

# Cyc – Allgemeinwissen für Computer

Konrad Anton  
anton@informatik.uni-freiburg.de

# Übersicht

- Motivation

# Übersicht

- Motivation
- Das Cyc-Projekt

# Übersicht

- Motivation
- Das Cyc-Projekt
- Logik in Cyc

# Übersicht

- Motivation
- Das Cyc-Projekt
- Logik in Cyc
- Die Ontologie

# Übersicht

- Motivation
- Das Cyc-Projekt
- Logik in Cyc
- Die Ontologie
- Bastelstunde

# Motivation

Warum Allgemeinwissen?

# Motivation

Warum Allgemeinwissen?

`schwimmtin(Fred, Mittelmeer).`

Schwimmt Fred im Mittelmeer? ja.



# Motivation

Warum Allgemeinwissen?

`schwimmtin(Fred, Mittelmeer).`

Schwimmt Fred im Mittelmeer? ja. Ist Fred im Wasser? weiß nicht.

# Motivation

Warum Allgemeinwissen?

`schwimmtin(Fred, Mittelmeer).`

Schwimmt Fred im Mittelmeer? ja. Ist Fred im Wasser? weiß nicht. Ist Fred größer oder kleiner als das Mittelmeer? weiß nicht.

# Motivation, 2.

Datenbanken

# Motivation, 2.

## Datenbanken

Name	Alter	Jahre im selben Job
-----		
Steffi	18	7

Plausibilität?

# Motivation, 3.

**Natürlichsprachliche Systeme**

# Motivation, 3.

## Natürlichsprachliche Systeme

- “Ich spüle das Geschirr auf dem Tisch”

# Motivation, 3.

## Natürlichsprachliche Systeme

- “Ich spüle das Geschirr auf dem Tisch”
- “Ich kaufe eine Banane mit meiner Kreditkarte und esse sie auf”

# Das Cyc-Projekt

- **Encyclopedia**



# Das Cyc-Projekt

- **Encyclopedia**
- Doug Lenat, Ramanathan Guha und viele Hiwis...

# Das Cyc-Projekt

- **Encyclopedia**
- Doug Lenat, Ramanathan Guha und viele Hiwis...
- Microelectronics and Computer Technology Corp. (MCC), 1984
- seit 1995 Cycorp, Inc.

# Das Cyc-Projekt

- **Encyclopedia**
- Doug Lenat, Ramanathan Guha und viele Hiwis...
- Microelectronics and Computer Technology Corp. (MCC), 1984
- seit 1995 Cycorp, Inc.
- angeblich über 100.000 Atome, 1.000.000 Assertions

# Das Cyc-Projekt

- **Encyclopedia**
- Doug Lenat, Ramanathan Guha und viele Hiwis...
- Microelectronics and Computer Technology Corp. (MCC), 1984
- seit 1995 Cycorp, Inc.
- angeblich über 100.000 Atome, 1.000.000 Assertions

Frei verfügbar: OpenCyc

# Eingesetzt...

- Lycos (vor TerraLycos)

# Eingesetzt...

- Lycos (vor TerraLycos)
- CycSecure

# Eingesetzt...

- Lycos (vor TerraLycos)
- CycSecure
- nichtöffentliche Projekte: Sponsoren: DoD, MS, GlaxoSmithKline, . . .

# Cyc – Bestandteile

- Sprache *CycL*



# Cyc – Bestandteile

- Sprache *CycL*
- Inference Engine

# Cyc – Bestandteile

- Sprache *CycL*
- Inference Engine
- Ontologie

# Cyc – Bestandteile

- Sprache *CycL*
- Inference Engine
- Ontologie
- CycNL

# Cyc-NL

Proprietär... Herstellerangaben:

- POS-Tagger: syntaktische Informationen in der KB

# Cyc-NL

Proprietär... Herstellerangaben:

- POS-Tagger: syntaktische Informationen in der KB
- Mehrdeutige Parses: disambiguieren nach common-sense Plausibilität

# Cyc-NL

Proprietär... Herstellerangaben:

- POS-Tagger: syntaktische Informationen in der KB
- Mehrdeutige Parses: disambiguieren nach common-sense Plausibilität
- Semantische Repräsentation als CycL-Formeln.

# Cyc-NL

Proprietär... Herstellerangaben:

- POS-Tagger: syntaktische Informationen in der KB
- Mehrdeutige Parses: disambiguieren nach common-sense Plausibilität
- Semantische Repräsentation als CycL-Formeln.
- Beh. (Lenat, 1994): 75% von USA-Today

# Cyc-NL

Proprietär... Herstellerangaben:

- POS-Tagger: syntaktische Informationen in der KB
- Mehrdeutige Parses: disambiguieren nach common-sense Plausibilität
- Semantische Repräsentation als CycL-Formeln.
- Beh. (Lenat, 1994): 75% von USA-Today
- nicht in OpenCyc, keine Demo...



