

Grundlagen der Künstlichen Intelligenz

Sebastian Iwanowski
FH Wedel

Kap. 1:
Einführung und Überblick

Vorlesungsüberblick

Inhaltliche Voraussetzungen:

GdP/GTI, Programmieren I und II

hilfreich: Objektorientierte Programmiertechnik

Lernziele dieser Vorlesung:

Anwendungsgebiete und -beispiele

Methoden, Techniken und Architekturen der KI

Theoretisches Verständnis der zugrundeliegenden Logik

Detailkenntnisse in Technischer Diagnose

Vorlesungsüberblick

Vorlesungsgliederung:

1. Einführung und Überblick
2. Logische Grundlagen der KI
3. Algorithmische Grundlagen der KI
4. Verschiedene Wissensrepräsentationsarten und deren Verarbeitung
5. Anwendungsgebiet: Technische Diagnose

Was ist KI ?

Der Turing-Test



Eine Software verhält sich intelligent, wenn ein Mensch ihr Verhalten nicht vom Verhalten eines Menschen unterscheiden kann.

Anwendungsgebiet: Spiele-KI

Rundenbasierte Strategiespiele

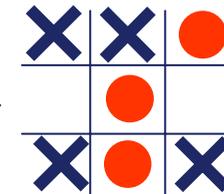
Problem:

Finde eine Strategie von einem Startzustand zu einem Gewinnzustand

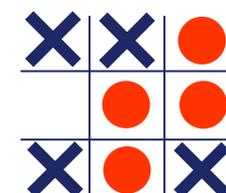
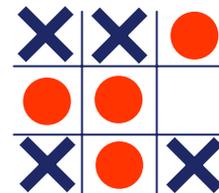


Bsp: Tic-Tac-Toe

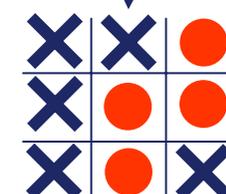
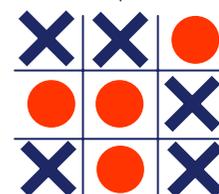
Ausgangszustand



Folgezustand ●



Folgezustand X



Quelle und nähere Infos: Seminarvortrag und Ausarbeitung von Nils Böckmann, SS 2005, Nr. 13

<http://www.fh-wedel.de/Archiv/iw/Lehrveranstaltungen/SS2005/SeminarKI.html>

Anwendungsgebiet: Spiele-KI

Rundenbasierte Strategiespiele

Schachcomputer

Meilenstein 1997:

Kasparov 2.5 – Deep Blue 3.5



Weitere infos: <http://www.research.ibm.com/deepblue>

Anwendungsgebiet: Spiele-KI

Rundenbasierte Strategiespiele

Schachcomputer

Unterschied 1

- Deep Blue kann über 200,000,000 Schach-Situationen pro Sekunde errechnen und bewerten.
- Kasparov nur 3.

Unterschied 2

- Deep Blue hat nur begrenztes Wissen über Schach, aber riesiges über Berechnungen.
- Bei Kasparov ist das umgekehrt.

Unterschied 3

- Deep Blue hat keine Emotionen oder Gefühle.
- Kasparov nutzt diese, um Weltklasseschach zu spielen.

Anwendungsgebiet: Spiele-KI

Rundenbasierte Strategiespiele

Schachcomputer

Unterschied 4

- Deep Blue hat von vielen Leuten (Schachmeistern etc.) profitiert.
- Kasparov wird nur von seinem Coach Yuri Dokhoian und seiner eigenen Leidenschaft, das beste Schach der Welt zu spielen, gefördert.

Unterschied 5

- Deep Blue ist kein lernendes System. Es ist daher nicht fähig, von seinem Gegner zu „lernen“ oder über die momentane Situation des Schachspiels „nachzudenken“.
- Kasparov ist fähig, sehr schnell aus seinen Fehlern und Erfolgen zu lernen.

Unterschied 6

- Deep Blue kann durch nichts und niemanden von seinem Spiel abgebracht werden.
- Kasparov ist stets durch menschliche Schwächen wie: Langeweile, Verlust der Konzentration etc. beeinflussbar.

Anwendungsgebiet: Spiele-KI

Rundenbasierte Strategiespiele

Schachcomputer

Unterschied 7

- Deep Blue kann sehr effektiv Schachprobleme zu lösen, aber nichts anderes.
- Gary Kasparov ist sehr vielseitig: Er schrieb mehrere Bücher, ist politisch aktiv, etc.

Unterschied 8

- Jeder Wechsel der Spielstrategie von Deep Blue muss von seinen Programmierern nach dem Spiel vorgenommen werden.
- Kasparov kann jederzeit seine Strategie ändern.

Unterschied 9

- Deep Blue kann nur Schachpositionen bewerten, nicht aber Schwächen seines Gegners.
- Kasparov kann die Schwächen seines Gegners herausfinden und dann ausnutzen.

Unterschied 10

- Deep Blue berechnet sehr viele Züge voraus, um die beste Position herauszufinden.
- Kasparov macht das genauso, schafft aber wesentlich weniger.

Anwendungsgebiet: Spiele-KI

Echtzeit-Strategiespiele

Beispiele:



Half Life



Command + Conquer 3

Anwendungsgebiet: Spiele-KI

Echtzeit-Strategiespiele

Typische KI-Anforderungen:

