

# ***Anwendungen der Künstlichen Intelligenz***

Sebastian Iwanowski  
FH Wedel

**Kap. 1:**  
Einführung und Überblick

# Vorlesungsüberblick

## Inhaltliche Voraussetzungen:

Diskrete Mathematik, Programmieren I und II

*hilfreich: Objektorientierte Programmiertechnik*

## Lernziele dieser Vorlesung:

Begeisterung für KI-Anwendungen und KI-Technologie

Kenntnis mehrerer Anwendungsgebiete von KI

Grobes Verständnis einiger Basistechnologien der KI

*Welche Anwendungsgebiete und Basistechnologien das sind,  
wird im Laufe dieser Einführung vorgestellt.*

# Was ist KI ?

## Der Turing-Test



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

# **Anwendung: Medizinische Diagnose**

## **Psychoanalyse: Eliza**      1966: Joseph Weizenbaum, MIT

**Computer nimmt psychotherapeutische Sitzung vor und reagiert so, „wie man sich das von einem Psychotherapeuten erwartet“.**

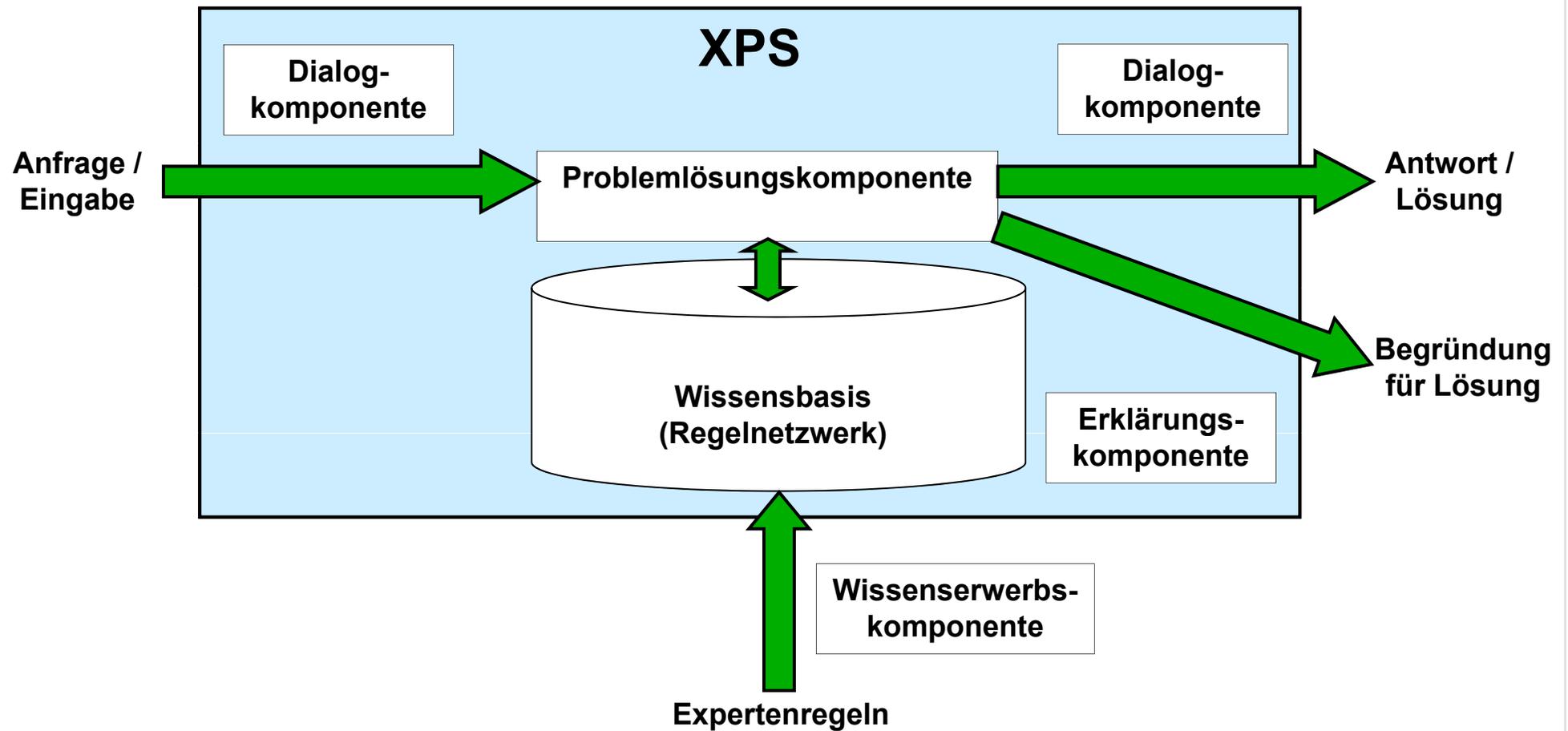
- **bestand bei vielen Menschen den Turing-Test**
- **eingebauter Sprachassembler und -composer**
- **Reaktionsregeln**

## **Medizinische Diagnose: Mycin**      1972: University of Stanford

- **zur Diagnose und Therapie von Infektionskrankheiten**
- **arbeitete mit probabilistischen Regeln**
- **lieferte gute Trefferquoten**
- **wenig Akzeptanz bei Ärzten wegen Misstrauen gegen Computer**

