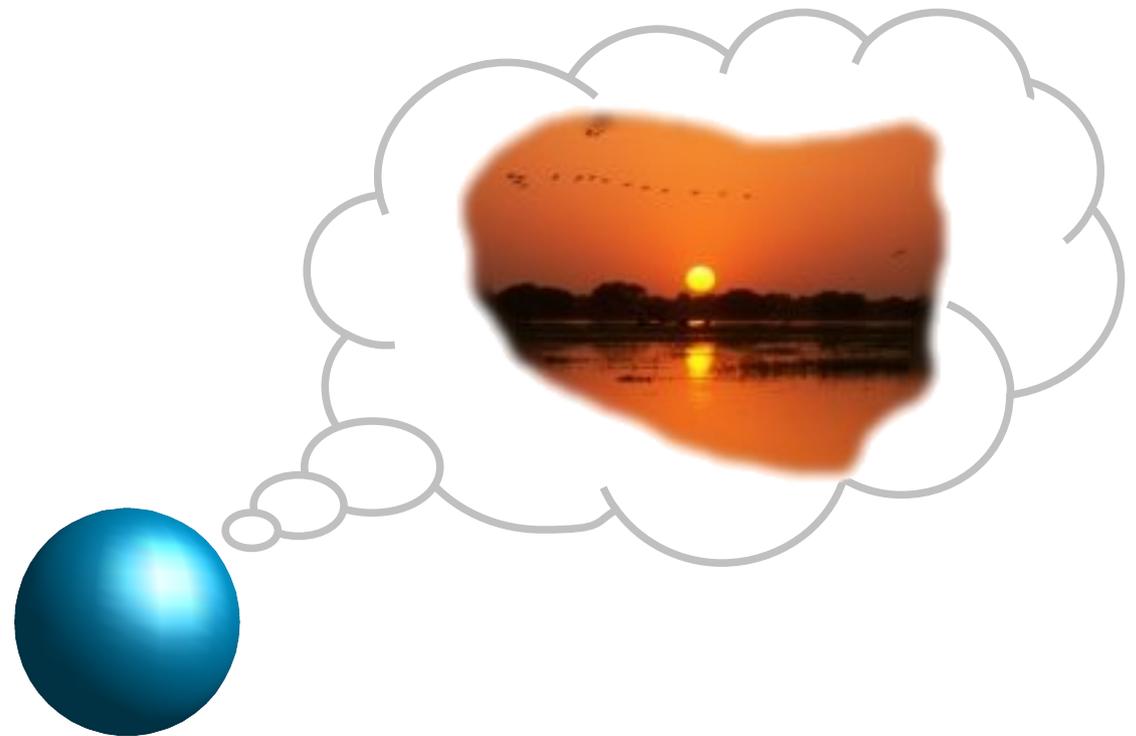
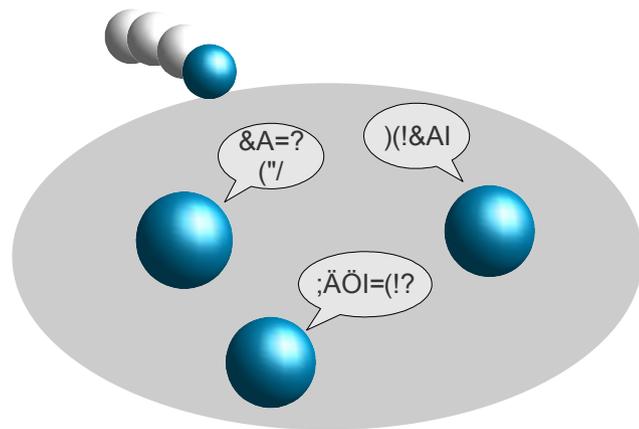


# Glaube, Wünsche, Absichten: Entwurf intelligenter Agenten



# Entwurf von Agenten

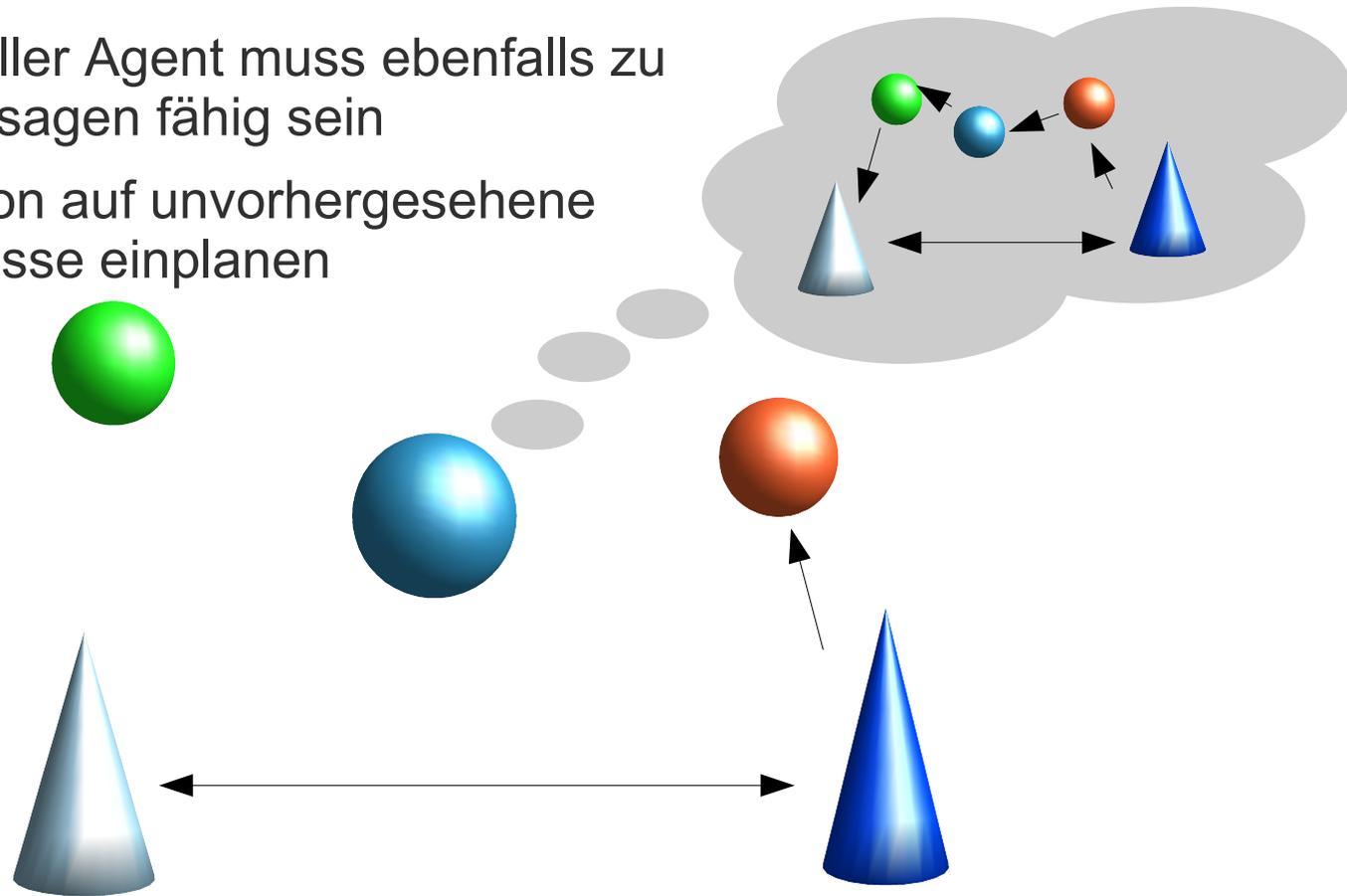
- Umsetzung der allgemeinen „Idee“ von Agenten
  - wie baut man daraus Software / Hardware?
  - neuartige Entwurfsmuster erforderlich
- Autonomiecharakter
  - Einfluss auf die Strukturierung
- Verfolgen eines „Plans“
  - wie geht das?
  - Wie macht das ein Mensch?