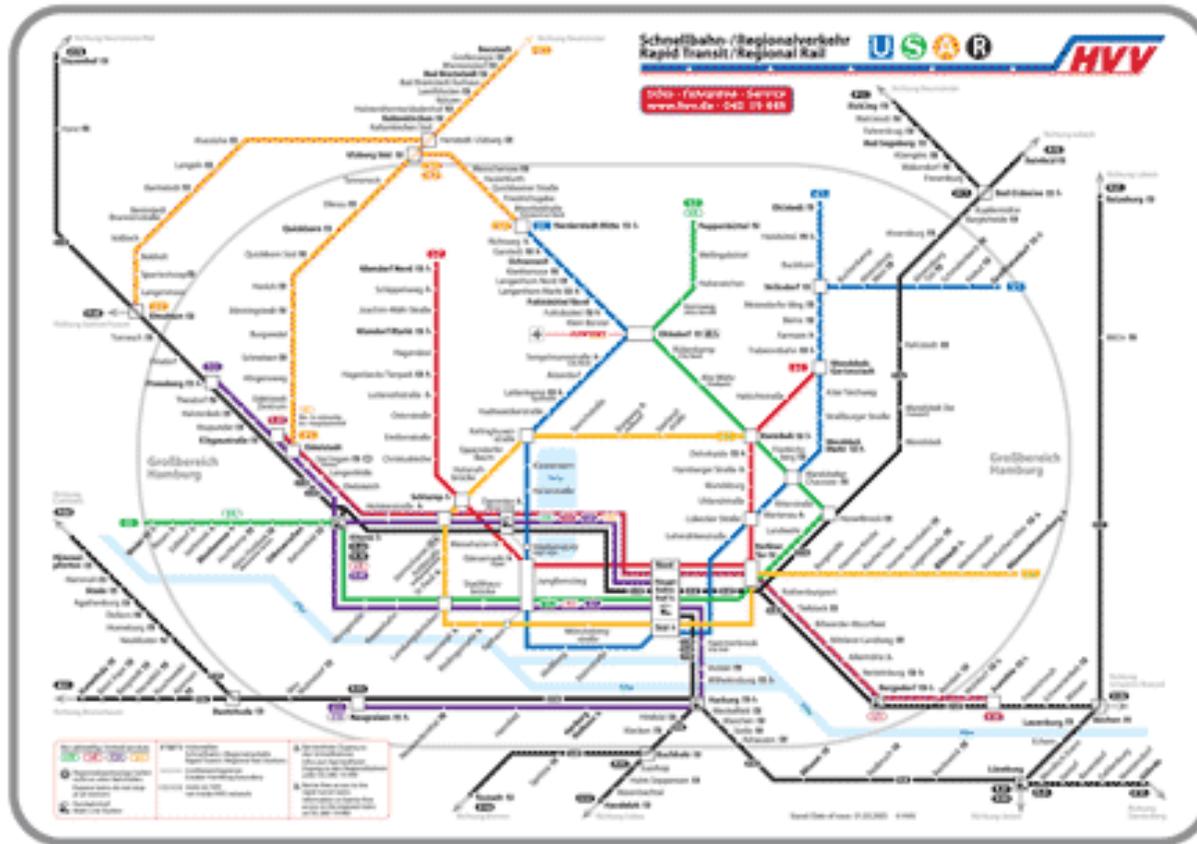
- Wegfindung und Terrainanalyse
- Ressourcen-Planung
- Taktiken und Strategien

Quelle: Seminarvortrag und Ausarbeitung von Julian Huppertz, SS 2007, Nr. 1

Basisproblem: Das Problem des kürzesten Weges

Problem:

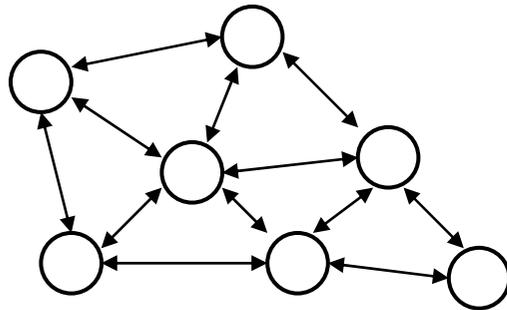
Finde zu zwei Punkten A und B in einem gegebenen Verkehrsnetz den kürzesten Weg von A nach B, der ausschließlich Strecken dieses Verkehrsnetzes benutzt.



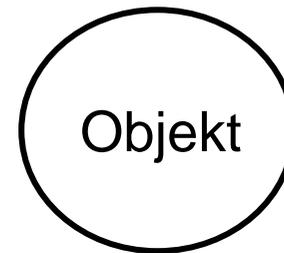
Weitere Infos: Seminarvortrag und Ausarbeitung von Stefan Görlich, SS 2005, Nr. 5,
<http://www.fh-wedel.de/archiv/iw/Lehrveranstaltungen/SS2005/SeminarKI.html>

Basistechnologie: Agentenorientierte Software

Multiagentensystem:



Softwareagent:



autonom

soziale Kompetenz

reaktiv

proaktiv

Weitere Infos: Seminarvortrag und Ausarbeitung von Matthias Rohr, SS 2004, Nr. 4,
<http://www.fh-wedel.de/~si/seminare/ss04/Termine/Themen.html>, erreichbar über **Archiv/iw**

Anwendungsgebiet: Technische Diagnose

Schilderung der historischen Entwicklung der KI-Technologien am Beispiel eines Anwendungsgebietes

Die folgenden Folien sind im Wesentlichen einem Vortrag entnommen zum Thema *Diagnose technischer Systeme* von Jakob Mauss (QTronic GmbH, Berlin)

Anwendungsgebiet: Technische Diagnose

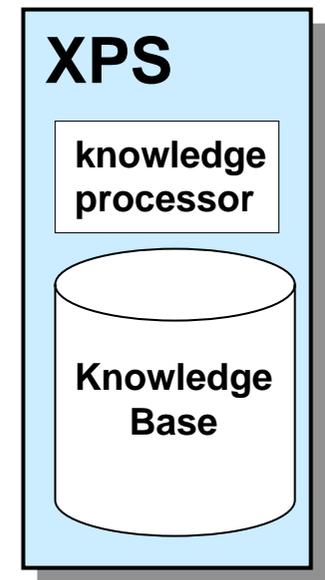
1970er: Diagnose = heuristische Klassifikation

Was ist Diagnose?

Diagnose ist der Schluss von beobachteten Symptomen auf klassifizierte Systemeigenschaften

Einsatzgebiete:

- Medizin
- industrielle Fertigungsprozesse
- Fahrzeugdiagnose
- Luft- und Raumfahrt



Anwendungsgebiet: Technische Diagnose

1970er: Diagnose = heuristische Klassifikation

Was ist **Technische** Diagnose?

