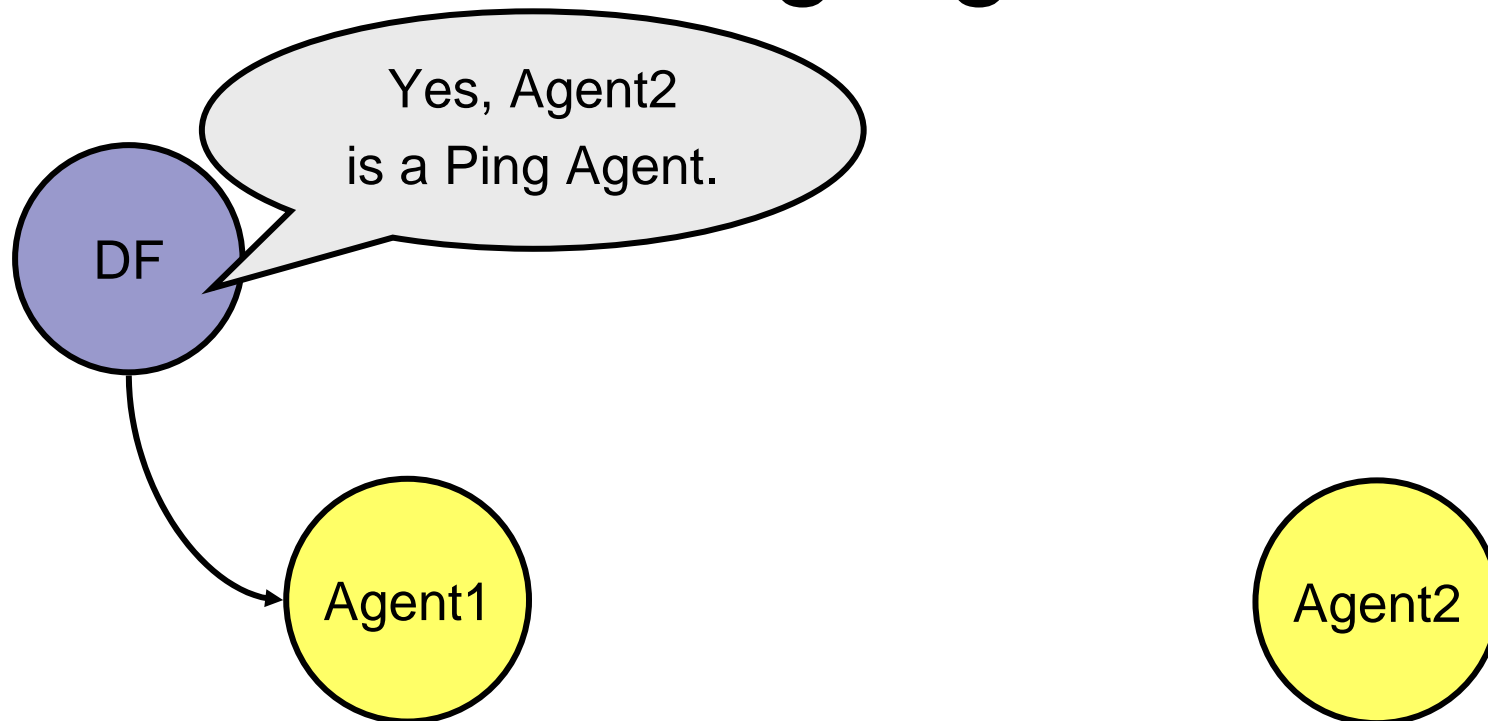
# FIPA Overview

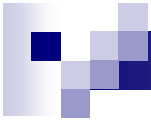


