

# Dialog-KI in IT-basierten Spielen

Kommunikation  
zwischen  
Mensch und NPC

*Vortrag von Friederike Bulka*

# Was ist eine Dialog-KI?

Es gibt (bis jetzt) 3 Arten wie Mensch und Rechner in Spielen kommunizieren.

(Die meisten Spiele benutzen mehrere dieser Möglichkeiten.)

# Was ist eine Dialog-KI?

## Beispiel 1

### Beispiel (1): keine sprachliche Kommunikation

Mensch kann nicht mit NPCs reden.

Spiele:  
**Strategiespiele,**  
**Wirtschafts-**  
**Simulationen**



# Was ist eine Dialog-KI?

## Beispiel 2

### Beispiel (2): Behelfssprache

Sprache die Mensch und Rechner leicht lernen können als Kompromiss. (Lambda-Kalkül)

Spiele: **Adventures**, teilweise auch **Simulationen**



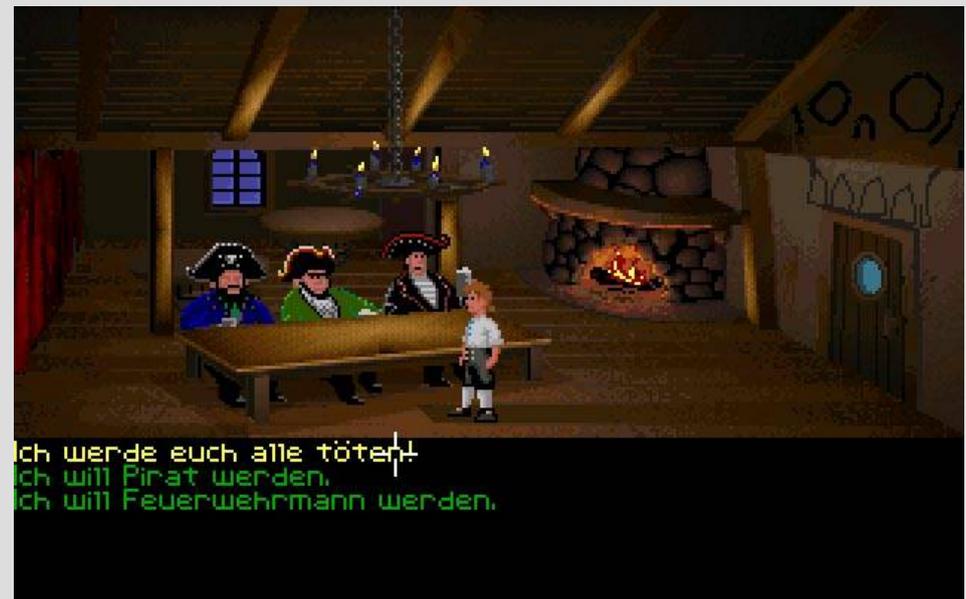
# Was ist eine Dialog-KI?

## Beispiel 3

### Beispiel (3): vorgegebene Dialogelemente

Kommunikation in „natürlicher Sprache“ aber Auswahl der Sätze ist begrenzt.

Spiele: **Rollenspiele, Adventures**



# Alternative: natürliche Sprache

Rechner muss natürliche Sprache erlernen.  
Wird im Moment noch nicht für NPCs verwendet.