Gegeben:

- Ein technisches System (z.B. Auto, Zug)
- Beobachtungen (z.B. Messwerte, Fehlercodes, Fahrerbeanstandung), nicht nominal.

Aufgabe:

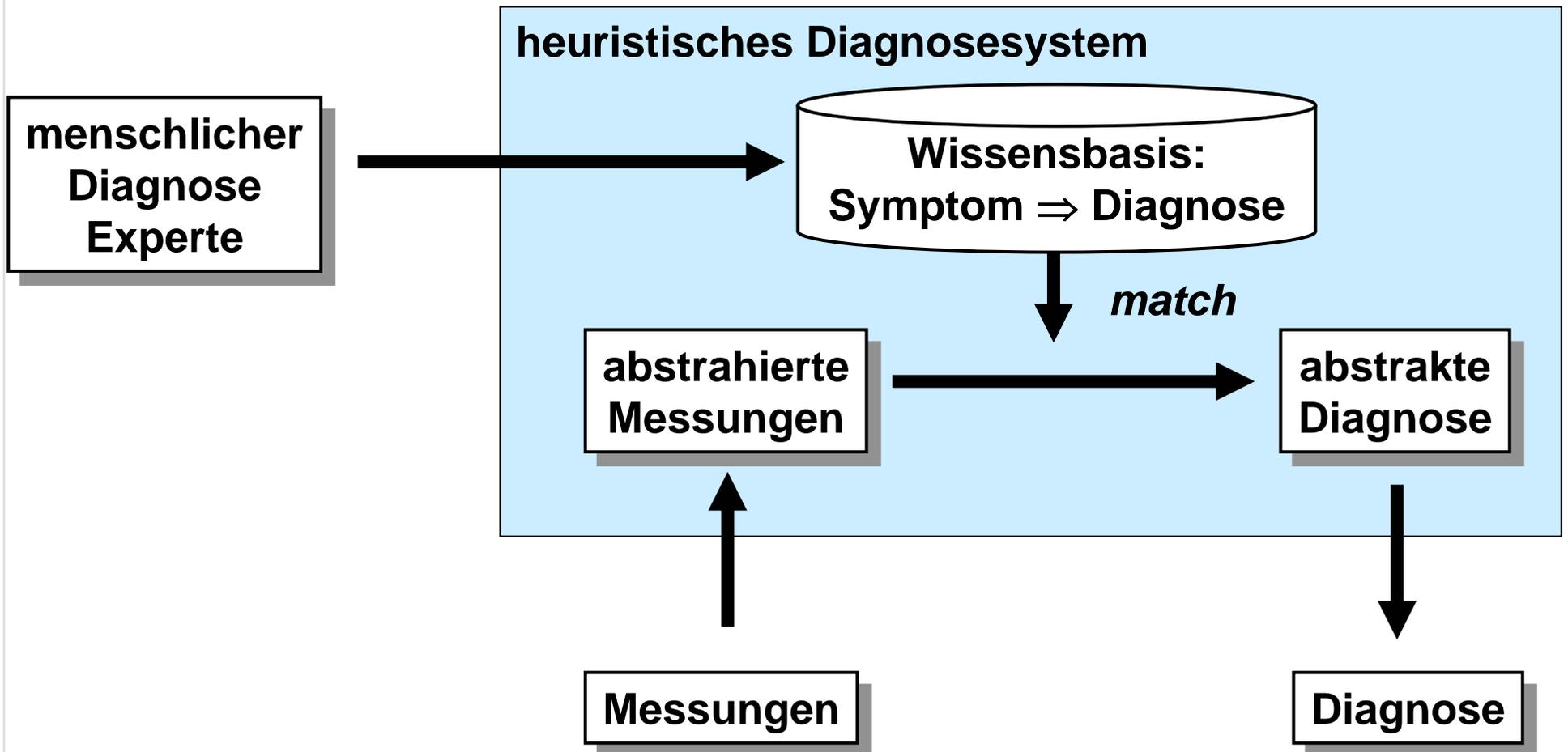
Bestimme,

- in welcher Weise das System defekt ist
- genau genug, dass das nominale Verhalten des Systems wiederhergestellt werden kann.



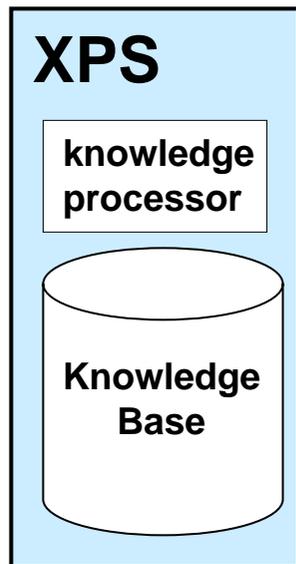
Anwendungsgebiet: Technische Diagnose

1970er: Diagnose = heuristische Klassifikation



Anwendungsgebiet: Technische Diagnose

1970er: Diagnose = heuristische Klassifikation

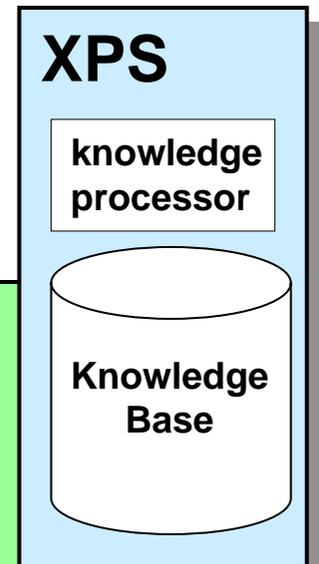


verschiedene Realisierungen des Konzeptes:

- regelbasierte Diagnose:
sicher, probabilistisch, fuzzy, ...
- Diagnose mit Entscheidungsbäumen
- fallbasierte Diagnose
(case-based reasoning CBR)
- Klassifikation mit neuronalen Netzen