# Hello World: Ping Agents

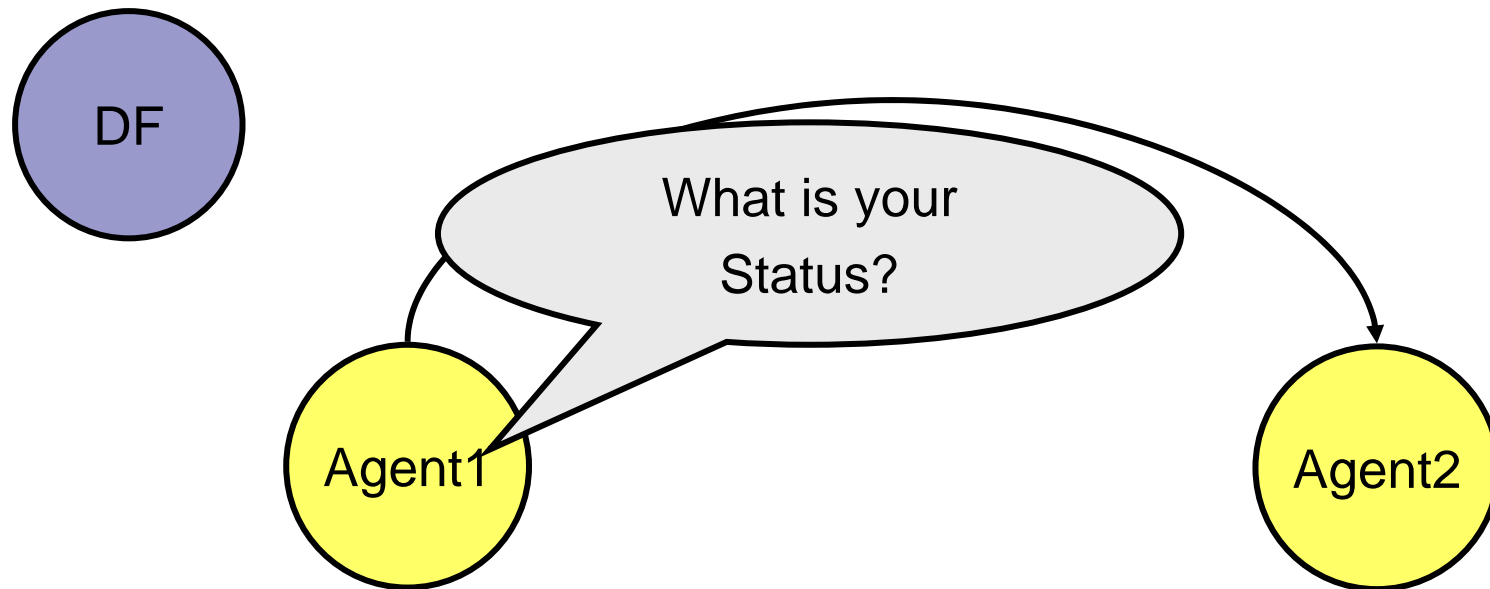


# Hello World: Ping Agents