# Logische Grundlagen

Ausgehend von Prädikatenlogik...

$$\forall x : (\mathit{human}(x) \rightarrow \exists y : \mathit{fatherof}(x, y))$$

# Logische Grundlagen

Ausgehend von Prädikatenlogik...

$\forall x : (\text{human}(x) \rightarrow \exists y : \text{fatherof}(x, y))$

(#\$forall ?X

(\$implies (\$isa ?X #Human)

(\$thereExists ?Y (\$fatherOf ?X ?Y)))

# Typvorgaben bei Relationen

(#\$arg1isa #\$father #\$Animal)

(#\$arg2isa #\$father #\$MaleAnimal)

# Typvorgaben bei Relationen

```
(#$arg1isa #$father #$Animal)
```

```
(#$arg2isa #$father #$MaleAnimal)
```

Mehr als PL 1. Ordnung!

```
(#$isa #$genls #$TransitiveBinaryPredicate)
```

## zwei Ebenen

- Epistemologische Ebene

## zwei Ebenen

- Epistemologische Ebene
  - ★ externe Darstellung
  - ★ Ziel: Lesbarkeit
- Heuristische Ebene

## zwei Ebenen

- Epistemologische Ebene
  - ★ externe Darstellung
  - ★ Ziel: Lesbarkeit
- Heuristische Ebene
  - ★ interne Darstellung
  - ★ Ziel: Effizienz
  - ★ Häufige Konstruktionen optimiert, etwa transitive Relationen
- dazwischen Tell-Ask-Interface

# Wahrheit in Cyc

Auf epistemologischer Ebene: Wahr, Falsch



# Wahrheit in Cyc

Auf epistemologischer Ebene: Wahr, Falsch

Auf heuristischer Ebene:

- monoton wahr
- default-wahr
- unbekannt
- default-falsch
- monoton falsch

# Strukturierung

Problem: 1 Mio. Assertions. . .

# Strukturierung

Problem: 1 Mio. Assertions. . . Lösung: Mikrotheorien

- Hierarchisch aufgebaut

# Strukturierung

Problem: 1 Mio. Assertions. . . Lösung: Mikrotheorien

- Hierarchisch aufgebaut
- Konstanten: global, aber...

# Strukturierung

Problem: 1 Mio. Assertions. . . Lösung: Mikrotheorien

- Hierarchisch aufgebaut
- Konstanten: global, aber...
- Zu jeder Assertion: Angabe der Mt, in der sie gilt

# Strukturierung

Problem: 1 Mio. Assertions. . . Lösung: Mikrotheorien

- Hierarchisch aufgebaut
- Konstanten: global, aber...
- Zu jeder Assertion: Angabe der Mt, in der sie gilt
- Beschränkt Inkonsistenzen

# Strukturierung

Problem: 1 Mio. Assertions. . . Lösung: Mikrotheorien

- Hierarchisch aufgebaut
- Konstanten: global, aber...
- Zu jeder Assertion: Angabe der Mt, in der sie gilt
- Beschränkt Inkonsistenzen
- Verkleinert Suchraum!

# Die Ontologie

Wie sieht für Cyc die Welt aus?



# Die Ontologie

Wie sieht für Cyc die Welt aus? Einblick in...

- Oberste Etage
- Raum und Zeit
- Vorgänge
- Agenten
- . . . .

# Die Ontologie

Wie sieht für Cyc die Welt aus? Einblick in...

- Oberste Etage
- Raum und Zeit
- Vorgänge
- Agenten
- . . . .

Ziel: zumindest Eindruck von der Größe vermitteln.

# Konzeptklassen, Instanzen...

# Konzeptklassen, Instanzen...

- Instanz (Individual), z.B. Goldi01
- Konzeptklasse (Collection), z.B. Fish
- Metaklasse (Collection of collections), z.B. BiologicalClass

# Konzeptklassen, Instanzen...

- Instanz (Individual), z.B. Goldi01
- Konzeptklasse (Collection), z.B. Fish
- Metaklasse (Collection of collections), z.B. BiologicalClass

z.B.

```
(#$genls #Fish #AquaticOrganism)
```

```
(#$isa #Fish #BiologicalClass)
```

```
(#$isa #Goldi01 #Fish)
```

# Oberste Etage

- `Thing`: alles ist ein `Thing`
- `Collection`: die Menge der Mengen z.B. `SiblingDisjointCollection`
- `Relation isa`: Element von
- `Relation genls`: Teilmenge von
- `Individual`: Menge der Nichtmengen
- `Intangible`: Menge der nicht greifbaren `Things`
- `IntangibleIndividual`: Menge der nicht greifbaren Nichtmengen

...

# Beschaffenheiten

Eine mögliche Unterscheidung:

# Beschaffenheiten

Eine mögliche Unterscheidung:

- Stoffe (z.B. Holz, Gehen)



# Beschaffenheiten

Eine mögliche Unterscheidung:

- Stoffe (z.B. Holz, Gehen)
- Objekte (z.B. Tisch, 100-Meter-Lauf)

# Beschaffenheiten

Eine mögliche Unterscheidung:

- Stoffe (z.B. Holz, Gehen)
- Objekte (z.B. Tisch, 100-Meter-Lauf)

Aus Raum- (StuffType, ObjectType) und Zeitperspektive (Temporal... )!