Anwendungsgebiet: Technische Diagnose

1970er: Diagnose = heuristische Klassifikation



Stärken:

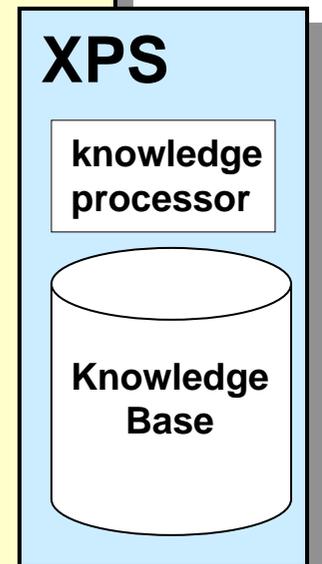
- breit anwendbar
- viele Werkzeuge verfügbar
- Erfahrungen mit vielen unterschiedlichen Anwendungen verfügbar

Anwendungsgebiet: Technische Diagnose

1970er: Diagnose = heuristische Klassifikation

Schwächen:

- Aufbau und Wartung der Wissensbasis:
knowledge-acquisition bottleneck
 - Verfügbarkeit menschlicher Experten
 - Erweiterung der Wissensbasis um neue Fälle
 - Objektivität des erfassten Wissens
 - Vollständigkeit
 - Wiederverwendung einer Wissensbasis
- Wirtschaftlichkeit der Softwareerstellung
- Diagnose unvorhergesehener Fehler
schwierig / unmöglich
- Diagnose von Mehrfachfehlern schwierig
- Erklärung generierter Diagnosen



Anwendungsgebiet: Technische Diagnose

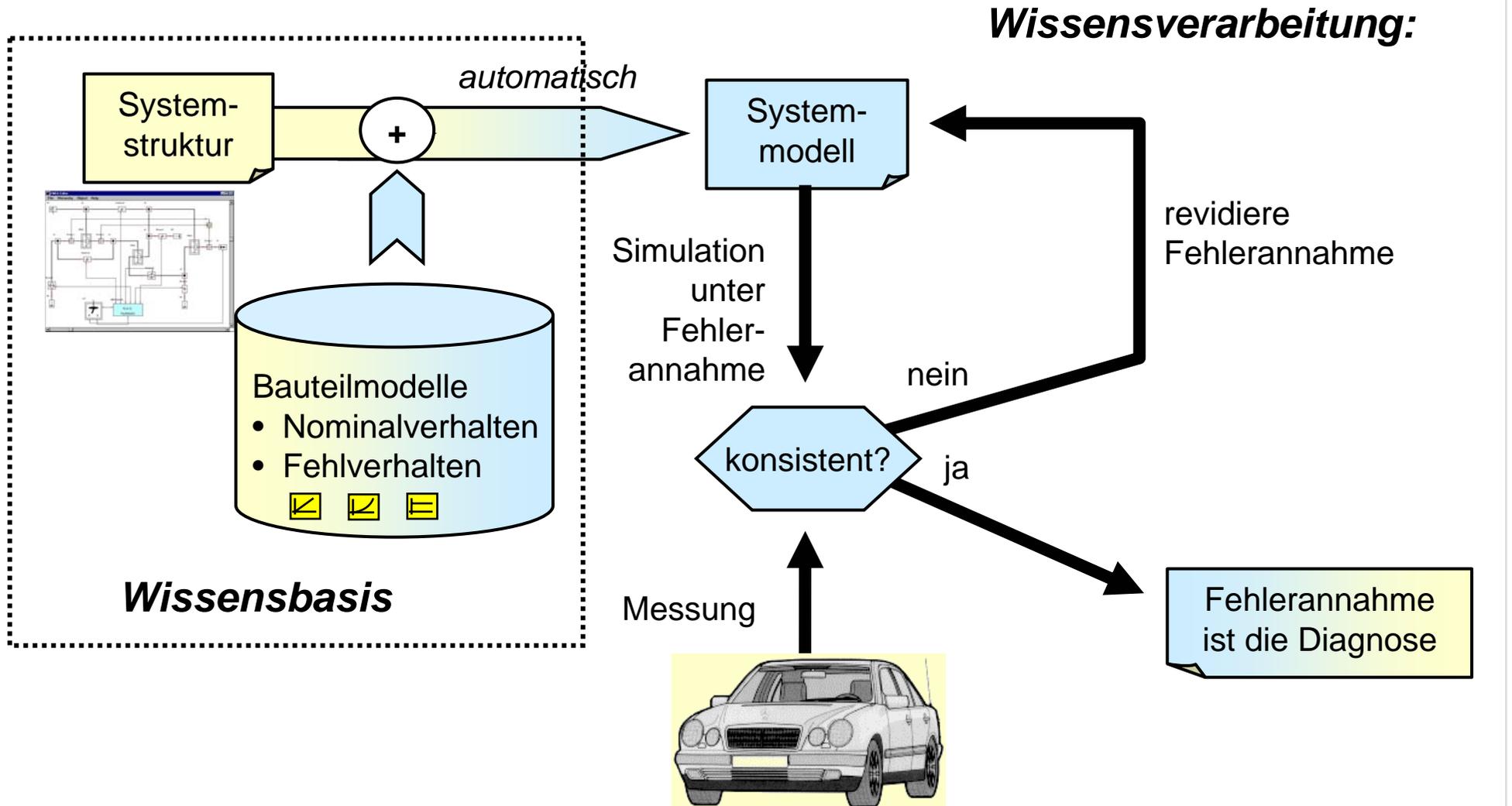
ca.1985: Diagnose = modellbasiertes Schließen

Ideen:

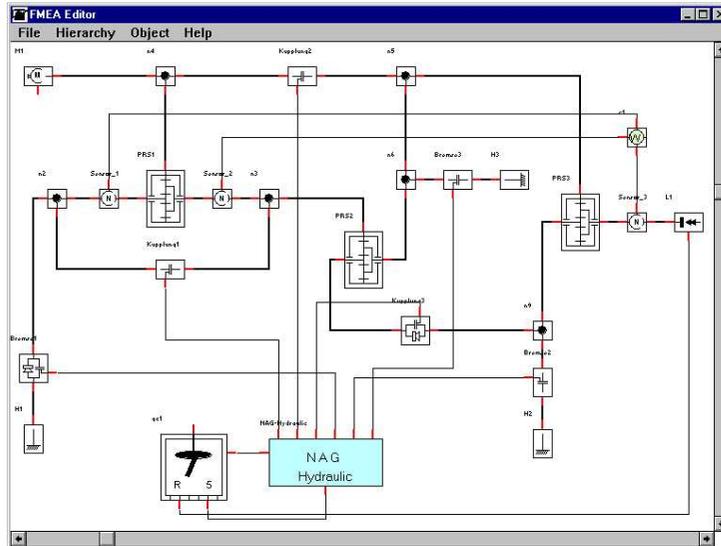
- Erfahrungswissen \Rightarrow Modell des zu diagnostierenden Systems
- objektives Modell, unabhängig von Diagnoseaufgabe
- löst viele der Probleme des heuristischen Ansatzes
- Schlagworte
 - Second Generation XPS
 - tiefes Wissen
 - Schliessen 'from first principles'
 - modellbasiert

Anwendungsgebiet: Technische Diagnose

1980er: Diagnose = modellbasiertes Schließen



Wissenserwerb bei der modellbasierten Diagnose



Systemstruktur:

Welche Komponenten von welchem Typ sind wie miteinander verbunden ?

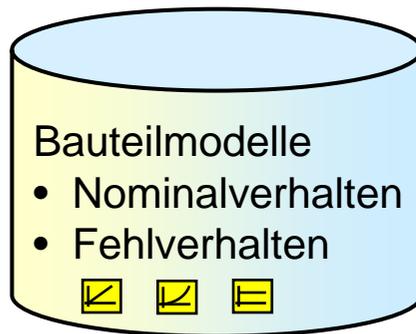
→ erhältlich aus CAD-Daten

Komponentenmodelle:

Wie ist die Abhängigkeit zwischen den Werten, die an den Verbindungspunkten einer Komponente anliegen ?

→ pro Komponententyp einmal zu modellieren

→ Modell ist wiederverwendbar für alle Systeme, in denen Komponenten dieses Typs enthalten sind



Die GDE:

1987: *Der Prototyp für die modellbasierte Diagnose*

Problem:

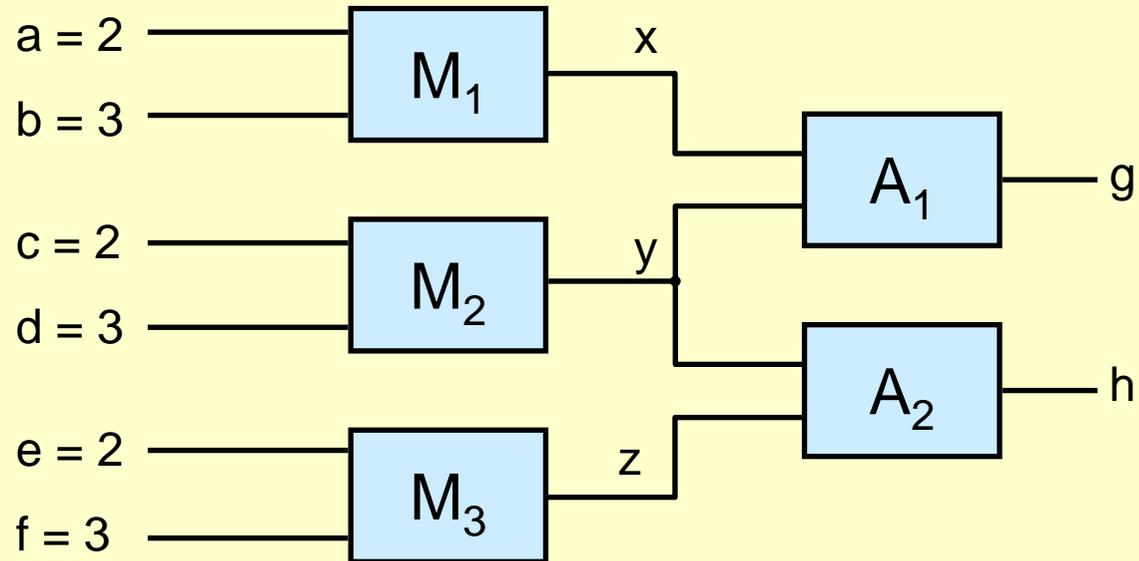
- ‚brute-force‘ Simulation **aller** Fehlerannahmen
kombinatorisch nicht realisierbar

Idee: General Diagnostic Engine GDE, deKleer & Williams 1987

- intelligente Suche im Raum der möglichen Fehlerannahmen
- nutzt inkonsistente Annahmen zum Verkleinern des Suchraums
- Prinzip: konfliktgesteuerte Suche