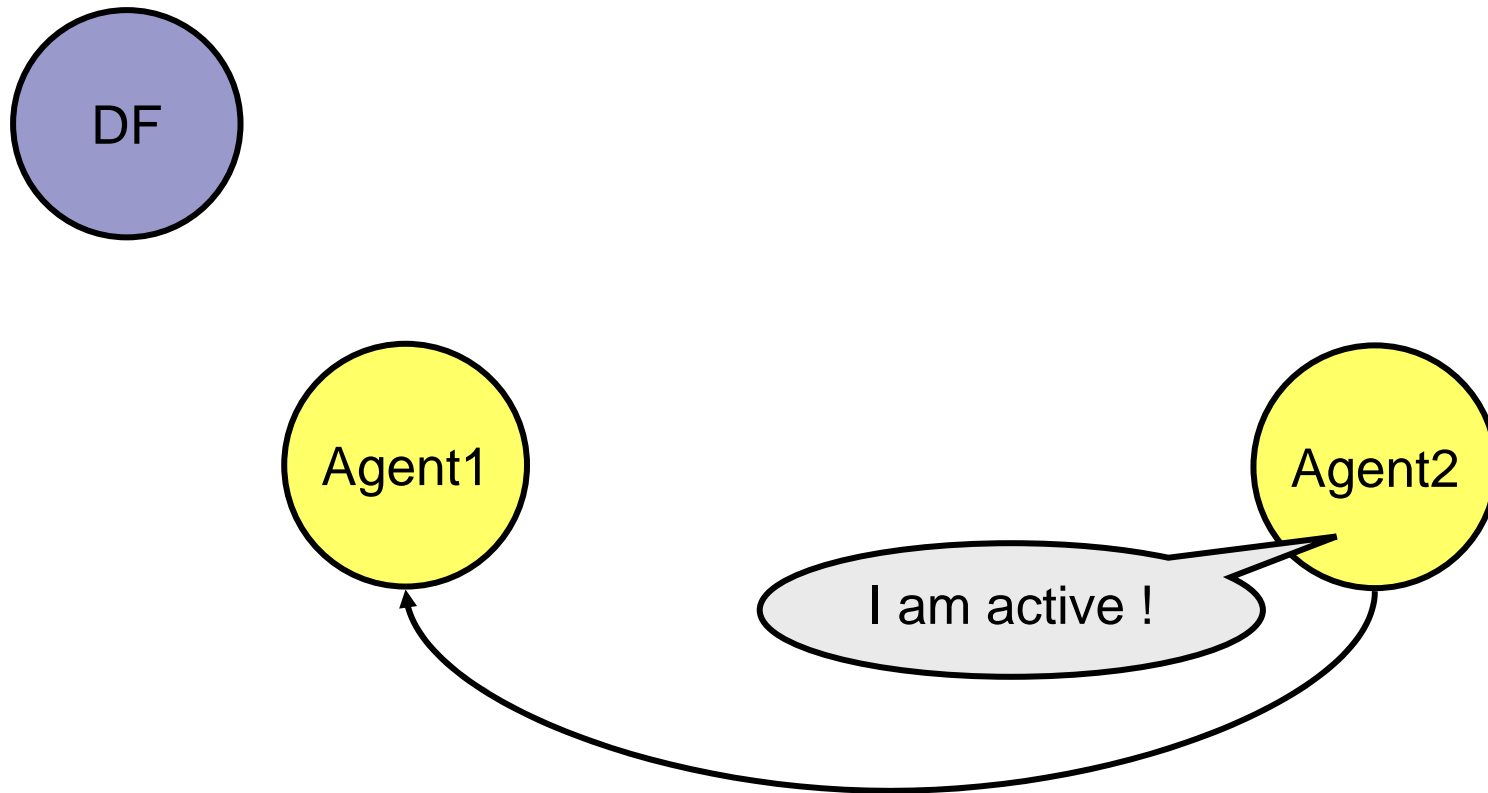


# Hello World: Ping Agents

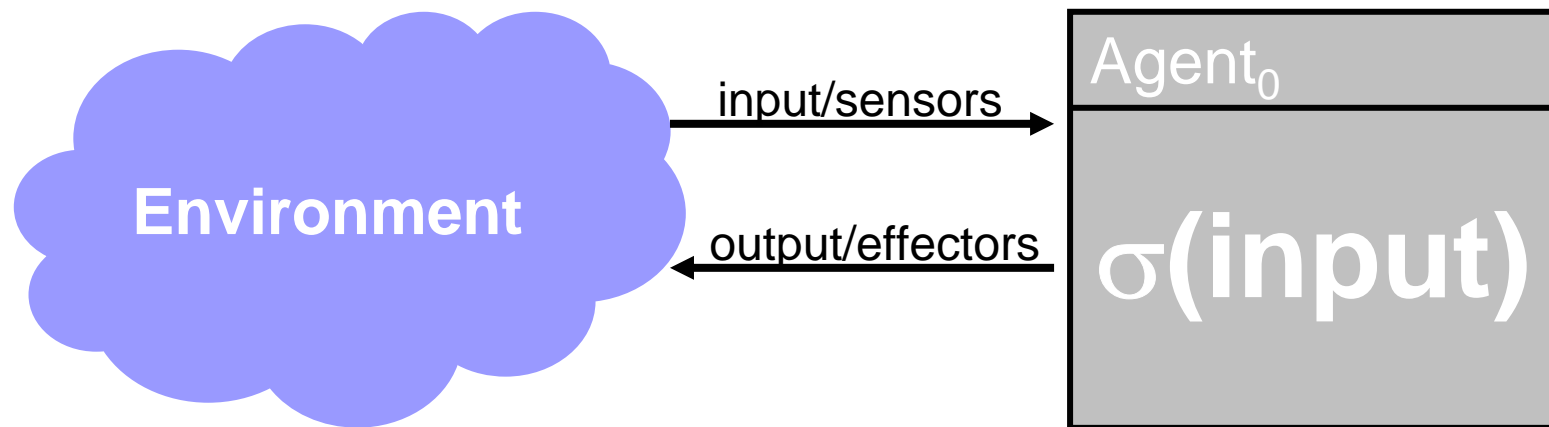