# Deliberatives Verhalten

- Erkenntnisse aus der Psychologie
- Der Mensch erzeugt sich ein Bild der Welt im Kopf
  - Simulation der Welt
  - Experimente
  - Versuch, das Verhalten der Welt / Teile davon vorherzusagen
- Plan
  - Kette von Aktionen, die sich als zweckdienlich in der Simulation erwiesen haben
  - Abklären von Alternativen auf dem Weg
- Durchführen eines Plans
  - Umsetzen der simulierten Aktionen
  - Abgleich der Reaktion der Welt mit der vorausgesagten Reaktion

# Das Bild der Welt

- Abbild der Welt
  - Planvoller Agent muss ebenfalls zu Vorhersagen fähig sein
  - Reaktion auf unvorhergesehene Ereignisse einplanen



# Das BDI-Modell

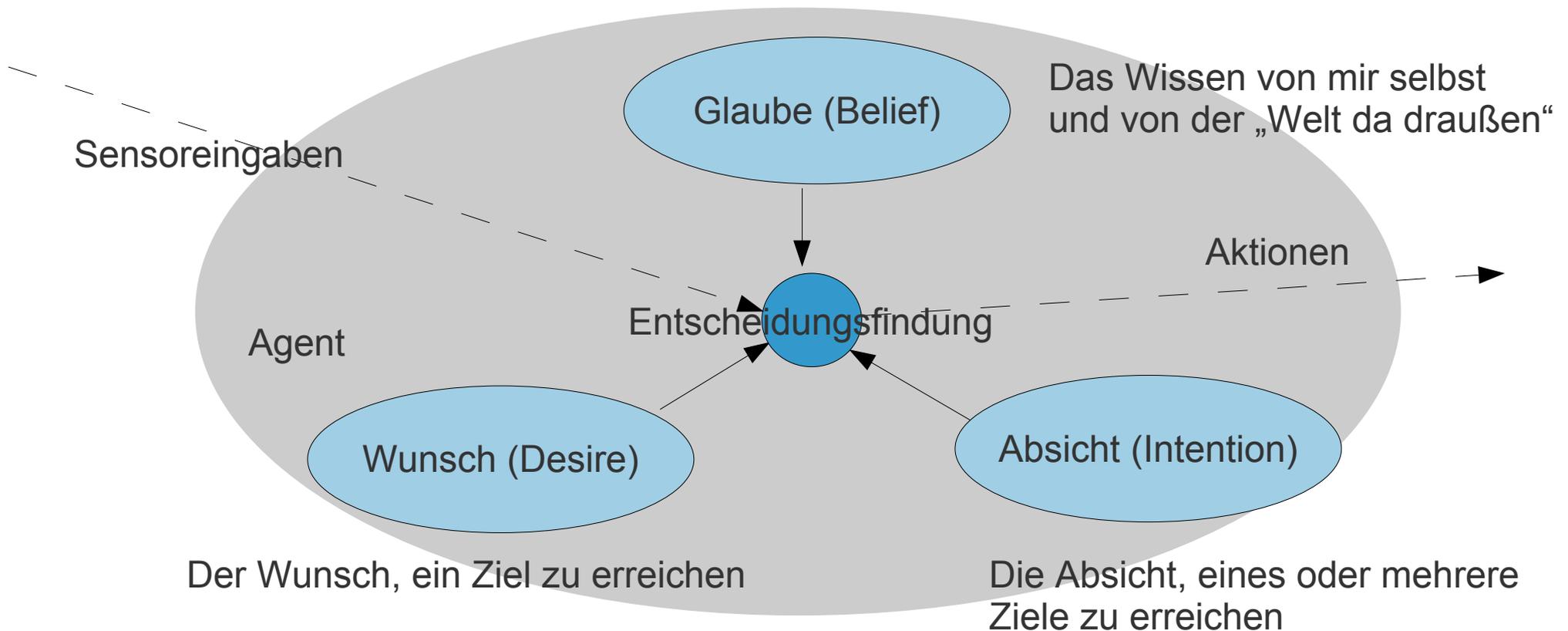
- Entwickelt von Michael Bratman (1987)
- Practical Reasoning (praktische Entscheidungsfindung)
  - Deliberation (Abwägung): Welche Ziele sind zu erreichen?
  - Means-End (Mittel-Zweck): Wie sind die Ziele zu erreichen?
- Zielt besonders auf die Rolle von Absichten (Intentionen) ab, die diese in der Entscheidungsfindung spielen

# BDI

- Bratman 1990: (practical reasoning)
  - *Die praktische Entscheidungsfindung betrifft die Abwägung einander gegenüberstehender Optionen, wobei sich die Optionen aus dem Wissen und den Wünschen des Agenten ergeben.*
- Die Entscheidungsfindung ist im Wesentlichen das Produkt aus den Faktoren Glaube, Wünsche, Absichten
- *Praktische* Entscheidungsfindung richtet sich auf Aktionen
- *Theoretische* Entscheidungsfindung richtet sich auf das Wissen
  - auch deduktive Entscheidungsfindung genannt
  - Beweisen von Aussagen, mathematische Basis

# Das BDI-Modell

- Analogie zu menschlichen kognitiven Vorgängen



# Der Glaube (Beliefs)

- Glaube
  - das, was der Agent von der Welt weiß
  - „inneres Bild der Welt“
- Glaube ist als gespeicherte Fakten zu verstehen
  - kein Zweifel an der Gültigkeit (nicht wie im religiösen Sinne)
- Weitere Möglichkeit: Ungewissheit (Uncertainty)
  - Bei FIPA-Architektur/Kommunikation vorgesehen (siehe dort)
- Man beachte den feinen Unterschied (offene Welt!)
  - Ich glaube nicht, dass X gilt = Ich weiß nicht, ob X gilt.
  - Ich glaube, dass X nicht gilt = Ich weiß, dass X nicht gilt.

Was ist in der Datenbank gespeichert?

