

## Reaktive Agenten

- Motivation: Klassische (planende) Agenten sind...
  - langsam
  - teuer in jeder Hinsicht
  - labil (insb. in dynamischen Umgebungen)
  - vollkommen auf ihre Wissensbasis angewiesen
  - können nur ein Ziel verfolgen
- Alternativer Ansatz:
  - Keine symbolische Darstellung des Weltwissens und keine expliziten Schlussfolgerungsprozesse
  - Stattdessen Menge von „Situation-action“-Regeln
  - Ständige Interaktion mit der Umwelt durch direkte Reaktion auf Sensorinput
  - Biologisch motiviert durch einfache Reiz-Reaktions-Schemata („Die Welt ist ihr eigenes Modell“)

## Subsumption architecture (Rodney A. Brooks, 1985)

- Ziele:
  - Mobile, robuste, einfache Roboter
  - Sollen auch in dynamischen Umgebungen sicher agieren können
  - Leichte Erweiterbarkeit und Wartbarkeit
  - Mehrere Ziele (z.B.: „*Ich will zum Gold*“ vs. „*Ich muss das Loch umfahren*“)
- Aufgabenorientierte Schichten statt funktionale Einheiten!

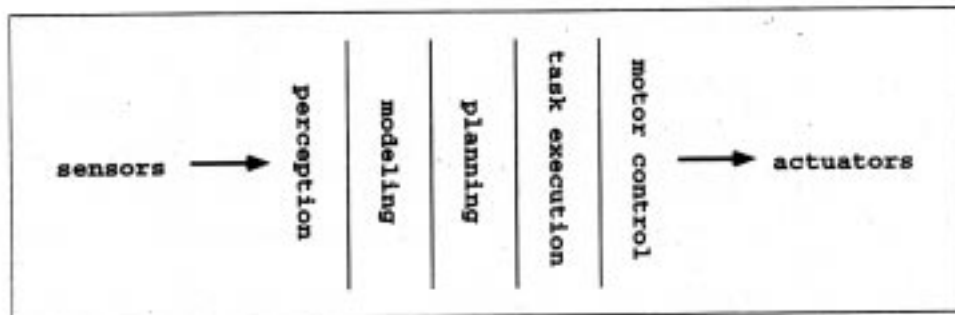


Abbildung 1: Funktionale Einheiten

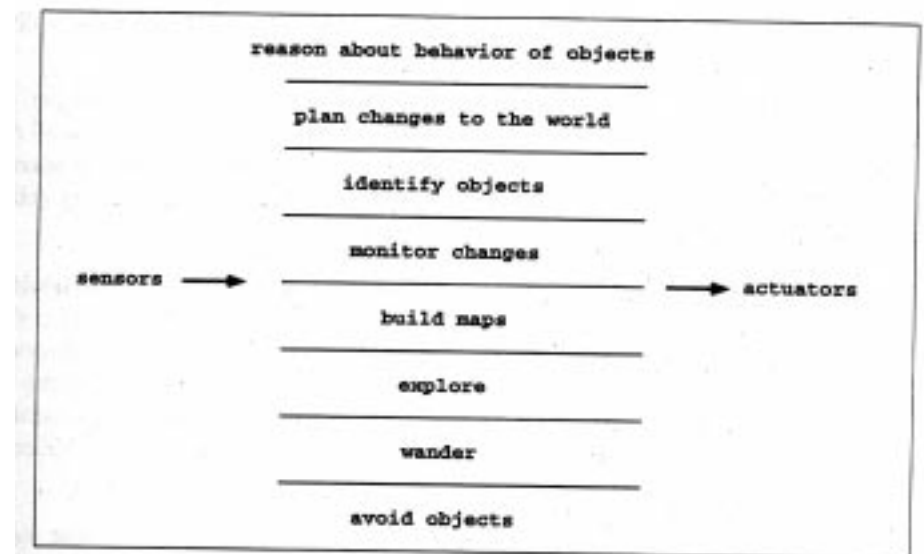


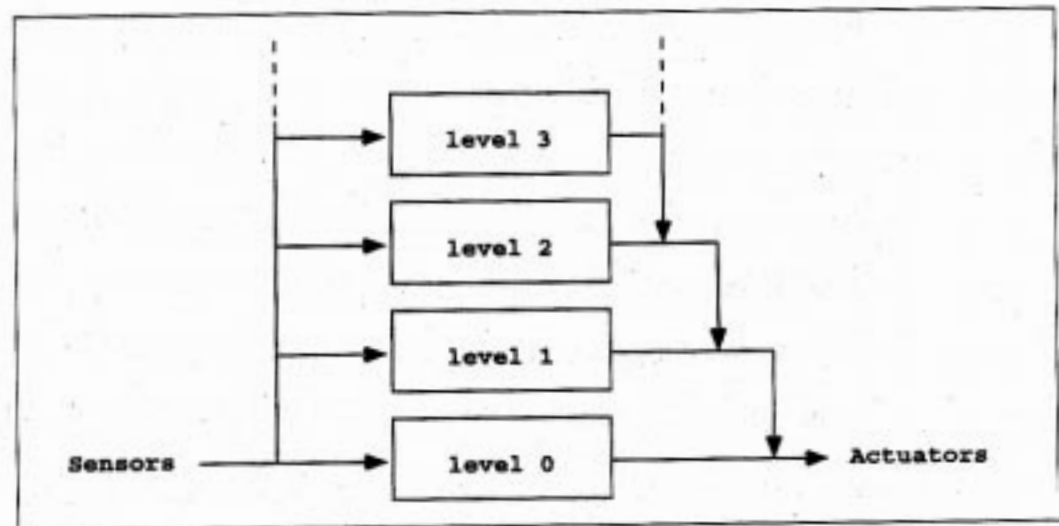
Abbildung 2: Aufgabenorientierte Schichten

## Kompetenzebenen

- Abstufung von unterschiedlich komplexen Verhaltensweisen (Kompetenzen)
- Acht Kompetenzebenen für mobilen Roboter (s.o.)
- Jede Ebene schliesst die Kompetenzen der darunter liegenden Schichten ein  
⇒ „*subsumption architecture*“

## Kontrollschichten

- Zu jeder Kompetenzebene eine Kompetenzschicht
- Schichten arbeiten parallel
- Jede Schicht kann jederzeit auf Sensoren und Aktuatoren zugreifen
- Höhere Schichten können die Ausgaben unterer Schichten unterdrücken



## In der Praxis...

- Damals nur bis Level 2 (Erkunden) implementiert
- Gute Verhaltensweisen für eine so einfache Architektur
- Wirklich einfach, ohne interne Repräsentation, leicht erweiterbar...?