# Basistechnologie: Expertensystem

## Architektur Expertensystem



# Anwendung: Technische Diagnose

## Was ist Technische Diagnose?

### Gegeben:

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

### Aufgabe:

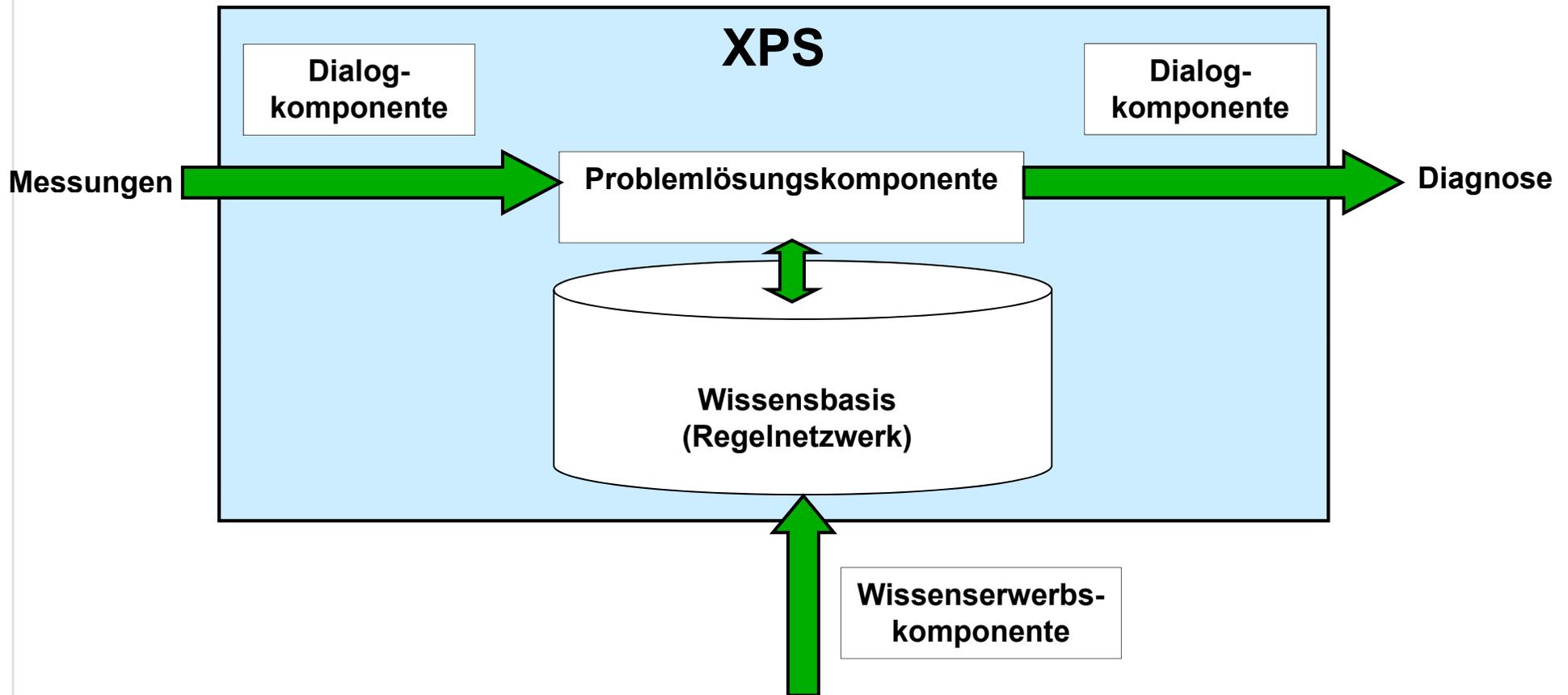
Bestimme,

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



# Anwendung: Technische Diagnose

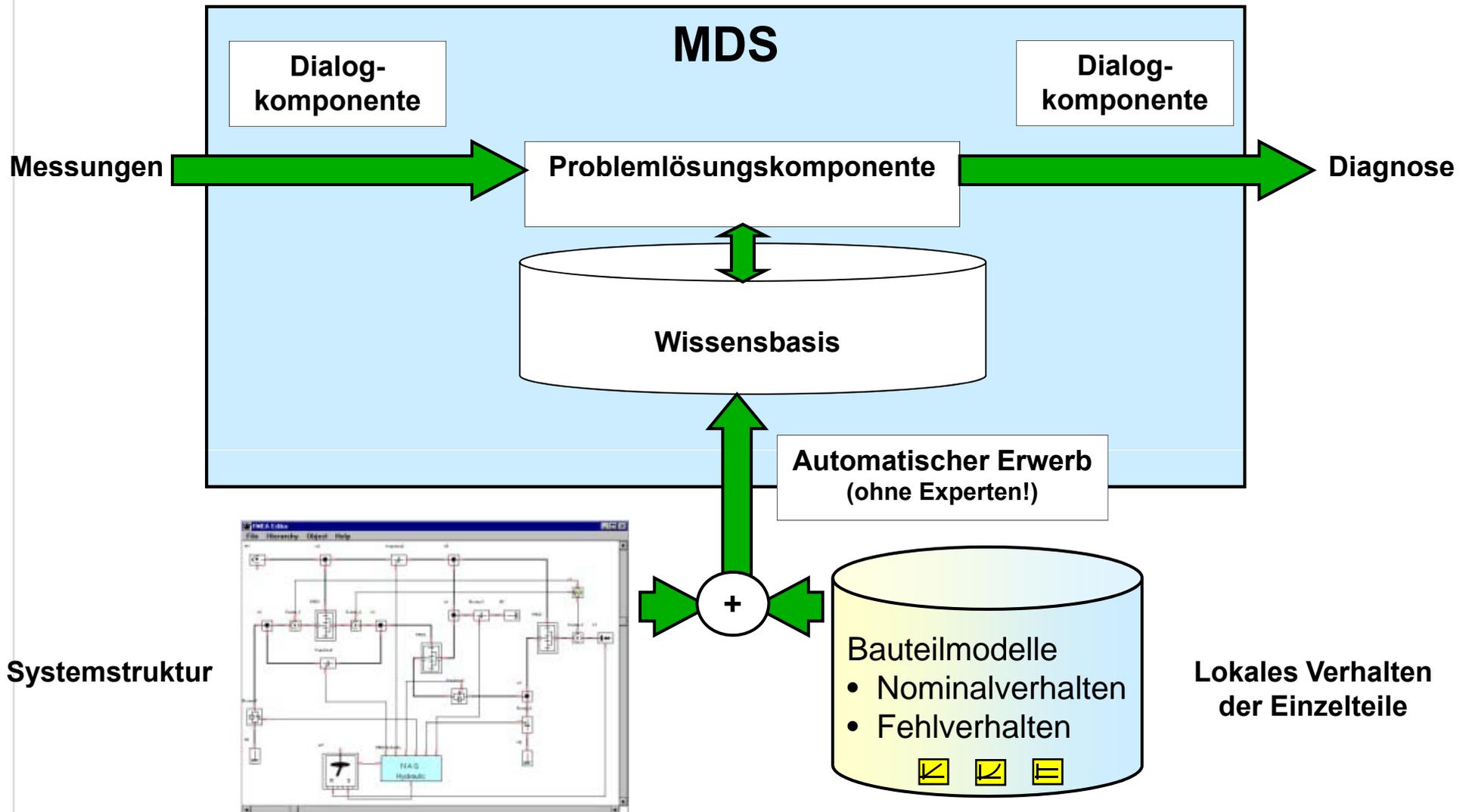
1970er: Diagnose = heuristische Klassifikation