# Wünsche (Desires)

- Wünsche
  - Wünsche liegen Entscheidungen zu Grunde
  - Auswahl zu verfolgender Wünsche ist abhängig von Ereignissen
  - Der Agent prüft, ob Wünsche erfüllt wurden
  - Wünsche müssen mit den Fakten abgeglichen werden (Erfüllbarkeit)
- Wünsche als Ziel (Goal)
  - Absichten können scheitern; das Ziel wird weiterhin verfolgt

# Absichten (Intentions)

- Absichten
  - Agent „beschließt“ Verfolgung eines Wunsches: Wunsch wird zur Absicht
  - Absicht sollte der Erfüllung eines Planes/Auftrags dienen
- Verknüpft mit dem Wissen
  - Eine Absicht verfolgen und zugleich glauben, dass man sie nicht erfüllen kann, ist *irrational*.
  - Eine Absicht verfolgen, ohne zu glauben, dass die Absicht erfüllt wird, ist *akzeptabel* als Eigenschaft *rationalen* Denkens.

# Wirkung von Absichten

- Absichten beeinflussen weitere Erwägungen
  - Verhindern von Konflikten: Aussortieren von Optionen, die der Verfolgung der Absicht widersprechen
  - Absichten sind dauerhaft – so lange,
    - bis ihre Wirkung erreicht ist
    - bis sie nicht mehr realisierbar sind
    - bis ihre Gründe nicht mehr zutreffen
  - Zukünftige Planung: Ich gehe davon aus, dass ich das Ziel mit meiner Absicht erreiche und kann spätere Aktionen darauf ausrichten
    - Würde ich wissen, dass ich das Ziel nicht erreiche, wäre dies irrational.

# Pläne

- Bedingungen
  - Vorbedingung
    - Was muss gegeben sein, dass dieser Plan in Angriff genommen wird?
  - Nachbedingung
    - Was gilt, wenn dieser Plan erfolgreich umgesetzt wurde?
- Prüfung der Vorbedingung gegenüber dem Wissen
  - Unerfüllte Vorbedingungen erzeugen Absichten
- Jeder Plan wird in Schritten vollzogen, die das Wissen beeinflussen
  - dadurch können Wünsche wieder unerfüllt werden
  - dadurch können wieder Absichten entstehen

# BDI

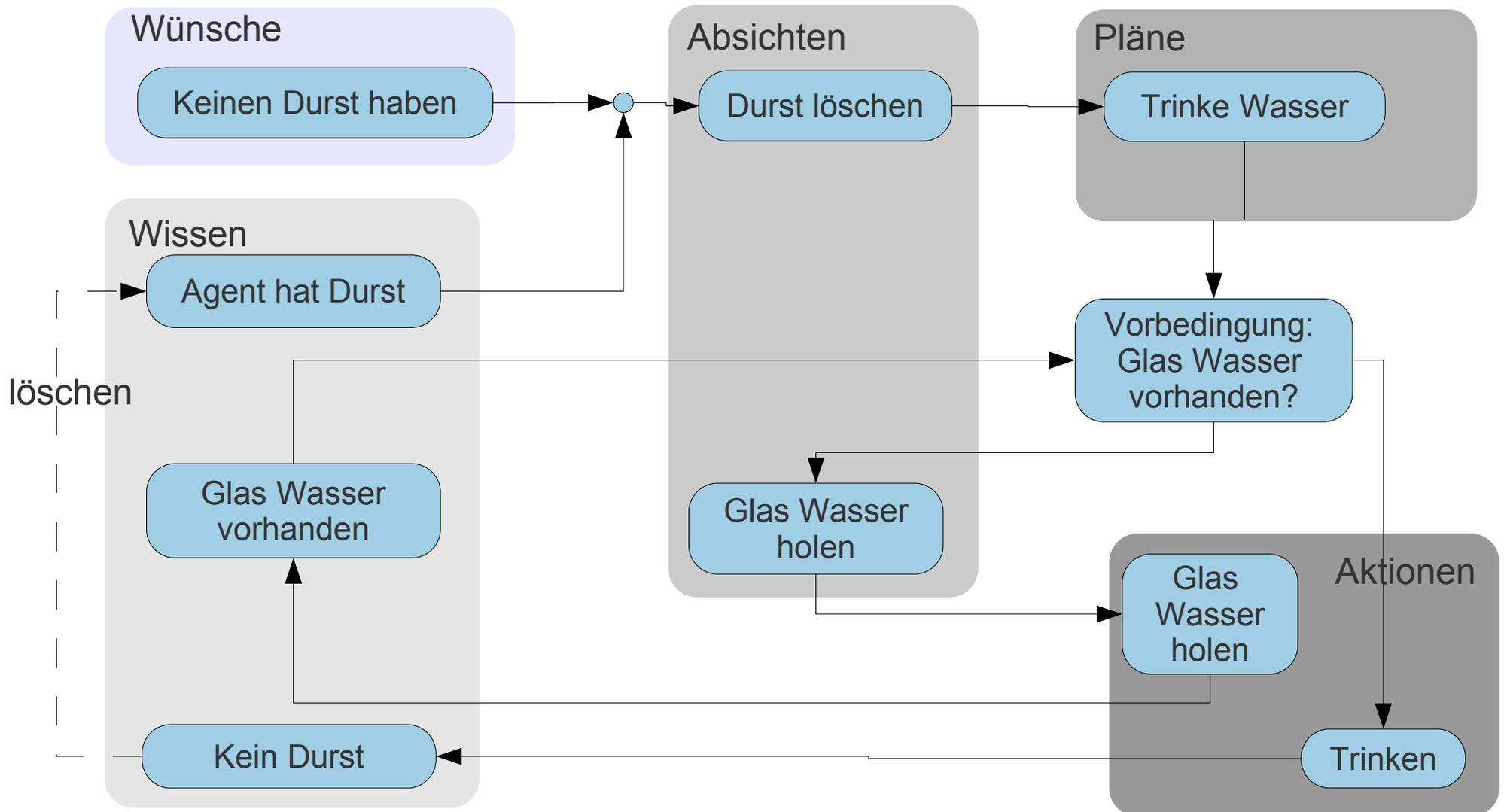
- Repräsentation von Wissen, Wünschen, Absichten
  - Es muss eine verarbeitbare Repräsentation geben.
  - Diverse Möglichkeiten
    - Skriptstücke, Klassen, if-then-Ketten zur Auswahl (wenig flexibel)
    - Prolog-Fakten und -Regeln
  - Im Folgenden nicht weiter spezifiziert



# Entscheidungsfindung

- Wissensrevision (Belief revision)
  - angestoßen durch Sensoreingaben (Wahrnehmung, Percepts) oder interne Schlussfolgerungen
- Optionen finden
  - Bestimmen der aktuell möglichen Handlungen
  - Beeinflusst durch Pläne
- Filter
  - Finden einer Menge von widerspruchsfrei verfolgbaren Wünschen
  - Erklären dieser Wünsche zu Absichten

# Durstiger Agent in BDI



# Entscheidungsfindung

- Optionen finden

- *options*:  $\wp(Bel) \times \wp(Int) \rightarrow \wp(Des)$

- Nimmt als Argument eine Teilmenge der Menge aktueller Fakten (*Bel*), eine Teilmenge der Menge aktueller Absichten (*Int*) und liefert daraus eine Teilmenge der Menge aktueller Optionen (*Des*) (=realisierbare Wünsche)
    - Andere Optionen sind nicht sinnvoll, da sie den aktuellen Fakten oder Absichten widersprechen.