GDE - Beispiel

Systemstruktur



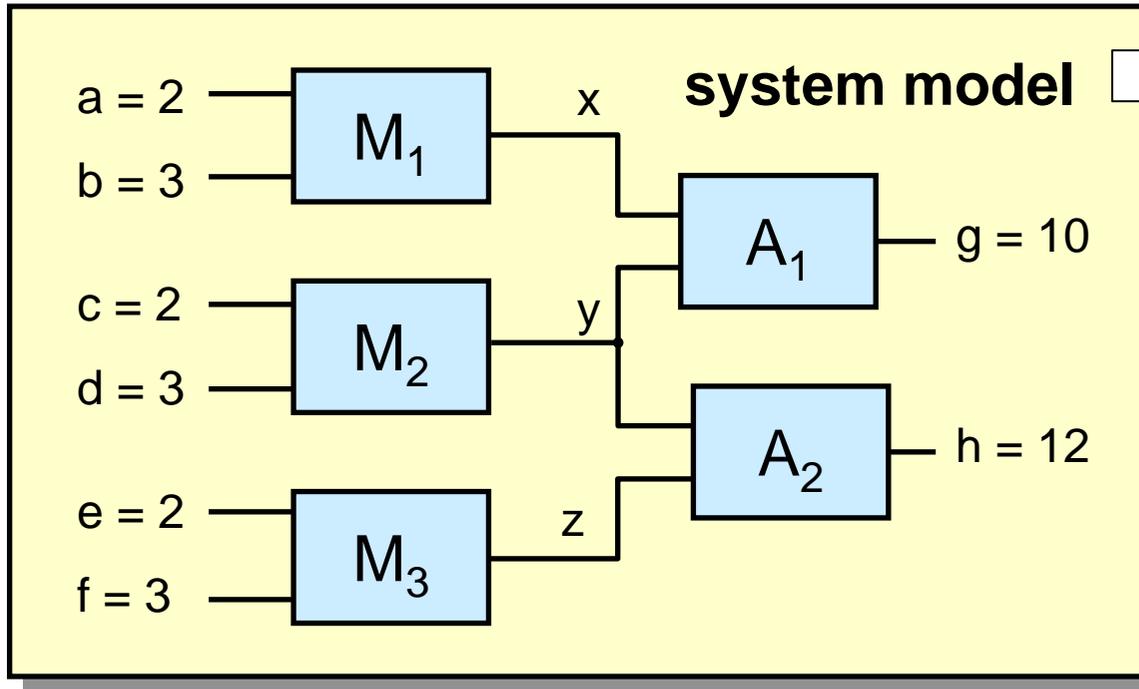
Komponentenmodelle

- Multiplizierer: $\text{mode=ok} \Rightarrow \text{out} = \text{in}_1 * \text{in}_2$
- Addierer: $\text{mode=ok} \Rightarrow \text{out} = \text{in}_1 + \text{in}_2$

Messungen

$$g = 10 \wedge h = 12$$

GDE - Beispiel



simulation

$x = 6$ {M1}

$y = 6$ {M2}

$z = 6$ {M3}

$g = 12$ {M1 M2 A1}, $g = 10$

$y = 4$ {M1 A1}

$h = 10$ {M1 A1 A2 M3}, $h = 12$

$y = 6$ {A2 M3}

two conflicts

diagnoses:

single-fault **M1**

single-fault **A1**

double fault **M2 M3**

:

M1	M2	M3	A1	A2
X	X		X	
X		X	X	X

Modellbasierte Diagnose: Entwicklung seit 1987

GDE - deKleer & Williams 87

Sherlock - deKleer & Williams 89
explicit fault models, probabilities

GDE+ - Struss & Dressler 89
explicit fault models, ordering

roδon
R.O.S.E. Informatik

many years of research ...

- speed-up: focusing techniques
- modeling assumptions, model switching
- dynamics, e.g. finite-state machine models
- real applications: power networks, vehicles, ...

Raz'r 1.0, 1997
commercial
product (OCC'M)

RA - Williams et al. 98
Remote Agent flight software
NASA AMES & JPL
CommonLisp

MDS 1.5, 2000
Daimler Research
Smalltalk

Modellbasierte Diagnose

Stärken (gegenüber dem heuristischen Ansatz):

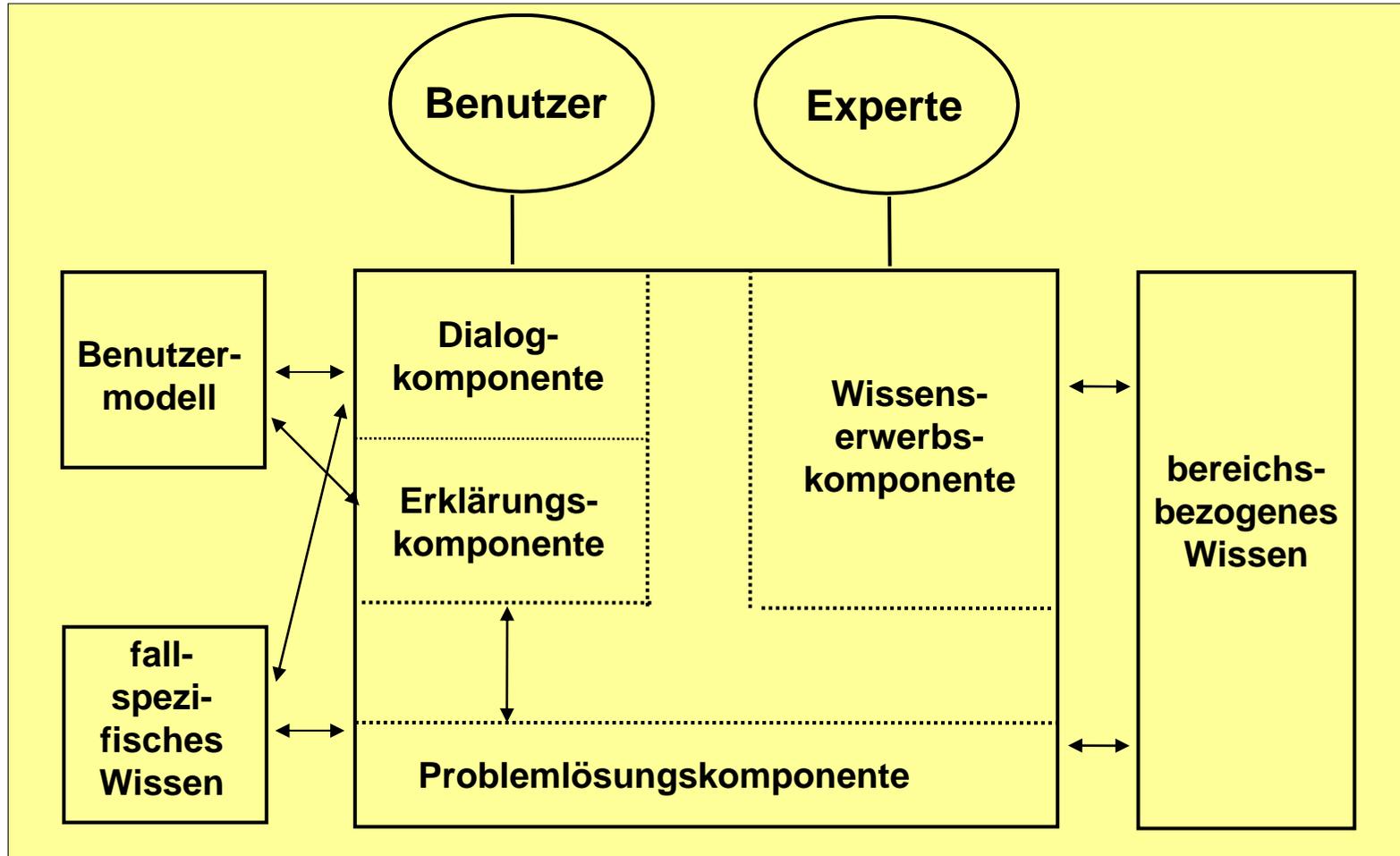
- wiederverwendbare Bauteilmodelle
- Aufbau und Wartung eines Systemmodells: einfach
- Systemmodell: objektiv, nachprüfbar, lokal modifizierbar
- neue Systeme auch ohne Erfahrung diagnostizierbar
- Mehrfachfehler und dynamische Systeme
- nachvollziehbare Erklärung von Diagnosen