# Beschaffenheiten

Eine mögliche Unterscheidung:

- Stoffe (z.B. Holz, Gehen)
- Objekte (z.B. Tisch, 100-Meter-Lauf)

Aus Raum- (StuffType, ObjectType) und Zeitperspektive (Temporal... )!

Beides gleichzeitig: (ExistingObjectType, ExistingStuffType)

# Zeit

# Zeit

- TemporalThing: etwas, das zeitliche Ausdehnung hat

# Zeit

- TemporalThing: etwas, das zeitliche Ausdehnung hat
- SomethingExisting: etwas, das während eines Zeitraums existiert (z.B. Stuhl, Vertrag)

# Zeit

- TemporalThing: etwas, das zeitliche Ausdehnung hat
- SomethingExisting: etwas, das während eines Zeitraums existiert (z.B. Stuhl, Vertrag)
- TimeInterval: nacktes Zeitintervall

# Zeit

- TemporalThing: etwas, das zeitliche Ausdehnung hat
- SomethingExisting: etwas, das während eines Zeitraums existiert (z.B. Stuhl, Vertrag)
- TimeInterval: nacktes Zeitintervall

Relationen z.B.

(#\$temporallySubsumes #\$BarockZeit #\$BachsGeburt )



# Zeit

- TemporalThing: etwas, das zeitliche Ausdehnung hat
- SomethingExisting: etwas, das während eines Zeitraums existiert (z.B. Stuhl, Vertrag)
- TimeInterval: nacktes Zeitintervall

Relationen z.B.

(#\$temporallySubsumes #\$BarockZeit #\$BachsGeburt )

Jahre  $\neq$  Jahreszahlen! (#\$YearFn 1980)

# Raum

- `SpatialThing`: Menge der Dinge, die “wo sind”
- Relationen wie “in” (mind. 16 Varianten), Ausrichtung, . . .

# Situationen

- `Situation`: Zusammenhang mehrerer Objekte
- `Event`: Ereignis
- `StaticSituation`: Situation, in der nichts eigentlich passiert (z.B. Rumsitzen)
- `Group`: Menge von Individuen, die als Menge eine zeitliche Ausdehnung hat

# Vorgänge

- subEvents: Teilereignis eines Ereignisses
- ActorSlots

```
(#$isa #Frühstück27Juni #Event) -- nicht empfohlene Klasse
($subEvents #Frühstück27Juni #Müslimachen27061)
($performedBy #Frühstück27Juni #Konrad)
($objectActedOn #Frühstück27Juni #Müsliportion1234)
($outputCreated #Müslimachen27061 #Müsliportion1234)
```

# Teil-Ganzes

physicalParts: Ganzes-Einzelteil-Beziehung

(#\$physicalParts #\$Car1 #\$Wheel22)

physicalPortions : Ganzes-Teil-Beziehung (Teil ist vom selben Material)

(#\$physicalPortions #\$MensaessenPortion11 #\$EinBissen53)

groupMembers : Gruppe-Teil-Beziehung

(#\$groupMembers #\$Angelika #\$SeminarMedinf2Leute)

# Agenten

- Agent: etwa jemand oder etwas, der etwas tut
- $\neq$  Actors: auch `#$Müsliportion1234...`
- IntelligentAgent: hat Wissen und Ziele
- Emotionen. z.B. Relationen `feelsTowardsObject`, `appropriateEmot`  
`(#$feelsEmotion #$Jochen20 ($HighAmountFn #$Fear))`

# Aussagen über Aussagen

(#\$knows #LvBeethoven

(#\$temporallySubsumes #BarockZeit

#\$JSBachsGeburt))

(#\$holdsIn (#YearFn 1996)

(#\$groupMembers #DemocraticParty

#\$BillClinton))

# Biologie

Abbildung der Linnéschen Hierarchie...



# Kleine Demo

über den Neujahrstag:

- Januar ist ein Monat
- Neujahrstag ist im Januar
- Neujahrstag ist ein Feiertag (in der nordam. Kultur)
- Beh.: der Neujahrstag ist im Winter

# Probleme

- Hoffnung auf Konvergenz?

# Probleme

- Hoffnung auf Konvergenz?
- (zu?) pragmatische Herangehensweise

# Probleme

- Hoffnung auf Konvergenz?
- (zu?) pragmatische Herangehensweise
- Kommerziell

# Probleme

- Hoffnung auf Konvergenz?
- (zu?) pragmatische Herangehensweise
- Kommerziell
- Nordamerikanische kulturelle Ausrichtung?

# Zusammenfassung

- Cyc ist ein ambitioniertes Projekt
- mächtige Wissensrepräsentationssprache
- jemals genug Allgemeinwissen?