**Bauplan- und Funktionswissen:  
Konkretes Symptom  $\Rightarrow$  konkrete Diagnose**

# Anwendung: Technische Diagnose

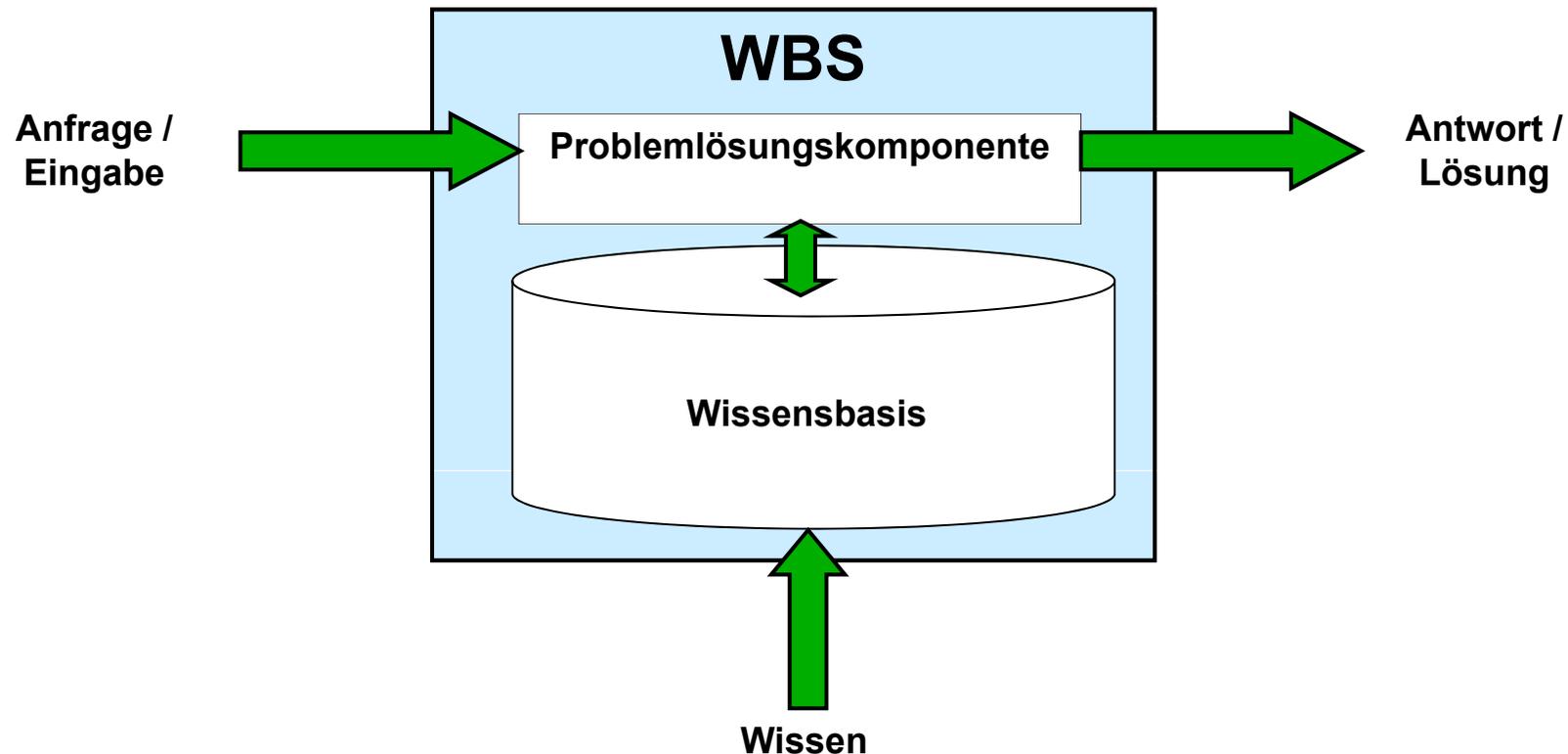
1980er: Diagnose = modellbasiertes Schließen



# Basistechnologie: Wissensbasiertes System



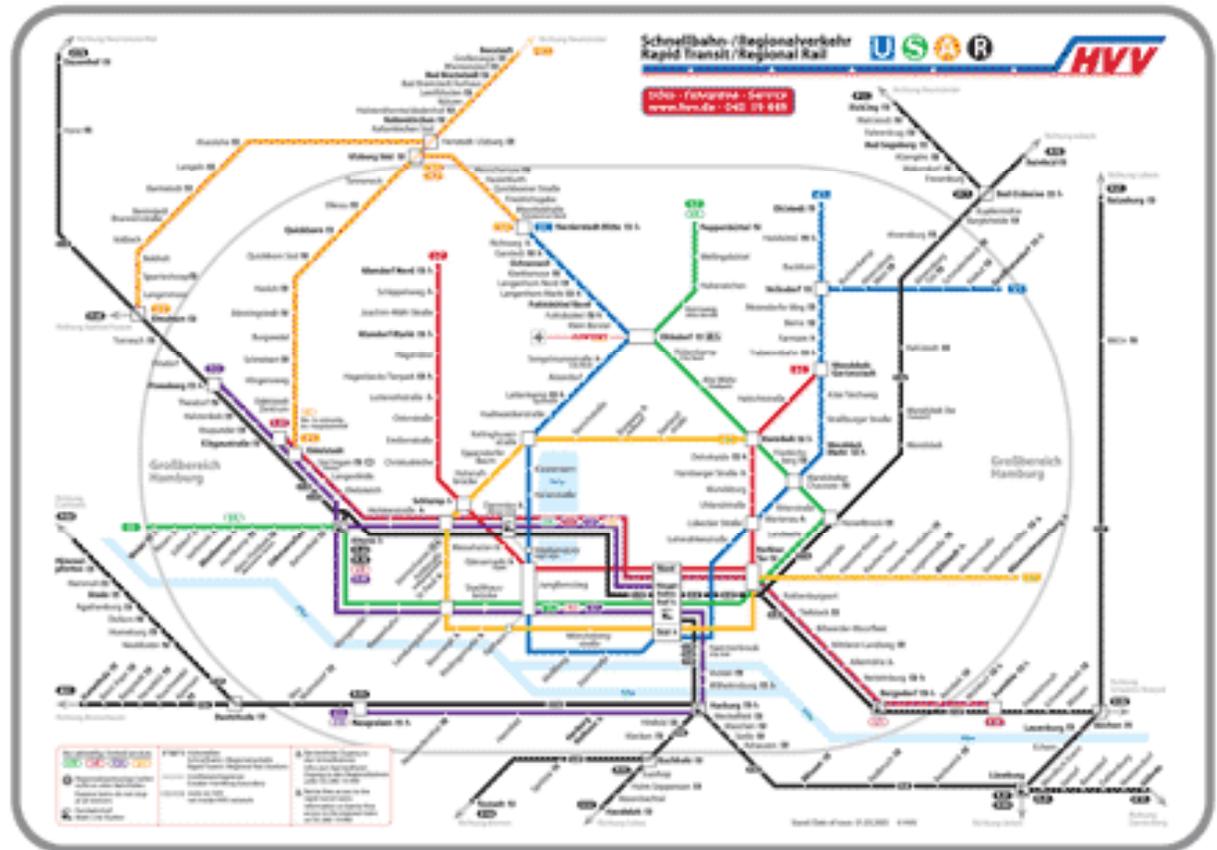
## Architektur WBS (allgemeiner)



# Anwendung: Fahrgastinformationssystem

## Aufgabe:

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



## Lösung:

### Algorithmus von Dijkstra

(siehe Vorlesung Diskrete Mathematik, Kap. 6, Graphentheorie)

### A\*-Algorithmus

Optimierungen durch weitere Heuristiken (z.B. Geofox für HVV)

Optimierungen durch preprocessing (z.B. Hafas für Deutsche Bahn)

Weitere Infos: Seminarvortrag und Ausarbeitung von Stefan Görlich, SS 2005, Nr. 5

# Anwendung: Fahrgastinformationssystem

## Dynamische Fahrgastinformation

### Datengrundlage:

- Statische Information: Fahrplan
- Erweitert um aktuelle Verbindungslage



Linie	Ziel	Abfahrt in
2	S Hackescher Markt	1 min
6	U Schwartzkopffstr.	6 min
4	S Hackescher Markt	10 min
3	S Hackescher Markt	14 min
5	S Hackescher Markt	15 min

Spandauer Straße

### Information bei kurzfristigen Abweichungen

- Ursache für Abweichungen
- Abweichungsumfang
- Ersatzrouten

### Dynamische Fahrgastinformationsmedien

- Lautsprecherdurchsagen
- Dynamische Anzeigegeräte
- vor allem aber: Smartphones