Schwächen:

- Aufbau wiederverwendbarer Bauteilmodell-Bibliotheken: schwierig, verborgene Modellierungsannahmen
- rechenintensiv
- bisher wenige Werkzeuge und reale Anwendungen verfügbar

Basistechnologie: Wissensbasiertes System

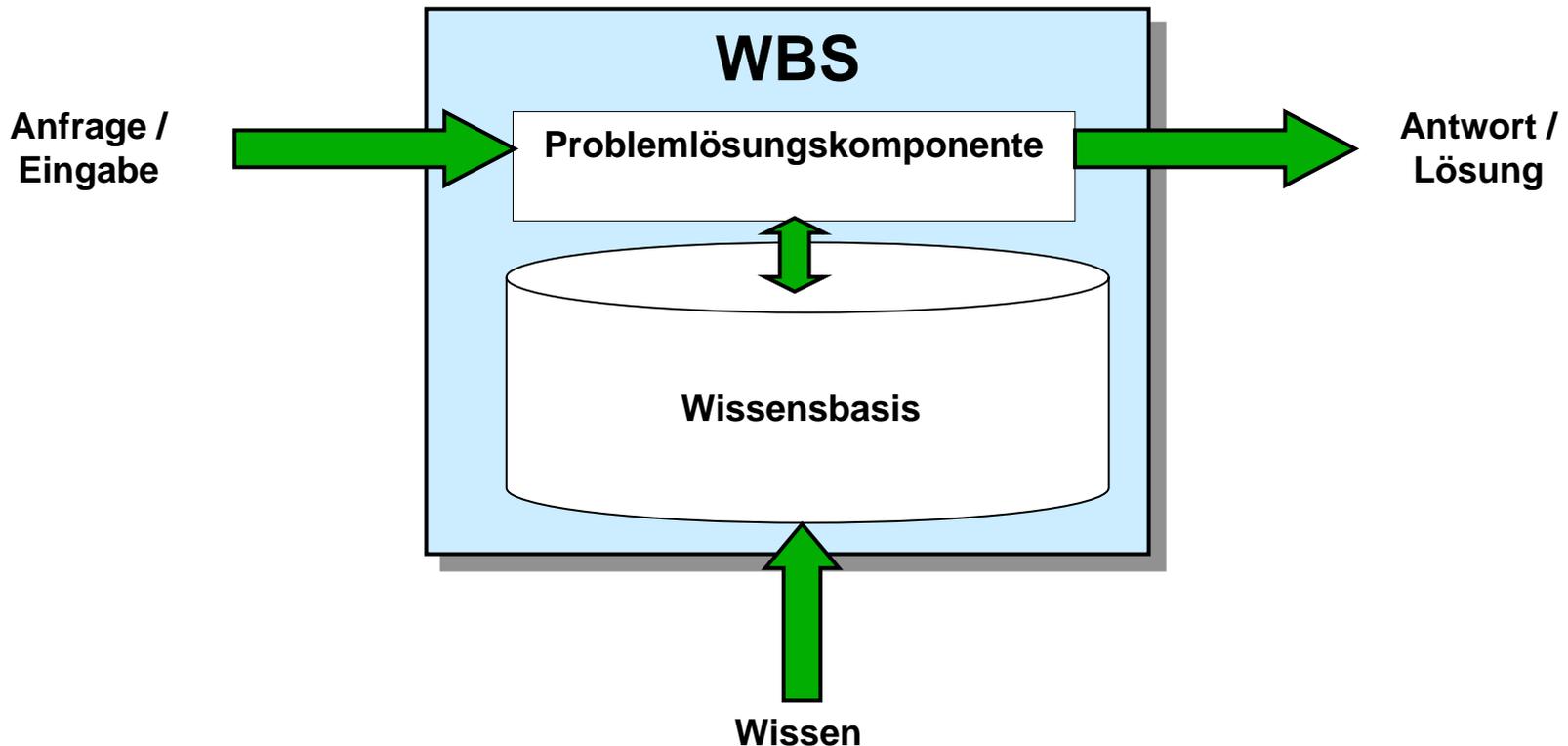
Architektur XPS (klassisch)