# Hello World: Ping Agents

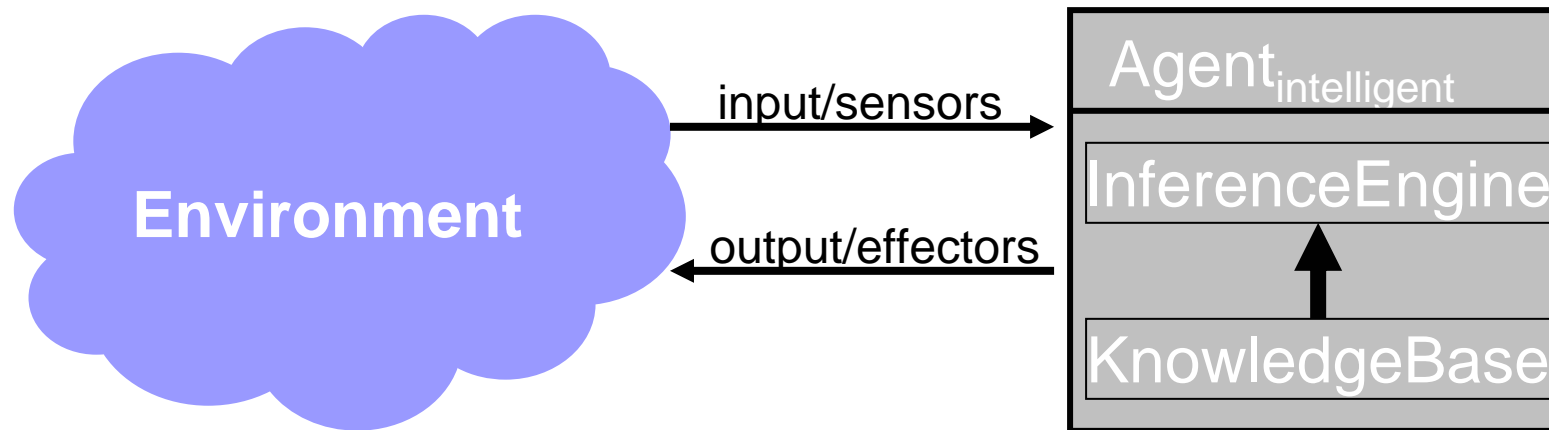


# Definition of an Agent