# **Anwendung: Fahrgastinformationssystem**

## **Personalisierte Dynamische Fahrgastinformation (PDFIS)**

*Masterarbeit von Michael Schiefenhövel (WS 2005/2006)*

Filterung von Informationen für den Fahrgast

- nur die Informationen, die er benötigt
- keine verwirrenden Mehrinformationen
- sowohl statische als auch dynamische FG-Informationen

Erstellung von Mehrwertdiensten

- Routenberechnung aktualisiert an konkrete Lage
- Touristeninformation

Personalisierte Fahrgastinformationsmedien

- Persönliche Auskunft (über Handy, etc.)
- Multimediaterninals

# **Anwendung: Fahrgastinformationssystem**

## **Fahrgastinformation für den HVV über Smartphones:**

Entwicklung und Implementierung konkreter Prototypen:

- iPhone
- Android-Handy

### ***Diplomarbeit Sebastian Hammes (eos-uptrade, SS 2010)***

- Ergebnisse flossen ein in HVV-App

### ***Bachelorarbeit Henning Reimer (HBT, SS 2010)***

- Ergebnisse flossen ein in Geofox-App

### ***Masterarbeit Josias Polchau (HBT, SS 2014)***

- Innovationspreis des Rotary Clubs Wedel

# Anwendung: Fahrgastinformationssystem

## Mobiler Fahrgastassistent im ÖPNV:

*enthält keine KI  
im klassischen Sinn*

Ein „Navigationsgerät“ für den ÖPNV

*Masterarbeit von Josias Polchau (SS 2014)*

Funktion	Nutzen	Realisierungsaufwand	
Informationen zur aktuellen Fahrt	██████████	██████████	realisiert
Aktualisierung der Route	██████████	██████████	realisiert
Erinnerungen	██████	███	realisiert
Verbesserung der Fußwegnavigation	██████████	██████████████████	
Füllstands-Anzeige	██████	██████████████	
Stau-Karte	██	██████	realisiert
Anzeige: Einstieg vorne/hinten	██████	derzeit nicht möglich	
Fahrradmitnahme	██████████	derzeit nicht möglich	
Routenpause	██████	██████████████	
Lautsprecheransagen	██████	derzeit nicht möglich	

- soll in Geofox-App eingebunden werden

# **Anwendung: Straßennavigation**

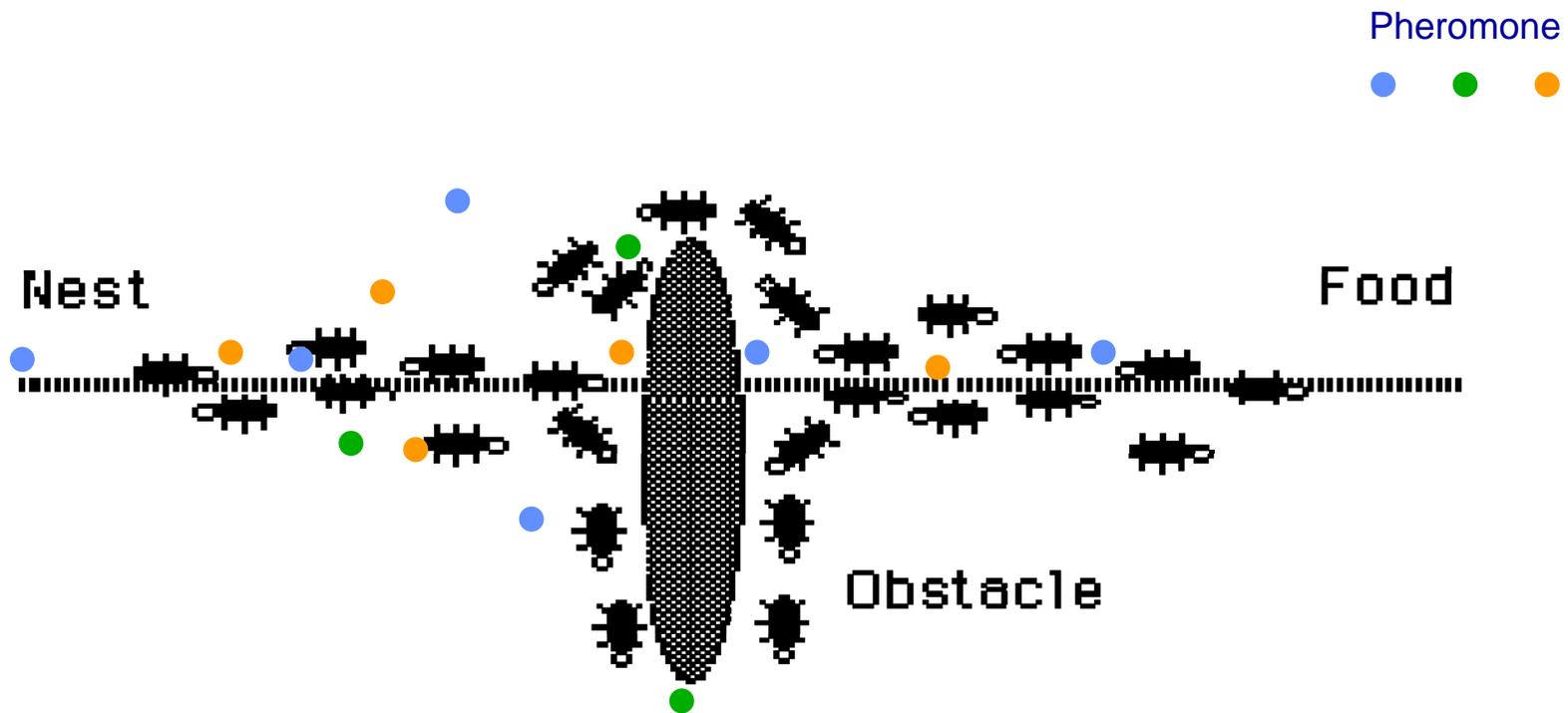
## **Unterschiede bei Übertragung auf Straßenverkehr:**

- **Streckennetz ist wesentlich dichter**
- **keine Fahrpläne bzw. Öffnungszeiten**
- **Fahrzeiten hängen sehr stark von Belegungsdichte ab**
- **Verkehrsträger werden nicht zentral gesteuert**

# Anwendung: Straßennavigation

## Schwarmintelligenz: Pheromonbasierter Ansatz

### Ameisen auf Futtersuche



# Anwendung: Straßennavigation

## Schwarmintelligenz: Pheromonbasierter Ansatz

### Analogon: Autos auf Routensuche



# **Basistechnologie: Schwarmintelligenz**

- **viele kleine autonome Einheiten, die einzeln nicht viel können**
- **Gesamtorganismus kann mehr als die Summe der Einheiten  
(emergentes Verhalten)**
- **festes Regelwerk für Gesamtorganismus**
- **anytime-Eigenschaft**

**Entwicklungs- und Forschungsschwerpunkt an der FH Wedel unter iw:  
Mehrere Softwareprojekte, Abschlussarbeiten und Veröffentlichungen seit 2006**

# Anwendung: Spiele-KI

**Schachcomputer** (Bsp. für rundenbasiertes Spiel)

**Meilenstein 1997:**

Kasparov **2.5** – Deep Blue **3.5**



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



# Anwendung: Spiele-KI

Fang den Fox - Mozilla Firefox

http://catchfox.hbt.de/fox.html

Meistbesuchte Seiten Windows Windows Media Kostenlose Hotmail Links anpassen FTP-Tutorial by Fabian... Recursion Software, In... Familienbilder2006

Suche Lesezeichen Rechtschreibprüfung Übersetzen Senden an Einstellungen

**Fang den FOX mit GEOFOX**

Spiel Fänger Ansicht Hilfe Freitag, 12.36 Uhr

**Figuren**

**FOX**

Die letzte bekannte Position des FOX ist:  
**Eppendorfer Baum**

Diese Figur hat bereits diese Runde gezogen.