- Optionen filtern

- *filter*:  $\wp(Bel) \times \wp(Int) \times \wp(Des) \rightarrow \wp(Int)$

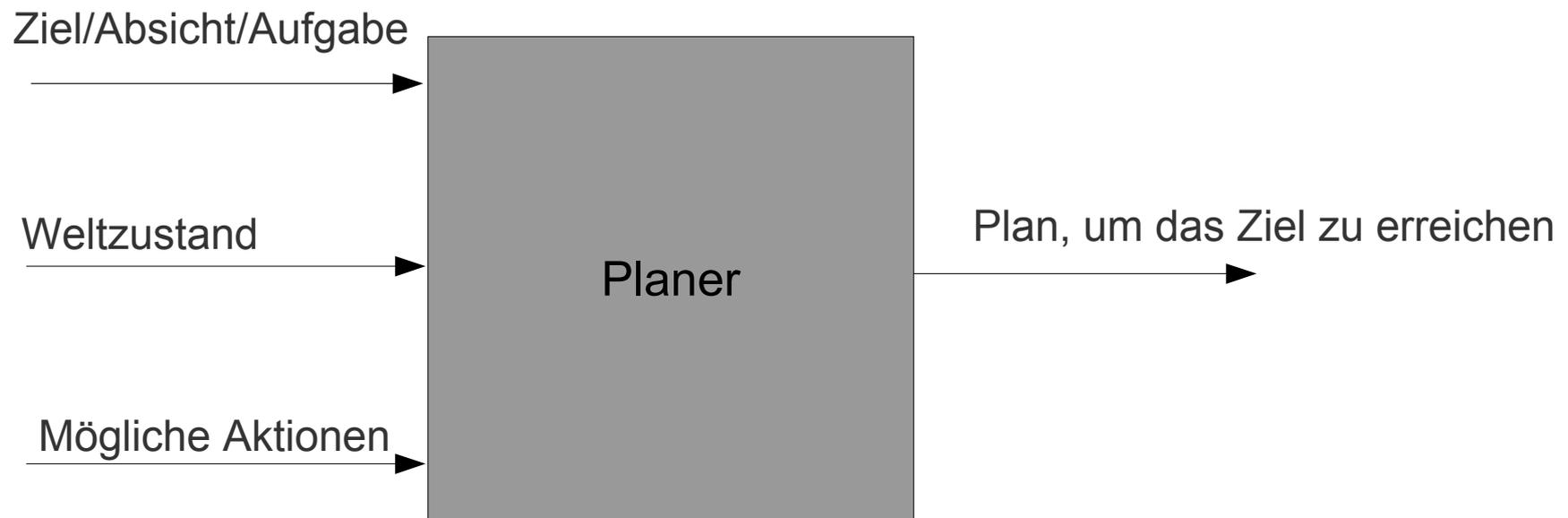
- Aus den gegebenen Fakten, Absichten und Optionen (Wünschen), liefere jene Absichten, die aktuell unternommen werden sollen.

- Auf dem Laufenden bleiben (belief revision function)

- *brf*:  $\wp(Bel) \times Per \rightarrow \wp(Bel)$

- Die Fakten müssen durch neue Wahrnehmungen (*Per*, Percepts) aktualisiert werden

# Vorgang des Planens



# Prädikatenlogik

- Zustand der Welt in Form von logischen Formeln beschreiben
- Aussagen über die Welt durch Prädikatenlogik (erster Ordnung)
  - Aussagen haben die Form *Prädikat(Subjekt, Subjekt,...)*, etwa
    - groß(Baum)
    - befindet\_sich(Michael, Raum 1406C)
    - steht\_zwischen(Michael, Philipp, Roland)
  - Wohlgeformte Formel (wff, kurz: Formel) bildet sich aus:
    - Prädikate (meist Großbuchstaben P, Q oder was vor der Klammer steht, Stelligkeit üblicherweise  $\geq 1$ )
    - Konstanten (Kleinbuchstaben)
    - Funktionen  $f(t_1, t_2, \dots, t_n)$ , Stelligkeit  $\geq 1$
    - Variablen (Kleinbuchstaben: x, y, z, a, b, ...)
    - Logische Operatoren:  $\wedge$  (und),  $\vee$  (oder),  $\neg$  (nicht),  $\forall$  (für alle),  $\exists$  (es gibt),  $\rightarrow$  (folgt),  $\leftrightarrow$  (folgt und umgekehrt)
    - Klammerung ( ) und ggf. = (Gleichheit, nicht unbedingt vorhanden)