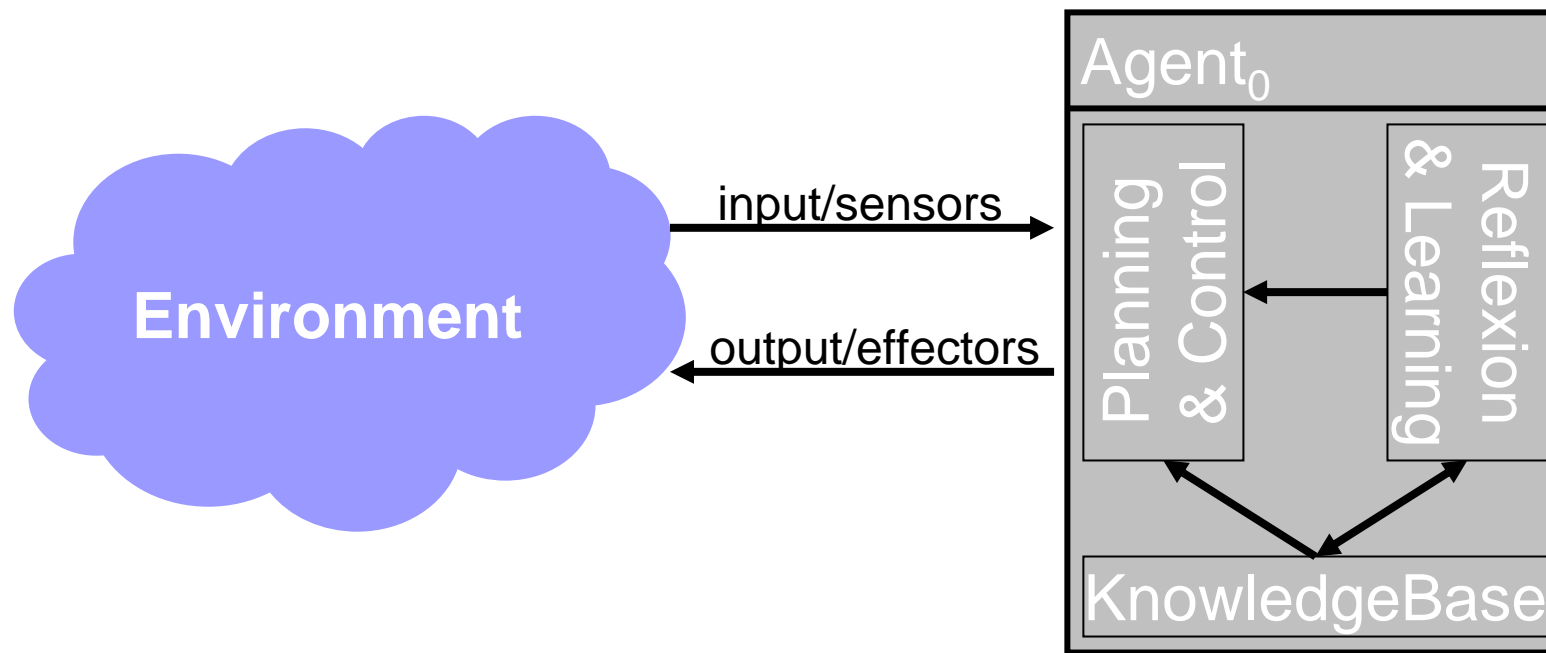


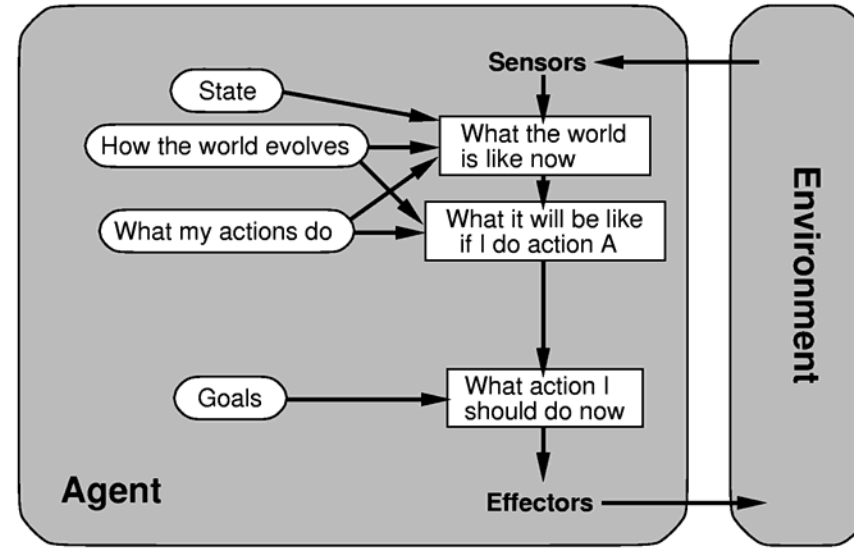
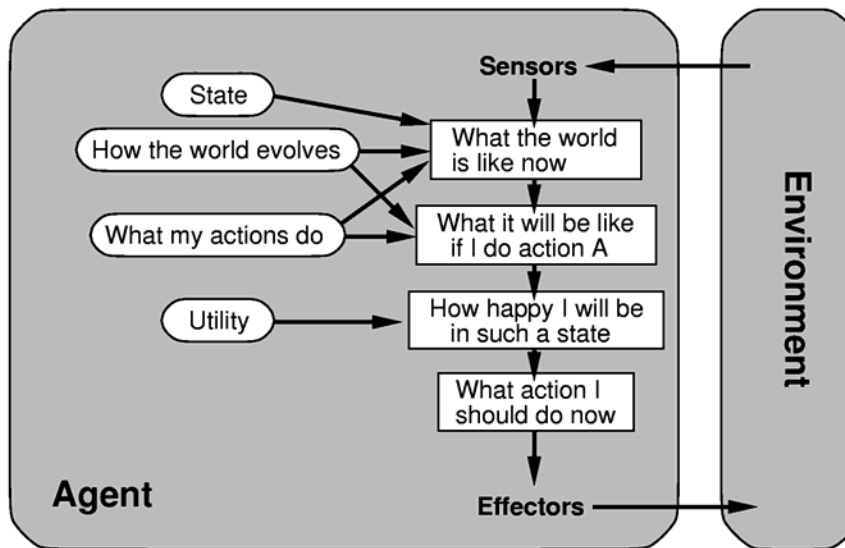