Basistechnologie: Wissensbasiertes System

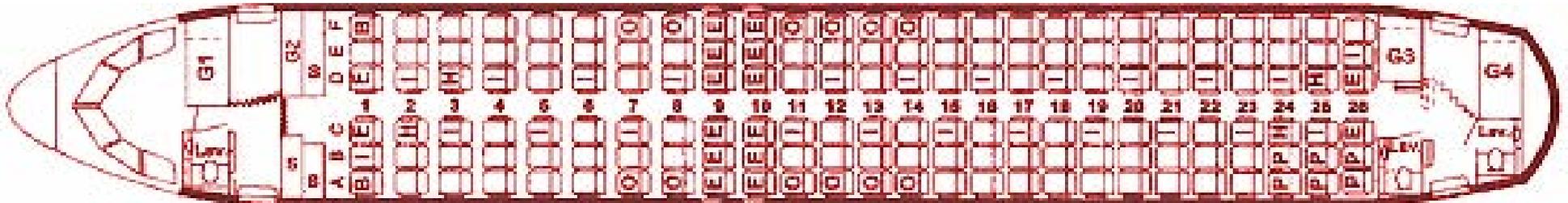


Architektur WBS (allgemeiner)



Anwendungsgebiet: Konfiguration

Kabinenlayout für Passagierflugzeuge



Platzierung der Kabineneinrichtung (Sitze, Küchen, Toiletten, etc.) unter Berücksichtigung von:

- Kundenwünschen
- Technischen Möglichkeiten
- Legalen Beschränkungen
- Optimalitätskriterien

Grundlage der KI: Logik

Intelligente Lebewesen können auch sehr allgemeines Wissen verarbeiten:
Je allgemeiner, desto intelligenter

Allgemeine Verarbeitungsfähigkeiten benötigen allgemeine
Beschreibungsmöglichkeiten für die Daten und Verarbeitungsregeln

Die allgemeinste objektive Beschreibungssprache
ist die Sprache der mathematischen Logik.

**Daher arbeiten traditionelle KI-Verfahren mit logischen
Beschreibungssprachen.**

- Probleme:**
- **Aufgaben liegen häufig anders formuliert vor.**
 - **Allgemeinheit geht auf Kosten der Effizienz.**

Ziele und Methoden der KI

Ziele für die SW-Lösungen

- **Allgemeinheit**
- **Flexibilität, Erweiterbarkeit**
- **Erklärbarkeit der Antworten**

Häufig eingesetzte Methoden

- **Logische Programmiersprachen (PROLOG)**
- **Objektorientierte Programmiersprachen (Smalltalk)**
- **Funktionale Programmiersprachen (Lisp)**
- **Verteilte Systeme (Neuronale Netze, Multiagentensysteme)**
- **Begriffswelten (Ontologien)**

Definitionen von KI

Systeme, die wie Menschen denken	Systeme, die rational denken
<p>„Die aufregende und neuartige Anstrengung, Computern das Denken beizubringen, ... KI will die Sache selbst: Maschinen mit Verstand, im vollen und wörtlichen Sinne.“ (Haugeland, 1985)</p> <p>„[Die Automatisierung von] Aktivitäten, die wir dem menschlichen Denken zuordnen, Aktivitäten wie beispielsweise Entscheidungsfindung, Problemlösung, Lernen ...“ (Bellman, 1978)</p>	<p>„Die Studie mentaler Fähigkeiten durch die Nutzung programmier-technischer Modelle.“ (Charniak und McDermott, 1985)</p> <p>„Die Studie der Programmtechniken, die es ermöglichen, wahrzunehmen, logisch zu schließen und zu agieren.“ (Winston, 1992)</p>
Systeme, die wie Menschen agieren	Systeme, die rational agieren
<p>„Die Kunst, Maschinen zu schaffen, die Funktionen erfüllen, die, werden sie von Menschen ausgeführt, der Intelligenz bedürfen.“ (Kurzweil, 1990)</p> <p>„Die Studie, wie man Computer dazu bringt, Dinge zu tun, bei denen ihnen momentan der Mensch noch überlegen ist.“ (Rich und Knight, 1991)</p>	<p>„Computerintelligenz ist die Studie des Entwurfs intelligenter Agenten.“ (Poole et al., 1998)</p> <p>„KI ... beschäftigt sich mit intelligentem Verhalten in künstlichen Maschinen.“ (Nilsson, 1998)</p>

Definitionen aus Russell / Norvig

Definitionen von KI

KI beschäftigt sich mit Problemen, die

- **in der Praxis relevant sind.**
- **häufig nicht exakt spezifiziert werden können.**
- **NP-vollständig sind, wenn sie exakt spezifiziert werden können.**

Definition iw

Zusammenfassung Kapitel 1

Anwendungsgebiete der KI:

- **Spiele, in denen die Maschine andere Spieler simuliert**
 - **Rundenbasierte Strategiespiele**
 - **Echtzeit-Strategiespiele**
- **Resourceverteilung**
 - **Streckenbelegungsplan in Eisenbahnnetzen**
- **Optimierungsprobleme mit dynamischen Parametern**
 - **Verkehrsnavigation**
- **Kundenberatung**
 - **Touristeninformationssystem**
- **Diagnose**
 - **Medizinische Diagnose**
 - **Technische Diagnose**

Zusammenfassung Kapitel 1

Basistechnologien der KI:

- **Agentenorientierte Software**
 - verteilt
 - autonom
 - proaktiv
- **Wissensbasierte Systeme**
 - Trennung in Wissen und Inferenzmaschine
 - Intelligenter Wissenserwerb und Wissensrepräsentation
 - Hauptfokus: Wiederverwendung

Zusammenfassung Kapitel 1

**Der klassische Gegensatz von verschiedenen
Forschergemeinden in der Informatik:**

KI vs. Algorithmik

- **flexibel**
- **kundenorientiert**
- **exakt**
- **effizient**

Das muss kein Widerspruch sein!

Literatur

KI allgemein:



Günter Görz / Claus-Rainer Rollinger / Josef Schneeberger: *Handbuch der Künstlichen Intelligenz*
Oldenbourg 2000 (3. Auflage), ISBN 3-486-25049-3



Stuart Russell / Peter Norvig: *Artificial Intelligence: A Modern Approach*
Pearson 2003 (2. Auflage), ISBN 0-13-080302-2