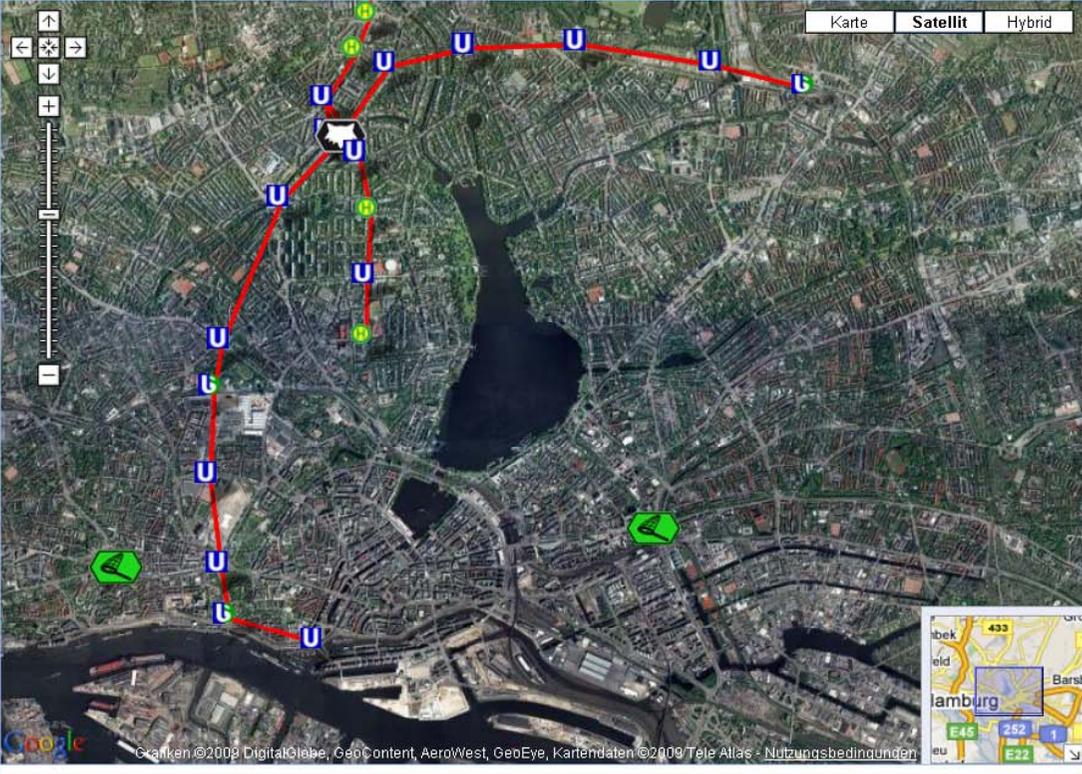
**Fänger 1**

**Fänger 2**

**Fänger 3**

**Fänger 4**

**Fänger 5**



Karte Satellit Hybrid

Copyright ©2008 DigitalGlobe, GeoContent, AeroWest, GeoEye, Kartendaten ©2009 Tele Atlas - Nutzungsbedingungen

**Status**

For informational stuff, maybe status bar  
Begin loading stations: 11:35:20  
Data received: 11:41:07  
StartindStations made: 11:41:97

Fertig

## Anwendung: Spiele-KI

### Rundenbasiertes Spiel „Fang den Fox“

- Diplomarbeit 2009 bei HBT (Betreiber von Geofox)
- 3. Preis der Hochbahn
- Computer steuert den „Fox“,  
der von menschlichen Spielern gefangen werden soll
- Spiel benutzt Echtzeitinfos des HVV
  
- ursprünglich für GoogleMaps erstellt, dann auf Lizenzkarte umgestellt
- Leider musste die Karte aus lizenzrechtlichen Gründen abgeschaltet werden.
- Neuimplementierung nur auf Basis von OpenStreetMap möglich

# Anwendung: Spiele-KI

## Beispiele für Echtzeit-Strategiespiele:



**Half Life**



**Command + Conquer 3**

# Anwendung: 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

# Basistechnologie: Suchstrategien

- **Konstruktion von Suchräumen**
- **Uninformierte Suchstrategien**
  - **Breitensuche**
  - **Tiefensuche**
  - **kombinierte Suche** → Spezialfall: Dijkstra-Algorithmus
- **Informierte Suchstrategien**
  - Spezialfall: A\*-Algorithmus

***wird auch verwendet in Navigationssoftware-Produkten***

# Anwendung: Spiele-KI

## Echtzeit-Strategiespiele

### Anforderung in modernen Spielen:

- Wegfindung und Terrainanalyse in Umgebungen, die sich dynamisch verändern

### Algorithmische Techniken dafür:

- Berechnung des Navigationsgraphen
- Lernen aus suboptimalen Wegen
- Arbeiten mit unsicheren Informationen