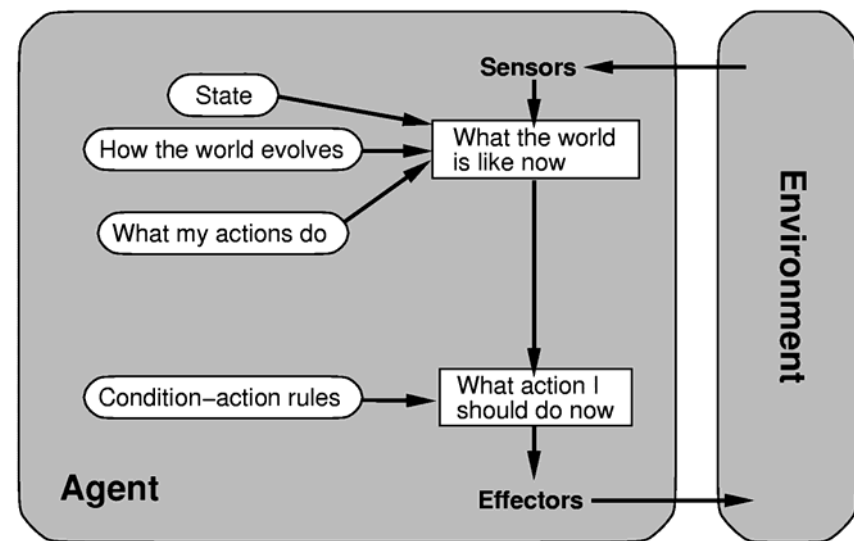
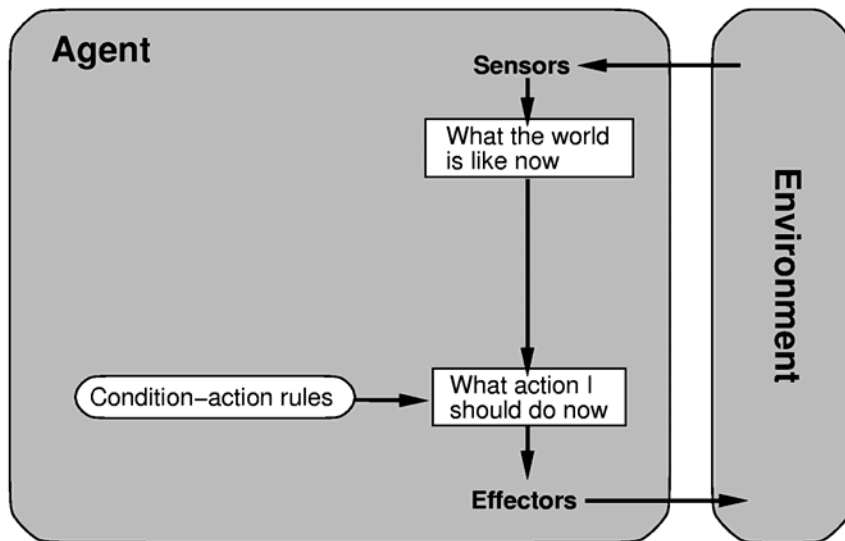