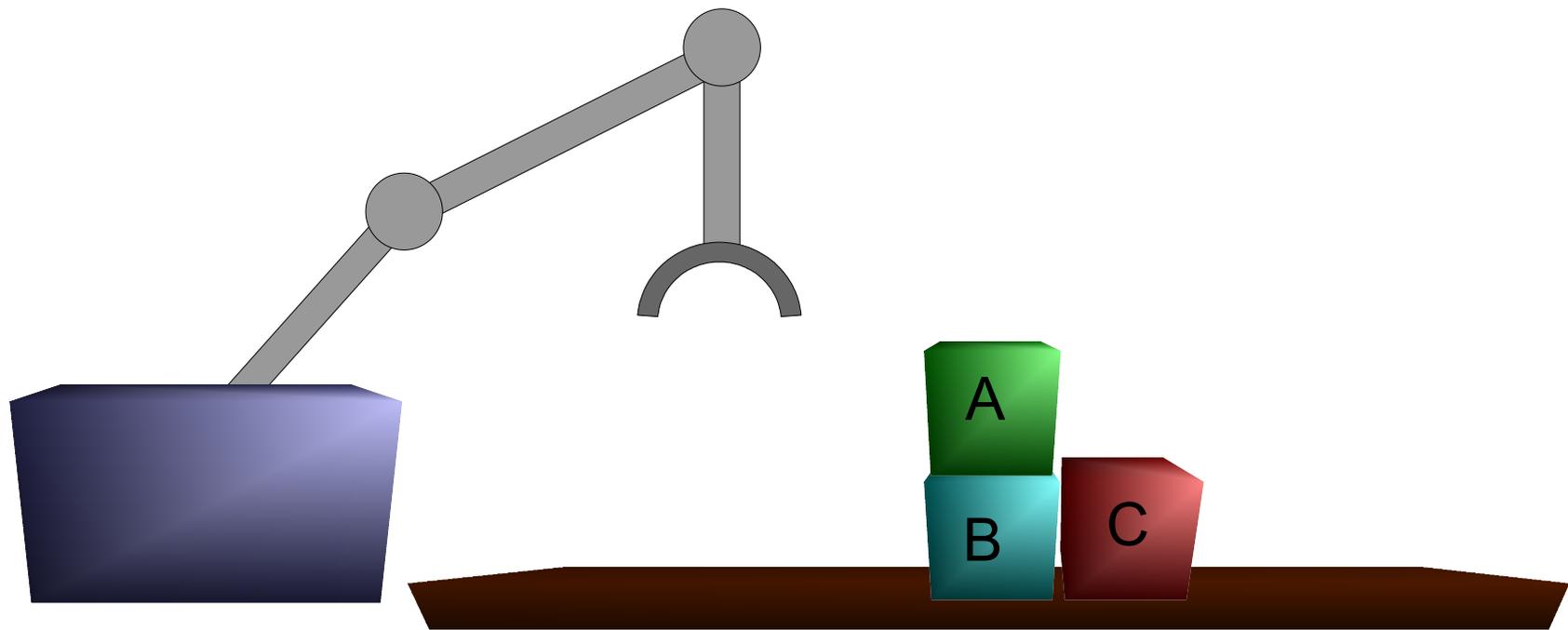
# Prädikatenlogik

- Bildungsgesetze für Terme
  - Jede Konstante ist ein Term
  - Jede Variable ist ein Term
  - Jeder Ausdruck  $f(t_1, t_2, \dots, t_n)$  ist ein Term, wenn  $t_1, t_2, \dots, t_n$  Terme sind ( $n > 0$ )
  - Nichts anderes ist ein Term
- Bildungsgesetz für Formeln
  - $P a_1 a_2 a_3 \dots a_n$  ist eine Formel, wenn  $a_1, a_2, a_3, \dots, a_n$  Terme sind und  $P$  ein  $n$ -stelliges Prädikat ist (auch:  $P(a_1, a_2, a_3, \dots, a_n)$ )
  - Wenn Gleichheit zur Logik gehört, ist  $(a_1 = a_2)$  eine Formel
  - Wenn  $F, G$  Formeln sind, dann auch  $\neg F, (F \wedge G), (F \vee G), (F \rightarrow G), (F \leftrightarrow G)$
  - Wenn  $F$  Formel ist, dann auch  $\forall x F, \exists x F$
  - Nichts anderes ist eine Formel
- Zweistellige Prädikate  $P(a,b)$  dürfen auch als  $aPb$  geschrieben werden (ebenso Funktionen); statt  $\langle (+(x,2),4) \rangle$  schreibt man  $(x+2) < 4$

# STRIPS

- STRIPS-Formalismus (Fikes, Nilsson 1971)
  - Stanford Research Institute Problem Solver
  - Angabe der Startbedingungen in Form einer Menge von Prädikaten
  - Definition von Aktionen mit
    - Name: Bezeichnung der Aktion, ggf. mit Argumenten
    - Vorbedingungen: Welche Prädikate müssen erfüllt sein, damit die Operation erlaubt ist?
    - Effekten: Wie ändert sich der Zustand der Welt nach der Operation?
  - Effekte werden üblicherweise in Form von Lösch- und Addierlisten formuliert
    - Welche Prädikate werden ungültig? (del)
    - Welche Prädikate gelten ab sofort? (add)

# Der stapelnde Roboter



# Der stapelnde Roboter

- Welt = { frei(A), auf(A,B), Tisch(B), Tisch(C), frei(C) }
  - frei(X): Auf X liegt kein Stein
  - auf(A,B): A liegt auf B
  - Tisch(X): X liegt auf dem Tisch
- Ziel: Welt = { Tisch(A), Tisch(B), Tisch(C) }
- Weitere Prädikate:
  - gehalten(X) = Arm hält X;
  - Arm\_frei
    - nullstelliges Prädikat, gilt, wenn es vorhanden ist; kann auch als frei(Arm) formuliert werden

# Der stapelnde Roboter

## Aktionen

- stapeln(x,y)
  - pre = { frei(y), gehalten(x) }
  - del = { frei(y), gehalten(x) }
  - add = { Arm\_frei, auf(x,y) }
- runternehmen(x,y)
  - pre = { auf(x,y), frei(x), Arm\_frei }
  - del = { auf(x,y), Arm\_frei }
  - add = { gehalten(x), frei(y) }
- aufnehmen(x)
  - pre = { frei(x), Tisch(x), Arm\_frei }
  - del = { Tisch(x), Arm\_frei }
  - add = { gehalten(x) }
- absetzen(x)
  - pre = { gehalten(x) }
  - del = { gehalten(x) }
  - add = { Arm\_frei, Tisch(x) }

Wie wählt man nun eine Kette von Aktionen, um das Ziel zu erreichen?

# Auswahl von Plänen

- $Ac = \{ \alpha_1, \dots, \alpha_n \}$  sei feste Menge von Aktionen
- Aktionsbeschreibung  $\langle P_\alpha, D_\alpha, A_\alpha \rangle$ 
  - $P_\alpha$ : Vorbedingung
  - $D_\alpha$ : Fakten, die durch  $\alpha$  ungültig werden (delete)
  - $A_\alpha$ : Fakten, die durch  $\alpha$  gültig werden (add)
- Planungsproblem (über einer Menge  $Ac$ ) ist festgelegt durch  $\langle \Delta, O, \gamma \rangle$ 
  - $\Delta$ : log. Formeln, betreffend das Wissen um den ursprünglichen Weltzustand
  - $O$ :  $\{ \langle P_\alpha, D_\alpha, A_\alpha \rangle \mid \alpha \in Ac \}$ , eine Menge möglicher Aktionsbeschreibungen
  - $\gamma$ : log. Formeln, welche das Ziel/die Absicht/die Aufgabe repräsentieren

Logische Ausdrücke  
erster Ordnung

# Auswahl von Plänen

- Plan  $\pi = (\alpha_1, \dots, \alpha_n)$ ,  $\alpha_i \in \mathcal{Ac}$
- Ein Plan definiert eine Folge von Umgebungsmodellen  $\Delta_i$ 
  - $\Delta_0 := \Delta$
  - $\Delta_i := (\Delta_{i-1} \setminus D_{\alpha_i}) \cup A_{\alpha_i}$  für  $i=1..n$
- Ein (linearer) Plan  $\pi$  ist *akzeptabel* bezüglich des Problems  $\langle \Delta, O, \gamma \rangle$ , wenn gilt
  - $\Delta_{i-1} \models P_{\alpha_i}$  für alle  $i=1..n$
- Ein (linearer) Plan  $\pi$  ist *korrekt* bezüglich  $\langle \Delta, O, \gamma \rangle$ , wenn gilt
  - $\pi$  ist akzeptabel
  - $\Delta_n \models \gamma$  (das Ziel wird also im letzten Schritt erreicht)

# Mittel-Zweck-Schlussfolgerung

Sei ein Problem  $\langle \Delta, O, \gamma \rangle$  gegeben. Finde einen korrekten Plan  $\pi$  für  $\langle \Delta, O, \gamma \rangle$  oder gib bekannt, dass es keinen korrekten Plan gibt.

$$plan: \wp(Bel) \times \wp(Int) \times \wp(Ac) \rightarrow Plan$$

- $plan$  wählt einen Plan  $\pi = plan(B, I, A)$  aus der Menge aller Pläne ( $Plan$ ) aus
  - Der Agent hat damit festgestellt, dass der gewählte Plan derjenige ist, der zur Erreichung seiner Ziele zweckmäßig ist (*Mittel-Zweck-Schlussfolgerung*).
- Wie entsteht ein Plan?
  - $Plan$  kann eine Sammlung vorgefertigter Aktionsketten sein
  - Typische Vereinfachung
    - Prüfe, ob die Vorbedingung des Plans mit dem Agentenwissen harmoniert.
    - Prüfe, ob die Nachbedingung des Plans mit dem Ziel harmoniert.
    - In diesem Falle wähle den Plan und führe ihn aus.