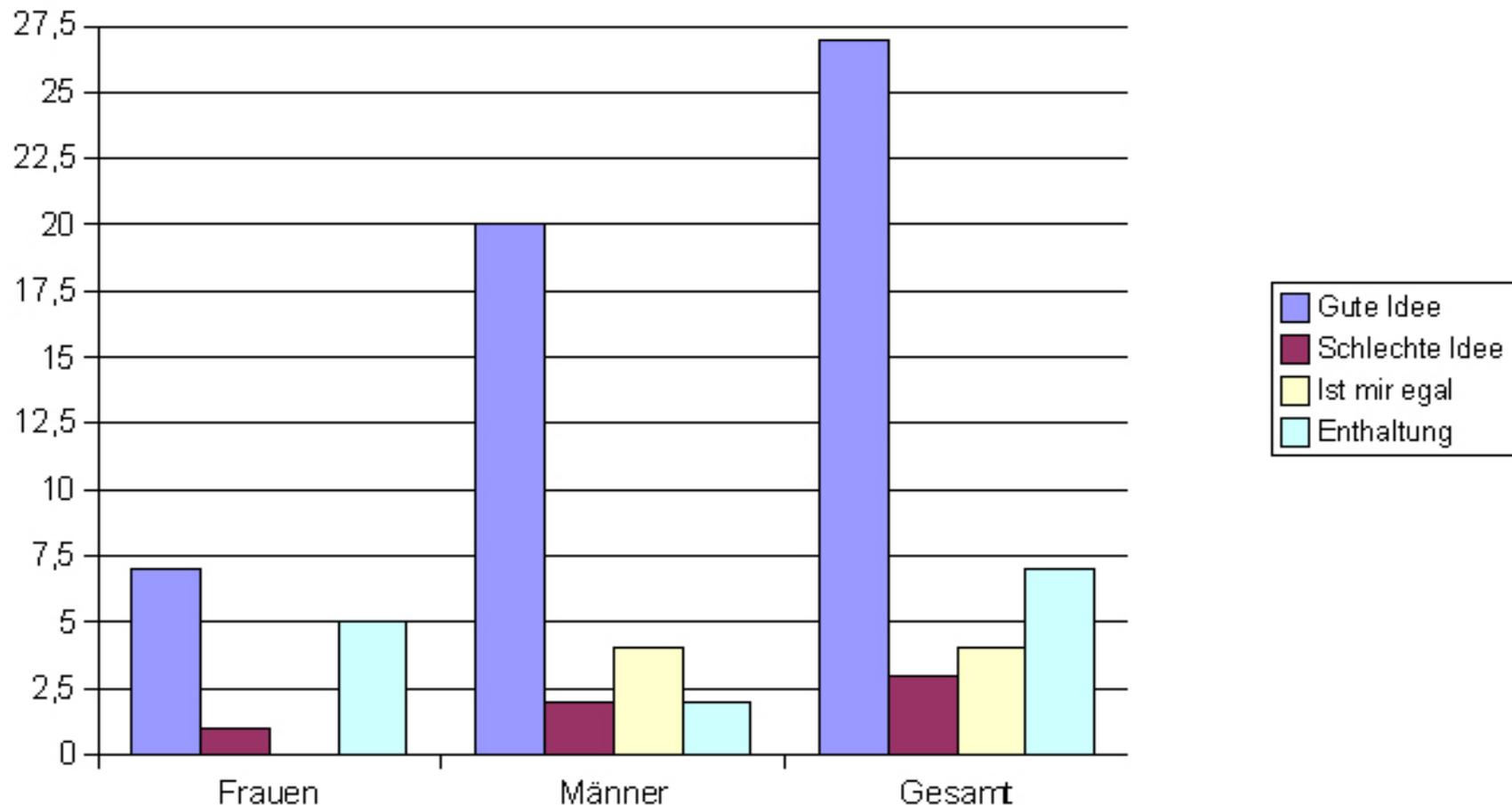
Vorteile:

- Einfache Benutzung für Spieler, weil Sprache bekannt
- breiteres Spektrum an Möglichkeiten
- „menschen-ähnliche“ NPCs
- realistische Spielwelt
- Unterstützung für den Spieler durch selbstständige NPCs (können Hinweise & Tipps geben)

# Ist Interesse vorhanden?

Ergebnis einer kleinen Umfrage auf der GC 2006.

Was halten Sie von natürlicher Kommunikation in Spielen?



# ELIZA

## A Computer Program For the Study of Natural Language Communication Between Man and Machine

- ▣ Joseph Weizenbaum entwickelte ELIZA (1964 bis 1966)
- ▣ Programm mit dem man in natürlicher Sprache reden kann
- ▣ ELIZA übernimmt die Rolle eines Psychaters, der Gegenfragen stellt

# ELIZA - Eigenschaften

Eliza kann mit folgenden Problemen umgehen:

- ▣ Erkennung der wichtigsten **Keywords**.
- ▣ Erkennen von minimalem **Kontext**.
- ▣ Eine geeignete **Transformationsregel** wird gewählt.
- ▣ ELIZA kann “intelligent” reagieren wenn die Eingabe **keine Keywords** enthält.
- ▣ ELIZA Scripts können **erweitert** und bearbeitet werden.