# Definition of an Intelligent Agent

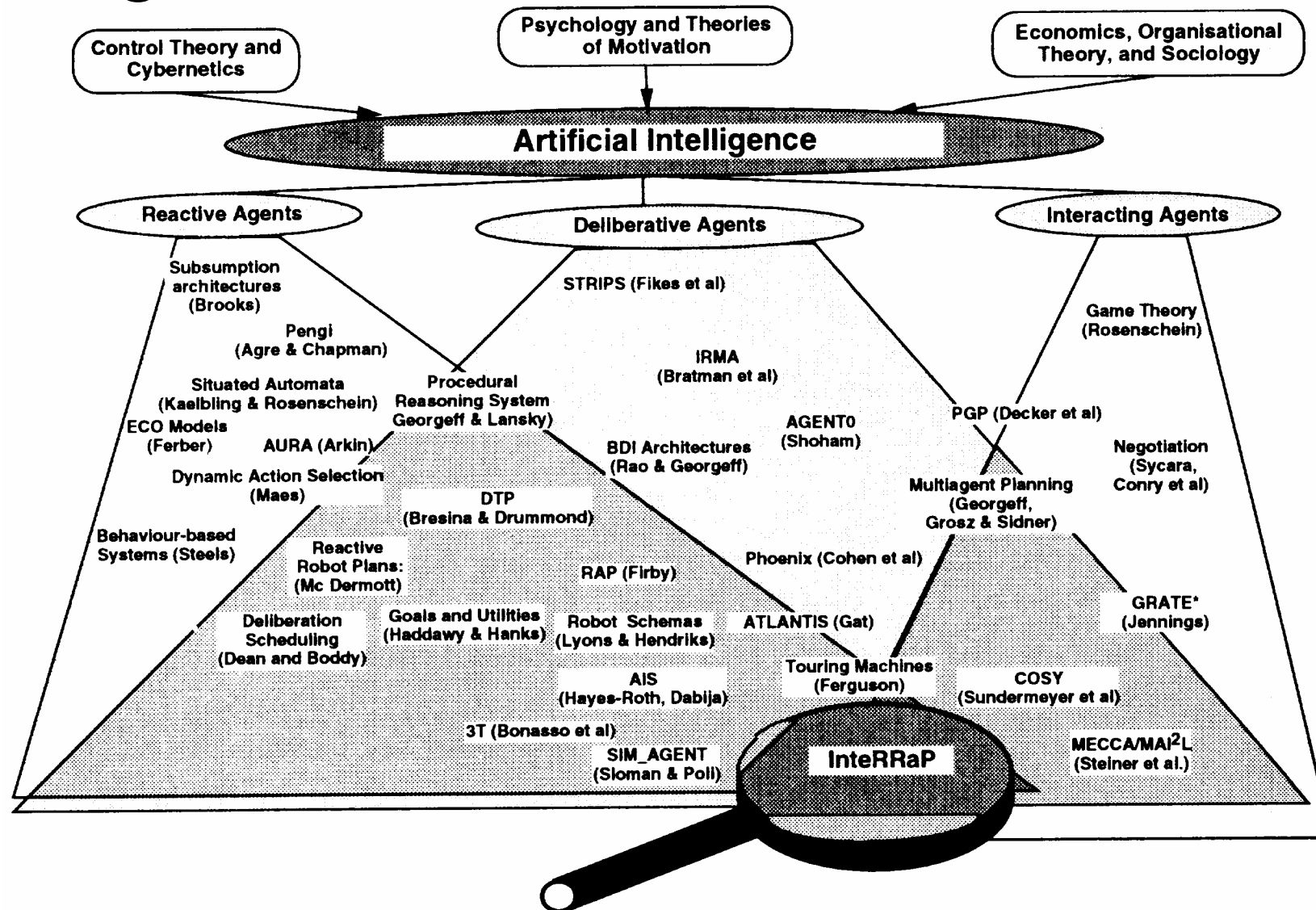


# Definition of an Intelligent Learning Agent





# Agent Architectures



# FIRST REISEBÜRO

Wo Ihr Urlaub beginnt.



Urlaubsreisen



Last Minute & More



## Fluganfrage

Bitte geben Sie Ihre Flugdaten ein ...

Flugdaten		
Abflughafen:	Abflugdatum:	Zeit:
Frankfurt	09.03.2002	ab 6 Uhr
Zielflughafen:	Rückflugdatum:	Zeit:
Paris	10.03.2002	ab 6 Uhr
<input checked="" type="radio"/> Hin- und Rückflug		
<input type="radio"/> Einfacher Flug (kein Rückflug)		
Fluggesellschaft:	Flexibilität:	Klasse:
Air France	+/- 0 Tage	Economy
<a href="#">Anfrage starten</a>		

the user  
specifies  
preferences

some  
preferences  
conflict

information  
for resolving  
conflicts



# Was fällt an dieser Situation auf?

- Reihenfolge Präferenzen
  - Einige der gewählten Merkmale sind wichtig, andere sind „nice to have“
  - Teilordnung der Präferenzen
- Konflikt Präferenzen
  - Benutzer können nicht erkennen, welche Präferenzen in Konflikt stehen
- Individuelle Beurteilung der Ergebnisse
  - Unterschiedliche Nutzer präferieren unterschiedliche Konfliktlösungen
- Welche Lösung sollte bevorzugt werden?

Departure: 9.3.2003  
Flexibility: +/- 1 day  
Airline: Air France

Flight 1: Air France 11.3.  
Flight 2: Lufthansa 9.3.