# Realisierung

- Definitionen
  - $pre(\pi)$ : Vorbedingung für  $\pi$
  - $body(\pi)$ : Inhalt von  $\pi$
  - $empty(\pi)$ : wahr, wenn  $\pi$  leer ist, sonst falsch
  - $execute(\pi)$ : Führe den Plan  $\pi$  aus (der Reihe nach jeden Schritt)
  - $head(\pi)$ : erste Aktion in  $\pi$
  - $tail(\pi)$ : restlicher Plan ohne die erste Aktion
  - $sound(\pi, I, B)$ : wahr, wenn  $\pi$  ein korrekter Plan für die Umsetzung der Absichten  $I \subset Int$  unter Fakten  $B \subset Bel$  ist, sonst falsch.

# Realisierung

```
B = B0;           // Anfangswissen
I = I0;           // Anfangsabsichten
while (true) {
  p = see();      // hole nächste Wahrnehmung
  B = brf(B, p);  // Aktualisiere das Weltwissen
  D = options(B, I); // Finde aktuelle Optionen
  I = filter(B, D, I); // und wähle Absichten
  pi = plan(B, I, Ac); // Wähle einen Plan
  while (not (empty(pi) or succeeded(I,B) or impossible(I,B))) {
    alpha = head(pi); // nächste Aktion
    execute(alpha); // ausführen
    pi = tail(pi); // Liste weiterschieben
    p = see(); // Was hat sich getan?
    B = brf(B, p); // Aktualisieren
    if (reconsider(I,B)) { // Lieber neue Absichten wählen?
      D = options(B, I); // Finde Optionen
      I = filter(B, D, I); // Bestimme Absichten
    }
    if (not sound(pi, I, B)) { // Ist der alte Plan noch sinnvoll?
      pi = plan(B, I, Ac); // Wenn nein, neuen Plan aktivieren
    }
  }
}
```

# Verpflichtung

- Wie sehr muss sich der Agent seinen Zielen verpflichtet fühlen? (*commitment*)
  - Auswahl eines neuen Plans
  - Auswahl eines neuen Ziels
- Verschiedene Formen
  - Blinde Verpflichtung: Der Agent folgt seiner Absicht, bis er glaubt, dass das Ziel erreicht ist.
  - Einfältige Verpflichtung: Der Agent folgt seiner Absicht, bis er glaubt, dass das Ziel erreicht ist oder das Ziel unerreichbar ist.
  - Kluge Verpflichtung: Der Agent folgt so lange seiner Absicht, wie er glaubt, dass diese ihr Ziel erreichen kann.

# Aufwand in BDI

- Abwägungen sind aufwändig
  - Begrenztheit der Ressourcen (Speicher, Rechenzeit)
  - Ständige Neuberechnungen von Alternativen können Widersprüche erzeugen und unbestimmt viele Rechenschritte erfordern
  - Zu seltene Abwägung: Agent verfolgt nichterreichtbare Ziele
  - Zu häufige Abwägung: Agent verfolgt praktisch keine Ziele, sondern ist nur mit dem Berechnen von Abwägungen beschäftigt

# Implementierungen

- PRS: Procedural Reasoning System (1987)
  - Erste Implementierung mit expliziter BDI-Struktur
  - Vorkompilierte Pläne, jeweils mit Ziel / Kontext(=Vorbedingung) / Formulierung
  - Pläne können neben Aktionen auch (Unter-)Ziele beinhalten
  - Ziele können mit Alternativen versehen sein oder Schleifen ausführen
- IRMA: Intelligent Resource-Bounded Machine Architecture (1988)
  - Agenten sollen eine vollständige Auswertung neuer Optionen umgehen können, wenn diese ihren bisherigen Absichten widersprechen, es sei denn, diese seien leicht als spezielle Fälle zu identifizieren

# Implementierungen

- ... neueren Datums
  - JADEX (Uni Hamburg)
    - LGPL-lizenzierter BDI-Aufsatz auf JADE ([jade.tilab.com](http://jade.tilab.com))
  - JAM
    - Hybridagentensystem, basierend auf PRS  
[http://www.marcush.net/IRS/irs\\_downloads.html](http://www.marcush.net/IRS/irs_downloads.html)
  - Jack
    - kommerzielles System
    - eigene Programmiersprache, basiert auf Java
    - <http://www.agent-software.com/shared/home/>
  - AgentBuilder
    - kommerzielles System
    - Aufwändiges Toolkit zur Erzeugung von Anwendungen mit intell. Agenten
    - <http://www.agentbuilder.com/>

# Emotionen

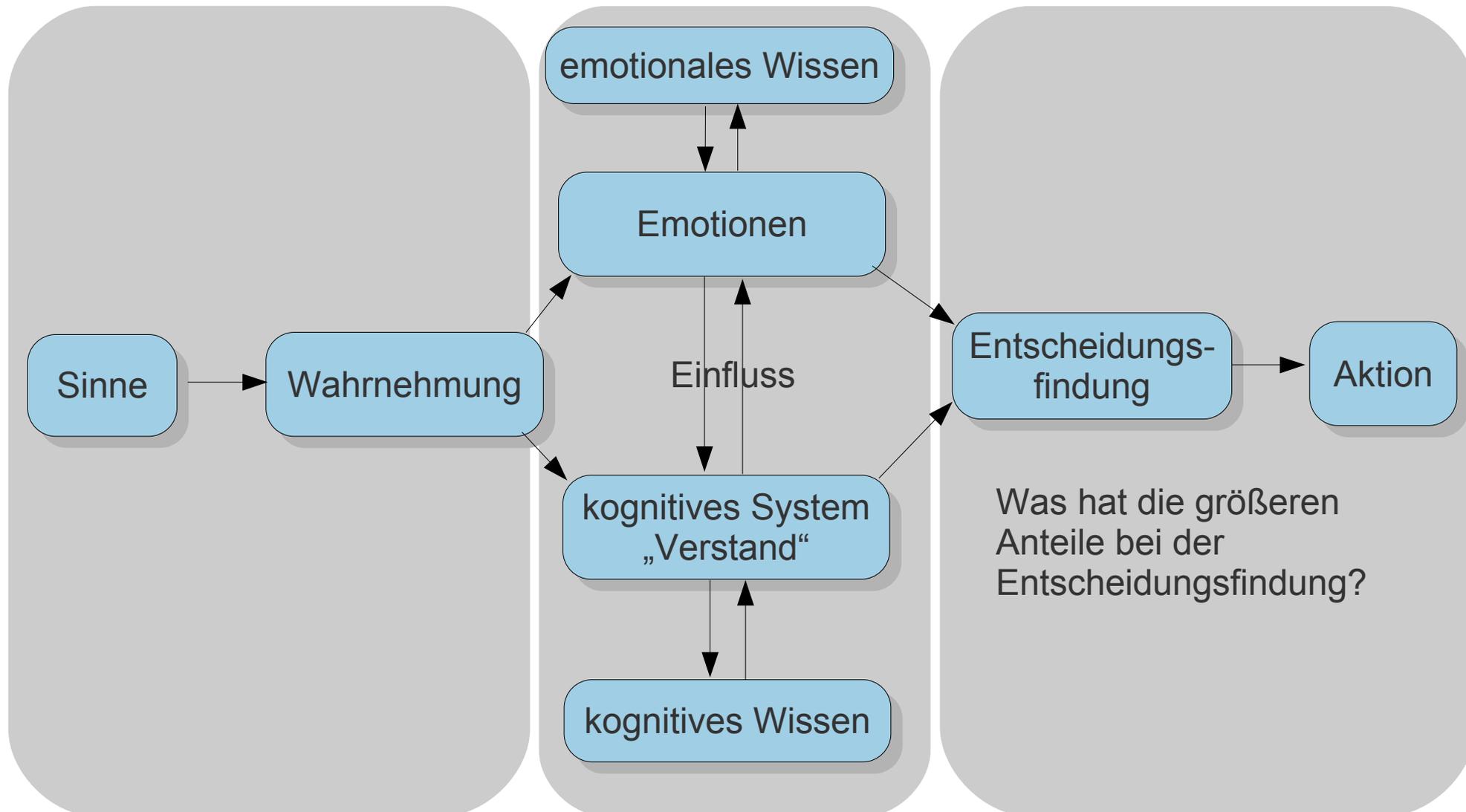
- BDI: Lücken in der Modellierung
  - Lernverhalten?
  - Kooperation mit anderen Agenten?
- Emotionen als Faktor
- Neuere Erkenntnisse der Wahrnehmungspsychologie
  - Der Mensch ist nun einmal kein „kühl“ abwägendes Planungssystem