# ELIZA - Funktionsprinzip

Eingabe des Spielers: *Hey NPC, you are stupid!*

Vorgehensweise von ELIZA:

- ▣ Keywords erkennen: YOU \*
- ▣ Minimaler Kontext: YOU ARE \*
- ▣ Satz in Gegenfrage umwandeln:

# ELIZA - Funktionsprinzip

Eingabe des Spielers: *Hey NPC, you are stupid!*

Vorgehensweise von ELIZA:

- ▣ Keywords erkennen: YOU \*
- ▣ Minimaler Kontext: YOU ARE \*
- ▣ Satz in Gegenfrage umwandeln:

1	2	3	4	5
<i>Hey NPC,</i>	<i>you</i>	<i>are</i>	<i>stupid</i>	<i>!</i>

# ELIZA - Funktionsprinzip

Eingabe des Spielers: *Hey NPC, you are stupid!*

Vorgehensweise von ELIZA:

- ▣ Keywords erkennen: YOU \*
- ▣ Minimaler Kontext: YOU ARE \*
- ▣ Satz in Gegenfrage umwandeln:

1	2	3	4	5
<i>Hey NPC,</i>	<i>you</i>	<i>are</i>	<i>stupid</i>	<i>!</i>

**Response:**

<i>What makes you think</i>	<i>I</i>	<i>am</i>	<i>stupid</i>	<i>?</i>
-----------------------------	----------	-----------	---------------	----------

# Die Grenzen von ELIZA: Trickillusion

ELIZA arbeitet mit einer **Trickillusion**:

Ihr Verhalten wirkt menschlich, ist aber sehr **leicht durchschaubar**.

## **Beispiel**

Spieler: *You are **asfd**.*

NPC: *Does it please you to think I am **asfd**?*

ELIZA kann nicht zwischen einer sinnvollen und sinnfreien Aussage unterscheiden.

Kein Mensch würde normalerweise so antworten.

# ALICE

## Artificial Linguistic Internet Computer Entity

- ▣ entwickelt von Dr. Richard Wallace gewann dreimal den Loebner-Wettbewerb
- ▣ „menschen-ähnlichster“ ChatBot
- ▣ nutzt die XML-DTD **AIML** (Artificial Intelligence Markup Language)

# ALICE - Eigenschaften

- ▣ kann mehrere Keywords erkennen
- ▣ Kontext kann in größerem Umfang erkannt werden als bei ELIZA
- ▣ kann auf nichtssagende Benutzereingaben reagieren.
- ▣ ALICE kann sich Dinge „merken“

## **Bsp:**

Spieler: *Hi, my name is Friederike.*

NPC: *Nice to meet you Friederike.*

...

Spieler: *What's my name?*

NPC: *Your name is Friederike.*

# ALICE - AIML

AIML erlaubt folgende Operationen:

1. **Reduzieren** von schwieriger auf einfachere Grammatik
2. **Teile und Herrsche**: Eingabesätze können geteilt und dann separat bearbeitet werden.
3. **Synonyme** können erkannt werden und auf eine Bedeutung gemapt werden.
4. **Schreib- und Grammatikfehler** des Spielers können erkannt werden.

# ALICE – AIML II

Weitere von AIML unterstützte Operationen:

5. **Keywords** können im gesamten Eingabetext erkannt werden.
6. **Bedingungen:** ALICE kann sich zwischen mehreren Regeln zu entscheiden (ähnlich: IF ... THEN ... ELSE ...).

Diese 6 Operationen können auch kombiniert werden.

Qualität des Bot ist abhängig von Menge und Qualität der Regeln.

# Beispiel für AIML-Regeln

```
<category>
  <pattern>IT'S *</pattern>
  <template>
    <srai>IT IS <star/></srai>
  </template>
</category>
```

```
<category>
<pattern>IT IS *</pattern>
<template>
  <condition>
    <li name="kleidung" value="tshirt">
      <think><set name="tshirt"><star/></set></think>
      You said your tshirt is <star/>.
    </li>
    <li>You said something is <star/>.</li>
  </condition>
</template>
</category>
```

# Die Grenzen von ALICE: Turing Test

Alan Turing (1950): *Kann eine Maschine denken? Wenn ja, wie erkennt man eine denkende Maschine?*

**Turing-Test:** Ein **Schiedsrichter** kommuniziert mit einer Maschine und muss entscheiden ob er sich mit einem **Mensch oder einer Maschine** unterhält.

Bisher konnte noch **kein Programm** den Turing-Test bestehen.

Daraus entstand ein Wettbewerb bei dem der **Loebner-Preis** an das beste Programm vergeben wird.