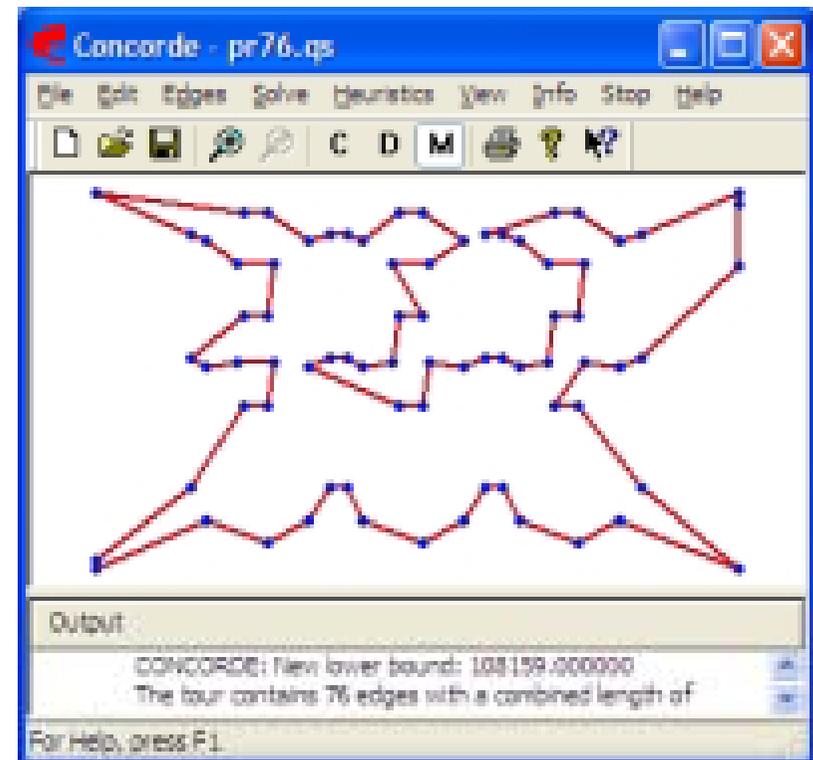
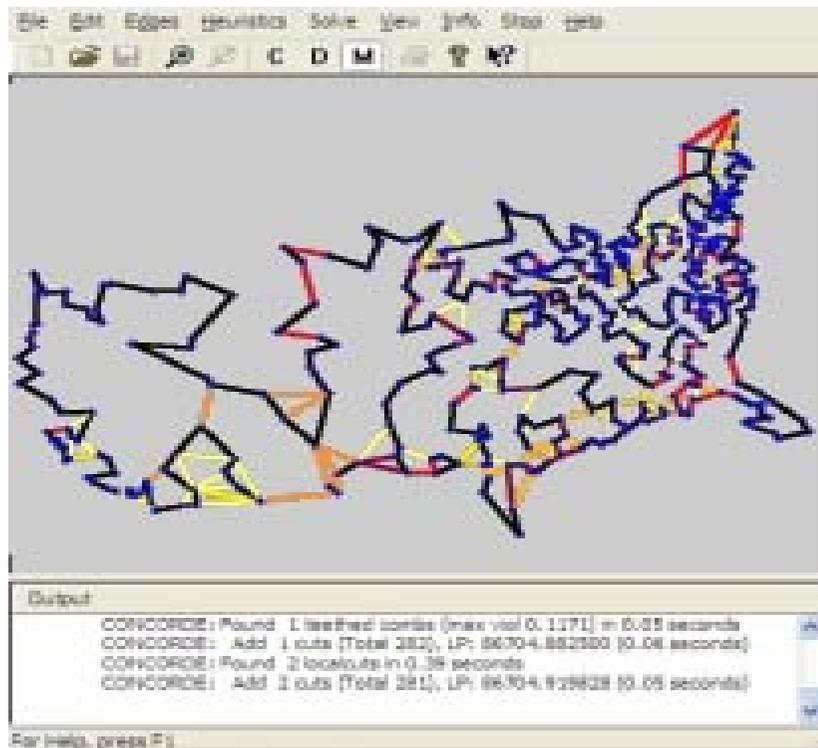
*eigentlich keine KI  
im klassischen Sinn*

*aber „Spiele-KI“ !*

# Anwendung: Das Traveling Salesman Problem (TSP)

## Beispiel für ein NP-vollständiges Problem:

Finde zu einer gegebenen Menge von Städten mit gegebenen Entfernungen die kürzeste Rundreise, die jede Stadt genau einmal durchquert.



Quelle: <http://www.tsp.gatech.edu//index.html>

# **Anwendung: Das Traveling Salesman Problem (TSP)**

## **Verallgemeinerung in der Logistik:**

- **Beachtung von zeitlichen Beschränkungen (Zeitfenster)**
- **Beachtung von Ladekapazitäten bei Auslieferungen**
- **Weitere systemspezifische Anforderungen**

## **Beispiel für Abschlussarbeiten in einer Firma:**

**implico: Tourenplanungsproblem in Öl- und Gasbranche (SS 2010, SS 2011, SS 2013)**

## **Langzeit-Entwicklungsprojekt: Touristen-Infomations-System**

**Christoph Forster / Thomas Kresalek / Felix Döppers:  
Masterprojekt Hamburger Touristen Information (ab 2009)**

<http://vsrv-studprojekt2.fh-wedel.de:8080/touristinformationsystem/home>

## **Lösung dynamischer Problemstellungen durch Ameisensysteme**

### **Beispiel einer Abschlussarbeit in einer Firma:**

**Christopher Blöcker: Dynamische Optimierung von Liefertouren mit einem Ameisen-System, SS 2011**

# **Anwendung: Das Stundenplanproblem (Scheduling)**

**Gegeben endliche Mengen Fächer, Räume, T(Zeiten)**

**Aufgabe: Generierung einer injektiven Funktion  $F \rightarrow R \times T$**

**Nebenbedingungen:**

- **Bestimmte Fächer dürfen nicht zur selben Zeit stattfinden**
- **Nicht jedes Fach darf zu jeder Zeit stattfinden**
- **Nicht jedes Fach darf in jedem Raum stattfinden**

**Weiche Kriterien (dürfen verletzt werden):**

- **Bestimmte Fächer sollen zu bestimmten Zeiten möglichst nicht stattfinden**
- **Bestimmte Fächer sollten möglichst hintereinander stattfinden**
- **Bestimmte Fächer sollten möglichst nicht am selben Tag stattfinden**

**Optimierungsfunktion:**

- **Möglichst wenige Verletzungen von weichen Kriterien**
- **Möglichst wenige Freistunden für Studiengänge**
- **Möglichst gleichmäßige Verteilung auf verschiedene Tage für ...**

# Basistechnologie: Constraint Satisfaction Problem (CSP)

## Spezifikation eines CSP:

- **Variablenmenge**
- **Definitionsbereiche (Domains)**
- **Constraints: Beziehungen zwischen den Variablen (harte und weiche)**  
(in der Regel Gleichungen oder Ungleichungen)
- **Optimierungskriterium**  
(in der Regel Funktion der Variablen, die minimiert oder maximiert werden soll)