© Paramount Pictures

Emotionen in Softwareagenten?

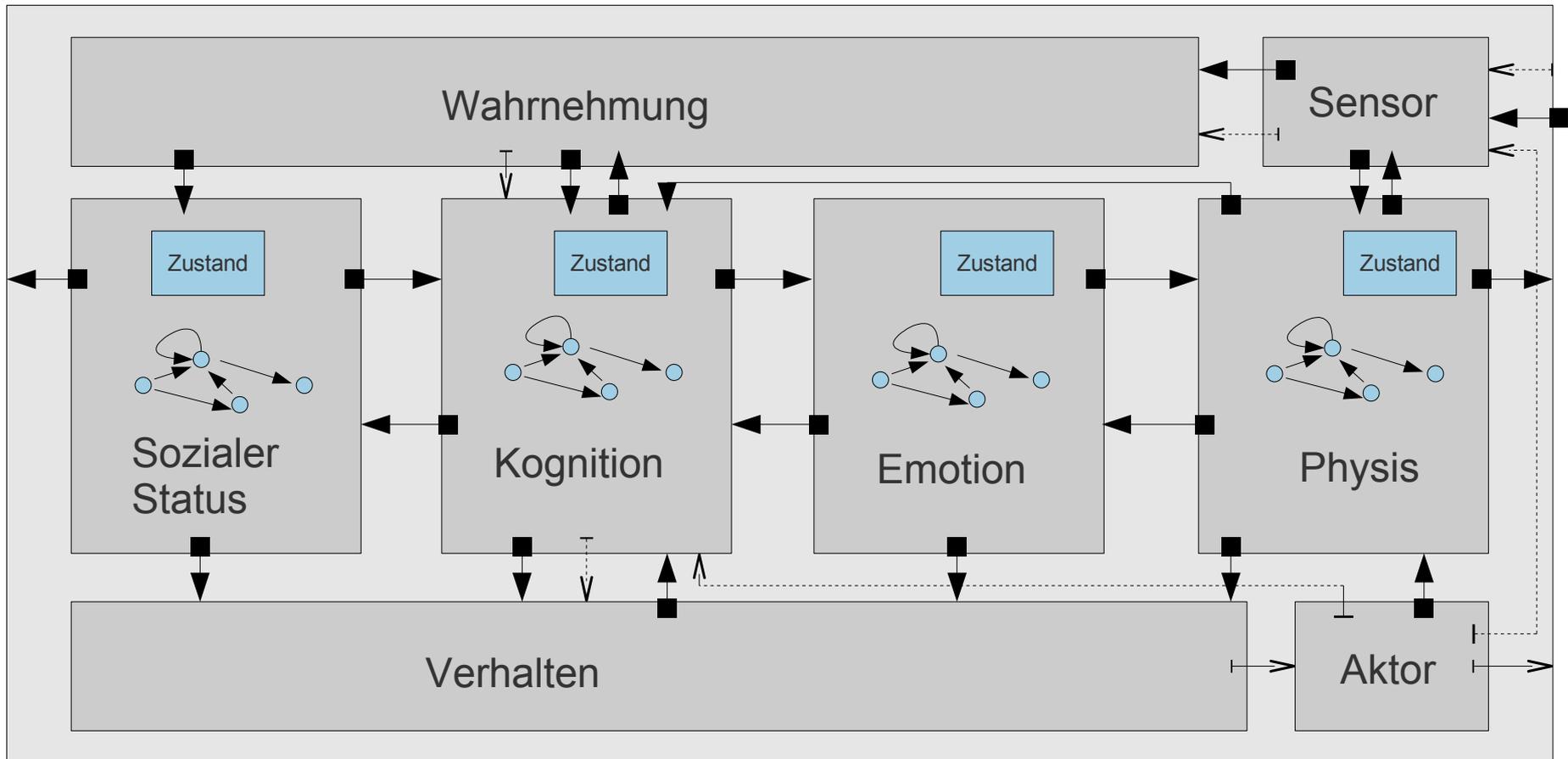
# Emotion und Kognition – generelles Modell



# Das PECS-Modell

- Drei-Schichten-Modell
  - Eingabeschicht
    - Sensoren und Wahrnehmung
  - Verarbeitungsschicht (Interne Schicht)
    - Physis, Emotion, Kognition, Status
  - Ausgabeschicht
    - Verhalten
    - Aktoren

# Basisarchitektur eines PECS-Agenten



- → Kausale Abhängigkeit (Einflussnahme in nicht spezifizierter Weise)
- - - > Datenfluss

# Eingabeschicht

- Verarbeitung der ankommenden Daten
  - Sensor empfängt Informationen aus der Umgebung
  - Sensor empfängt Informationen aus der Gemeinschaft der PECS-Agenten
  - Wahrnehmung leistet Vorverarbeitung, filtert Informationen

# Interne Schicht

- Funktion
  - beschreibt Zustand des Agenten
  - beinhaltet Zustandsvariablen und Zustandsübergangsfunktionen
  - Wissensbasis
  - bestimmt Verhalten
- Physis
  - betrifft physische Gegebenheiten des Agenten
  - in reiner Softwareumgebung eher unwichtig, aber für autonome Systeme mit Hardware (Roboter) sehr wichtig
    - Position im Raum
    - Beschaffenheit der Umgebung