# Persönliche Assistenten

## ■ Anpassbarkeit

- ☐ Repräsentation von Wahl, Präferenz und Konfliktlösungsstrategie

## ■ Deliberation

- ☐ Auflösung der Konflikte auf eine Art und Weise, die für den Benutzer transparent ist
- ☐ Balanzierung von zu häufigen Rückfragen während der Konfliktlösung und zu schlechten Ergebnissen in der Konfliktlösung

## ■ Benutzerinteraktion

- ☐ Reaktives gegen pro-aktives Verhalten
- ☐ Pull versus Push bei der Präsentation von Informationen

## ■ Verhandlung

- ☐ Verhandlung im Auftrag des Nutzers mit anderen Agenten

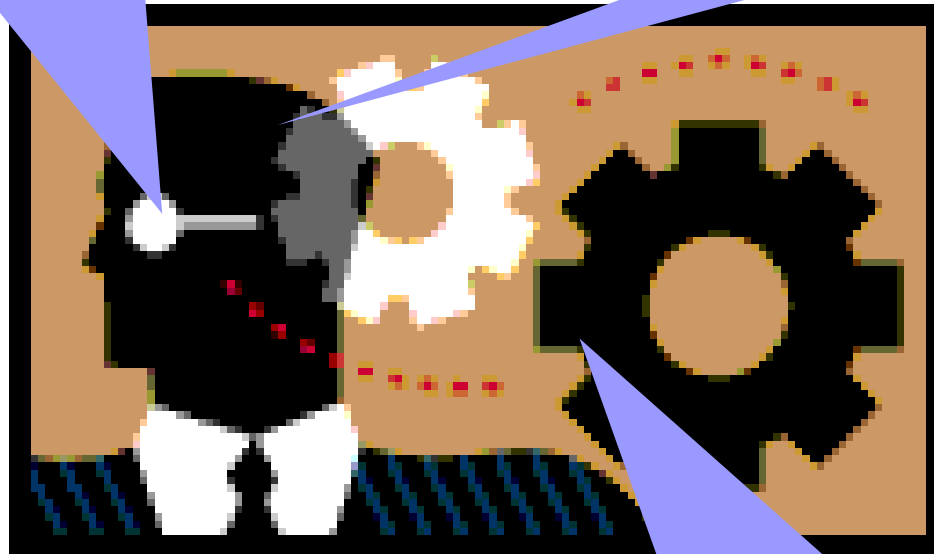
# Agenten mit BDI

## Beliefs

The plane leaves at 1 a.m.

## Desire

Get on the next plane



## Intention

Leave office at 11 p.m. Get taxi to airport





# Practical Reasoning

- Proactive behavior
  - Agent exhibits goal-directed behavior by taking initiative
- Deliberative architecture
  - Goal-directed behavior by means-end reasoning
- Reasoning with goals
  - Finding means to achieve ones own goals
  - Predicting the bahvior of others in terms of their goals
- Example
  - Lisa: The meeting is at 10.00. I will have to leave at 9.30.
  - Peter: Lisa does not know that the meeting was cancelled. She will arrive at 9.50.



# Aspekte von BDI-Architekturen

- Die BDI-Architektur integriert drei wichtige Eigenschaften [Woo00, S. 7]:
  1. Sie basiert auf einer bekannten und breit akzeptierten Theorie rationaler Handlungen von Menschen
  2. Sie ist in einer großen Zahl komplexer Anwendungen erfolgreich.
  3. Sie kann auf eine hierfür entwickelte Familie extensiv formalisierter Logiken zurückgreifen.



# Kontrollzyklus eines BDI Interpreters

```
1. while (true) do  
  
2.   observe the world;  
3.   update internal world model;  
4.   deliberate about what intention to achieve next;  
5.   use means-ends reasoning to get a plan for the  
    intention;  
6.   execute the plan;  
  
7. end while
```



## Verfeinerter Kontrollzyklus eines BDI Interpreters

```
1. beliefs := beliefs'
   // beliefs' sind initiale Überzeugungen

2. while (true) do

3.   get next perception p1;
   // p1 ist eine Wahrnehmung
4.   beliefs := belief-revision-function(beliefs, p1);
   // beliefs sind die Überzeugungen des Agenten
5.   intentions := deliberate(beliefs);
   // intentions sind die Absichten des Agenten
6.   plan := planning(beliefs, intentions);
   // plan ist eine Aktionssequenz
7.   execute(plan);

8. end while
```

# Ablauf in Belief-Desire-Intention Architekturen