## gültige Lösung:

Belegung aller Variablen mit Werten, sodass alle harten Constraints erfüllt sind

## optimale Lösung:

gültige Lösung, die das Optimierungskriterium optimiert

***Vielfältige Anwendungsmöglichkeiten in Fragestellungen der Logistik***

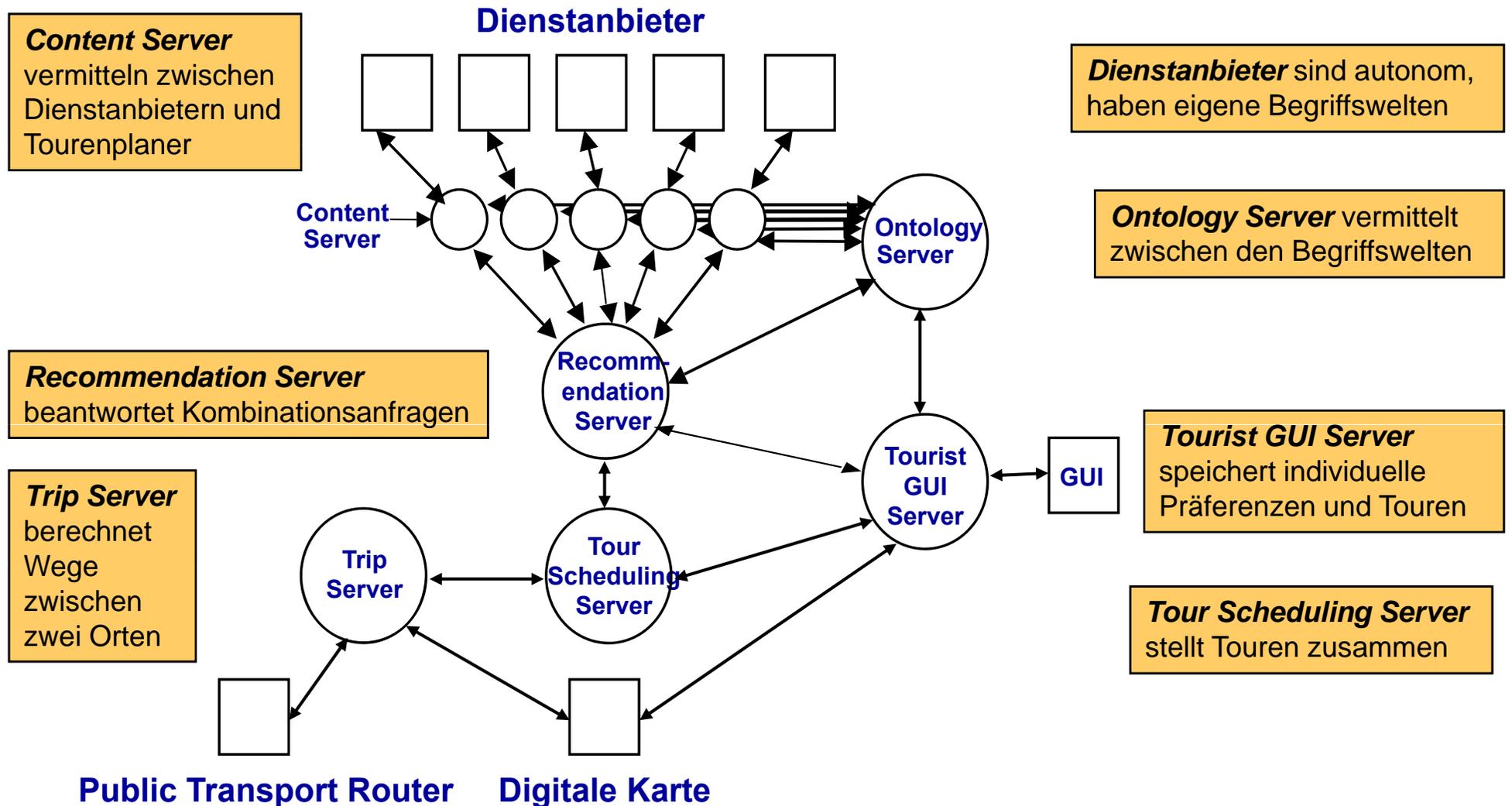
# Anwendung: Touristeninformationssystem

## Forderungen:

- Tourist hat die endgültige Kontrolle
- Dienstanbieter ist autonom und trägt die Verantwortung für die Informationen
- Unabhängiges Makeln zwischen verschiedenen Anbietern
- Flexibilität gegenüber Anforderungsänderungen, sogar während der Tour
- Fehlertoleranz gegenüber Ausfall von Dienstanbietern
- **Beliebige** Dienstanbieter sollen während des Systembetriebs automatisch eingeklinkt und ausgeklinkt werden können.

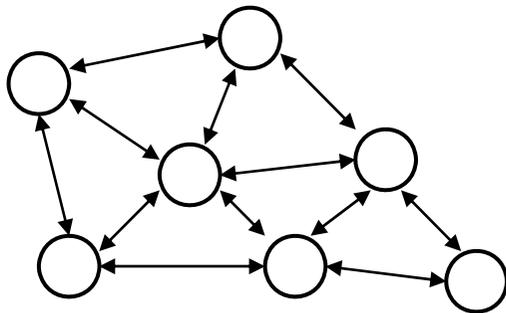
# Anwendung: Touristeninformationssystem

## Architektur des Tourenplaners: Prototyp einer SOA

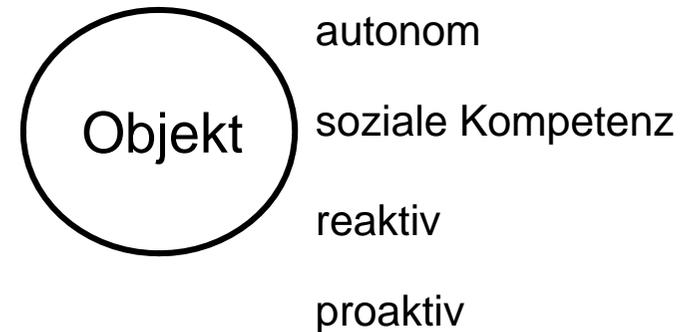


# Basistechnologie: Agentenorientierte Software

Multiagentensystem:



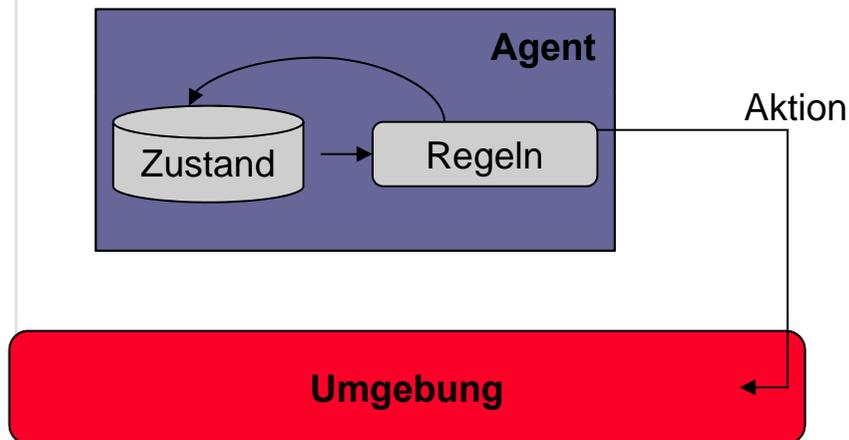
Softwareagent:



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**

# Basistechnologie: Agentenorientierte Software

## Agenteigenschaft: Proaktivität (Zielgerichtetheit)



Agenten reagieren nicht nur auf Reize der Umgebung, sondern besitzen internen Zustand und sind zu zielgerichtetem Planung und Handeln fähig.

=> Sie ergreifen die **Initiative**

*„The difference between an automation and an agent is a somewhat like the difference between a dog and a butler. If you send your dog to buy a copy of the New York Times every morning, it will come back with its mouth empty if the news stand happens to have run out one day. In contrast, the butler will probably take the **initiative** to buy you a copy of the Washington Post, since he knows, that sometimes you read it instead.“*

**Le Du**

Quelle: Seminarvorlesung zur Agentenorientierten Software von Matthias Rohr, SS 2004, Nr. 4

# Basistechnologie: Semantisches Netzwerk

- **Ontologiemanagement**
- **Beschreibungssprache**
- **Beschreibungslogik**

entwickelt in den 1990er Jahren  
mit Syntaxstandards, die ab ca.  
1985 innerhalb der KI-Gemeinde  
entwickelt wurden.