# Interne Schicht

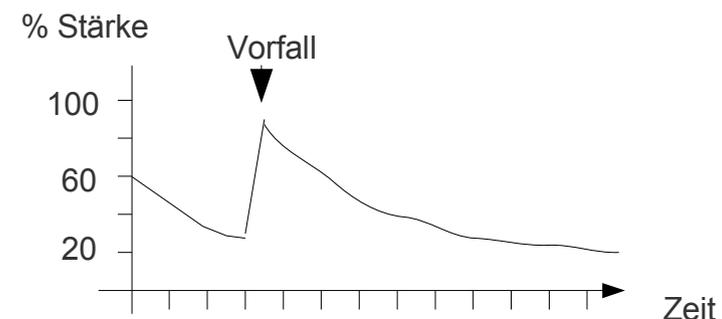
- Sozialer Status
  - Agenten als kooperierende Komponenten („sozial“ in diesem Sinne)
  - Soziale Bedürfnisse beeinflussen das Verhalten
- Emotion
  - modulieren das Verhalten des Agenten außerhalb der Kognition
  - Modell verfolgt den Ansatz der kognitiv ausgelösten Emotion
    - man beachte die Abhängigkeiten
- Kognition
  - Informationsverarbeitende und speichernde Vorgänge
  - Eigentliche Schlussfolgerungen auf Regelbasis

# Ausgabeschicht

- Verhalten
  - Verhaltensrepertoire
  - Koordination der Verhaltensformen
  - Weiterleitung an die Aktoren
- Aktoren
  - Aktoren können externe Aktionen oder interne Aktionen bewirken
  - „Selbstbewusstes“ Handeln: Informationsfluss zur Kognition

# Wirkung von Emotionen

- Beispiel Angst
  - Agent kommt in eine „gefährliche“ Umgebung
    - vielleicht sicherheitstechnisch unsicher?
    - Agent bemerkt fremden Zugriff auf seine Daten
- Kognition (Angriff bemerkt) löst Emotion „Angst“ aus
  - Emotionswert ist Teil des Zustands der Komponente „Emotion“
- Emotion klingt mit der Zeit ab
  - etwa vermöge einer negativen Exponentialfunktion



# Bedeutung der sozialen Komponente

- Szenario „Lerngruppe“
  - Agenten lernen/erforschen auf eigene Faust oder in Gruppen
  - Vorteil des Alleinseins: Schnellerer Fortschritt (wo möglich)
  - Vorteil der Gruppe: Fortschritt dort, wo er alleine nicht mehr möglich ist
  - Zusammenhalt der Gruppe
  - Eine Gruppe besteht aus einer Mindestmenge an Agenten; bei weniger Agenten löst sie sich auf
  - Problem: Abwägung
    - sich lieber der Gruppe anschließen und den Lernfortschritt allgemein verlangsamen?
    - sich lieber wieder eigenen Studien widmen und riskieren, dass man nicht mehr vorankommt? Und dabei Ursache sein, dass die Gruppe zerfällt?
- ➔ Soziales Verhalten: Das Wohl einer Gruppe fördern

# Lerngruppe

- Repertoire der Agentenaktionen
  - Lernen (kann von außen nicht beobachtet werden)
  - Gruppe auswählen
  - Gruppe beitreten
  - Gruppe starten
  - Gruppe verlassen
  - Gruppe auflösen
- Bedürfnisse des Agenten
  - Wissenserwerb: das eigene Wissen mehrer
  - Soziale Befriedigung: in einer Gruppe agieren

# Lerngruppe

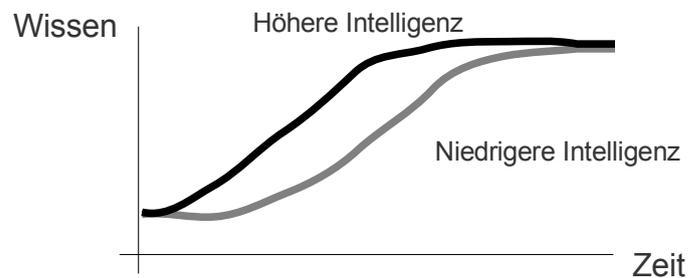
## Modell

$$\text{WissErw}' = a * \text{WissGrz} * \text{WissNorm} * \text{Intelligenz}/100 * \text{WissErw}$$

Differentialgleichung für den Wissenserwerb

$$\text{WissGrz} = (\text{WissErwMax} - \text{WissErw}) / \text{WissErwMax}$$

Lerngrenze



Individuelles Lernen

# Lerngruppe

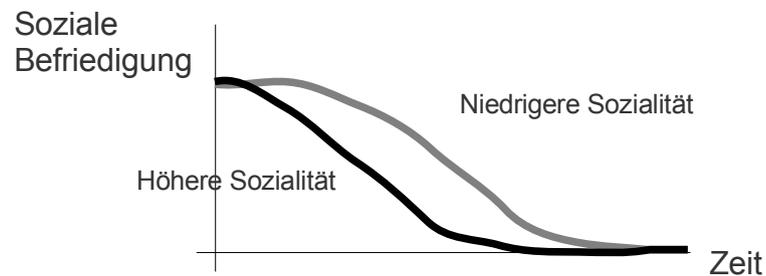
## Modell

$$\text{SozBef}' = -b * \text{SozGrz} * \text{SozNorm} * \text{Sozialität}/100 * \text{SozBef}$$

Differentialgleichung für die soziale Befriedigung

$$\text{SozGrz} = (\text{SozBefMax} - \text{SozBef}) / \text{SozBefMax}$$

Lerngrenze



Individuelles Lernen

# Lerngruppe

## Modell

$$\text{WissErw}' = a * \text{WissGrz} * \text{WissNorm} * \text{Intelligenz}/100 * \text{WissErw} * \text{QualitätGruppe}$$

Differentialgleichung für den Wissenserwerb in der Gruppe

$$\text{SozBef}' = -b * \text{SozGrz} * \text{SozNorm} * \text{Sozialität}/100 * \text{SozBef} * \text{QualitätGruppe}$$

Differentialgleichung für die soziale Befriedigung in der Gruppe

# Lerngruppe

- Werte aus den Gleichungen werden zu Verhalten
- Aktionen
  - Gruppe auswählen
    - WissErw' wird zu klein (Lerngeschwindigkeit sinkt unter Mindestniveau)
    - SozBef wird zu klein (Agent sehnt sich nach Gesellschaft)
  - Gruppe verlassen
    - WissErw' wird zu klein (Auch in der Gruppe kommt man nicht mehr voran)
    - SozBef ist hoch (Agent kann auf Gesellschaft für eine Weile verzichten)
  - ...

# Ergebnis

- Agenten zeigen ein Verhalten, das dem menschlichen nahe kommt
  - Es kommt zu dynamischer Gruppenbildung (mit Neubildung und Auflösung)
- Komplexe Simulationen von Menschengruppen möglich
- Auch Kombinationen mit Emotionen erwägen
  - Angst machende Agenten in der Gruppe?

# Vergleich

- BDI
  - eignet sich sehr gut zur Modellierung von Schlussweisen
  - basiert auf (früheren) psychologischen Erkenntnissen und Modellen
  - große Unterstützung in der Agentenwelt, zahlreiche Modellsysteme
- PECS
  - kann sehr gut in der Simulation von Gruppenprozessen eingesetzt werden
  - besseres Abbild der menschlichen Psyche als BDI
  - bislang geringe Verbreitung

Einsatz solcher Methodiken für „intelligente“ Agenten sinnvoll, jedoch ist der Aufwand abzuwägen.