# Grenzen von ALICE: begrenzttes Wissen

Das Wissen eines einzelnen Menschen umfasst extrem viele **Informationen**.

## 2 Probleme:

1. **Speicher- und Rechenkapazität** des Rechners sind begrenzt.
2. **Zeitaufwand**: Es dauert sehr lang einem Rechner die ganzen Daten „beizubringen“.

# Grenzen von ALICE: Polysemie

Ein Wort kann **mehrere Bedeutungen** haben. Bsp: *Leiter*

Die KI kann sich nur für **genau eine** Bedeutung entscheiden.

Fehler der KI:

1. **Ironie, Wortspiele** etc. können von der KI nicht verstanden werden.

2. **Eigennamen** werden oft nicht verstanden. - Großes Problem bei Übersetzungsmaschinen.

Bsp: *West Ham College* ≠ *Westschinkenhochschule*

# AIML in Spielen? Vorteile

## Vorteile:

- ▣ Spiele haben **eingeschränkte Welt**, die dem Rechner bekannt ist.
- ▣ Spiel-spezifische **Eigennamen sind bekannt** - sowohl von NPCs als auch vom Spieler.
- ▣ AIML-Datei lässt sich leicht um ein paar Regeln **erweitern**.
- ▣ Es gibt bereits viele ChatBots mit guter Spracherkennung. Diese können als **Grundlage** verwendet werden.
- ▣ AIML ist sehr **mächtig**. (<think>-Befehle)

# AIML in Spielen? Nachteile

Nachteile:

- ▣ AIML-Regeln müssten in **großem Umfang** erstellt und an die Spielwelt und ihre Sprache angepasst werden.
- ▣ Gefahr, dass die AIML-Regeln unübersichtlich werden.

# Verbesserungs-Ideen

**Hauptprobleme:** begrenztes Wissen und Polysemie

Diese Probleme werden in Spielen bereits verringert aber nicht unbedingt beseitigt.

Begrenztes Wissen:

**Internet nutzen** um Wissen zu sammeln.

Polysemie:

**Datenbank** mit mehrdeutigen Wörtern. Sobald ein mehrdeutiges Wort auftaucht muss der **Kontext genauer betrachtet** werden.

# Beispiel: mit Polysemie umgehen

Spieler: *I spent all my gold on my new mount.*

Rechner erkennt „spent“ als mehrdeutiges Wort und sucht im Satz nach Hinweisen welche Bedeutung gemeint ist.

**keywords:** *gold, on („spent ... on“)*

Response: *So you have no money left?*

Spieler: *Yes. Now I can't afford to spend the day without working.*

„spend“ ist mehrdeutig, **keywords:** *„the day“, working*

...

# Dialog-KI in Spiele einbinden

Dialog-KI benötigt **viel Speicherplatz und Rechenkapazität.**

**Aber:** Gewisse Grundregeln werden immer benötigt.  
(Spracherkennung)

## **mögliche Idee:**

Dialog-KI unabhängig vom Spiel entwickeln und in den Rechner integrieren - **KI-Karte** die AIML-Regeln und einen Parser enthält.

Nützliches für Entwickler:

Tools um das Erstellen von AIML-Regeln zu vereinfachen.  
Verwaltungsprogramm für die Dialog-KI?

# Dialog-KI-Karte

## **Vorteile:**

Kann von allen Spielen genutzt werden. Spiel selbst muss nur noch eigene Spiel-spezifische Regeln hinzufügen. Verbesserungen der Grund-KI wirken sich positiv auf alle Spiele aus.  
Spiele können „schlanker“ werden.

## **Nachteile:**

Spieler ohne Dialog-KI-Karte haben weniger Spielspaß. Im schlimmsten Fall können sie einige Spiele nicht spielen. Bei Verbesserungen muss eventuell neue KI-Karte gekauft werden.

# Zusammenfassung

Es gibt bereits viele Erfindungen im Bereich der Dialog-KI.

Aber es gibt noch mindestens genauso viel zu erfinden.

Spiele sind ideale Plattform zum experimentieren:

- ▣ viele Benutzer (Kritiker)
- ▣ verbesserte Dialog-KI kann Spiele interessanter machen
- ▣ Durch gute Dialog-KI und Spracherkennung kann man vielleicht irgendwann wirklich (mit einem Mikrofon) mit den NPCs „sprechen“.

# NPC der Zukunft?