## Moderne Anpassung (ab 2001): *Semantic-Web-Standards*

Initiator: Tim Berners-Lee

Ontologiemanagement, Beschreibungssprache und Beschreibungslogik  
in XML oder vergleichbaren Standards

### **Wichtig:**

Universell gültige Definitionen in einer maschinen- und browserlesbaren Syntax

# Anwendung: evi.com

## Semantische Antwortmaschine

wie wolframalpha.com, aber auch als ortsbezogene App

The screenshot shows a web browser window with several tabs. The active tab is titled "Evi | Ask me anything". The address bar shows "om" and "CAL". The main content area features the Evi logo, which is a smiley face inside a circle, followed by the text "Evi an amazon company". Below the logo is a search input field containing the text "How tall is the statue of liberty" and a blue question mark button. Underneath the search field is a promotional box for the Evi mobile app. The box contains the text "Say hello to Evi" and a paragraph describing the app's capabilities: "Evi is our best selling mobile app that can answer questions about local knowledge, weather, books, music, films, people and places, recipe ideas, shopping and much more. Over the next few months we will be adding all of Evi's power to this site." Below this text is a link that says "Find out more...". To the right of the text are three buttons for downloading the app: "Available on the App Store", "Available on kindle fire", and "ANDROID APP ON Google play".

Verstehen der Frage mit Hilfe von Computerlinguistik  
und Ableiten der Antwort mi Hilfe von Ontologien

# Basistechnologie: Computerlinguistik

Maschinelle Verarbeitung natürlicher Sprache

Gebiet verbindet das Wissen aus der Informatik und der Linguistik

Teilgebiete

- Phonetik (gesprochenes Wort verstehen)
- Morphologische Prozesse (Rechtschreibkorrektur)
- Grammatische Analysen (Grammatiküberprüfung)
- Statistische Verfahren (Häufigkeitsanalysen)
- Lexikographie (Thesaurus)
- Semantische Analyse

# 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**

# Merkmale klassischer KI-Lösungen

Der klassische Gegensatz von verschiedenen  
Forschergemeinden in der Informatik:

## KI vs. Algorithmik

- flexibel
- kundenzentriert
- exakt
- effizient

**Das muss kein Widerspruch sein!**

# Merkmale klassischer KI-Lösungen

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.**

# **Basistechnologie: Logische Programmiersprache**

- **Input:**  
**Spezifikation des Problems mit logischer Beschreibungssprache**
- **Output:**  
**Antwort in logischer Beschreibungssprache**
- **automatisch:**  
**Generierung des Outputs aus Input**
- **Zur Effizienzverbesserung:**  
**Beeinflussung der Generierung des Outputs aus Input**

# Zusammenfassung und Vorlesungsvorschau

## KI-Ziele für die SW-Lösungen

- **Allgemeinheit**
- **Flexibilität, Erweiterbarkeit**
- **Erklärbarkeit der Antworten** (nur „klassische“ KI)

## Typisch verwendete Tools und Methoden in der KI

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

# Zusammenfassung und Vorlesungsvorschau

## Anwendungsgebiete der KI:

- **Diagnose**

- Medizinische Diagnose
- Technische Diagnose

-  
*Überblick für alle + Vertiefung*

- **Optimierungsprobleme mit dynamischen Parametern**

- Fahrgastinformationssysteme
- Straßenverkehrsnavigation
- Logistik (TSP, Scheduling)

*Überblick für alle*

*Überblick für alle + Vertiefung*

*für alle + eventuell als Vertiefung*

- **Ressourcenverteilung**

- Straßenverkehrskoordination

*Überblick für alle*

- Zuordnungsprobleme mit vielfältigen Nebenbedingungen (Bsp. Stundenplan)

*Überblick für alle + Vertiefung*

# Zusammenfassung und Vorlesungsvorschau

## Anwendungsgebiete der KI:

- **Flexibles Management von verteiltem Wissen**
  - **Touristeninformationssystem** *eventuell als Vertiefung*
- **Semantische Antwortmaschine**
  - **Evi** *als Vertiefung*
- **Spiele, in denen die Maschine andere Spieler simuliert**
  - **rundenbasiert** -
  - **Echtzeit** *aber im SS 2015 als Seminar*

# Zusammenfassung und Vorlesungsvorschau

## Basistechnologien der KI:

- **Logische Programmiersprachen** *kurzer Einblick für alle in Prolog*
  - **Was** wird vom Menschen spezifiziert
  - **Wie** wird automatisch generiert
- **Suchstrategien** *für alle auch im Detail*
  - uninformiert vs. informiert
- **Constraint Satisfaction Problem (CSP)** *Überblick für alle*
  - Suche nach gültigen Lösungen
  - Suche nach optimalen Lösungen
- **Wissensbasierte Systeme (Spezialfall: Expertensysteme)** *für alle + Vertiefung*
  - Trennung in Wissen und Inferenzmaschine
  - Intelligenter Wissenserwerb und Wissensrepräsentation
  - Hauptfokus: Wiederverwendung

# Zusammenfassung und Vorlesungsvorschau

## Basistechnologien der KI:

- **Schwarmintelligenz**

*für alle + Vertiefung*

- verteilt
- statistisch
- nebenläufige Aktualisierung

- **Agentenorientierte Software**

-

- verteilt
- autonom
- proaktiv

- **Computerlinguistik**

*als Vertiefung*

- Ontologien: Aufbau und Verwaltung von Begriffswelten
- Maschinelle Verarbeitung natürlicher Sprache

# Praktische Alternativen

## Hamburger Touristen Information

*Vorbereitung für  
Abschlussarbeit*

- **Ontologiemangement**

## Ant Parcour

*für Javakundige*

- **Behebung von Bugs**
- **Behebung von Verhaltensauffälligkeiten**

## Ant Scout

*für OOP-Experten*

- **sparsamerer Einsatz von Ameisen**
- **Vergleich mit Dijkstra**

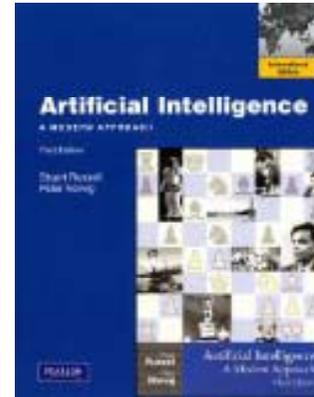
## Ihr Thema ...

# Literatur

## KI allgemein:



Günter Görz / Claus-Rainer Rollinger / Josef Schneeberger: *Handbuch der Künstlichen Intelligenz*  
Oldenbourg 2003 (4. Auflage), ISBN 3-486-27212-8  
ältere Auflage: 2000



Stuart Russell / Peter Norvig: *Artificial Intelligence: A Modern Approach*,  
Pearson 2010 (3. Auflage), ISBN 0-13-080302-2  
ältere Auflage: 2003

Wolfgang Ertel / Josef Schneeberger: *Grundkurs Künstliche Intelligenz*  
Vieweg 2009 (2. Auflage), ISBN 987-3-8348-0783-0

## spezielle Gebiete der KI:

Christoph Beierle / Gabriele Kern-Isberner: *Methoden Wissensbasierter Systeme*,  
Vieweg 2008 (4. Auflage), ISBN 978-3-8348-0504-1

Michael Wooldridge: *An Introduction to MultiAgent Systems*,  
Wiley 2009 (2. Auflage), ISBN 978-0-4705-1946-2

Marco Dorigo / Thomas Stützle: *Ant Colony Optimization*, MIT Press 2004, ISBN 0-262-04219-3

Steve Rabin: *AI Game Programming Wisdom*, Charles River Media, Bände 1 – 4, ISBN 978-1-58450-523-5 (Bd. 4)