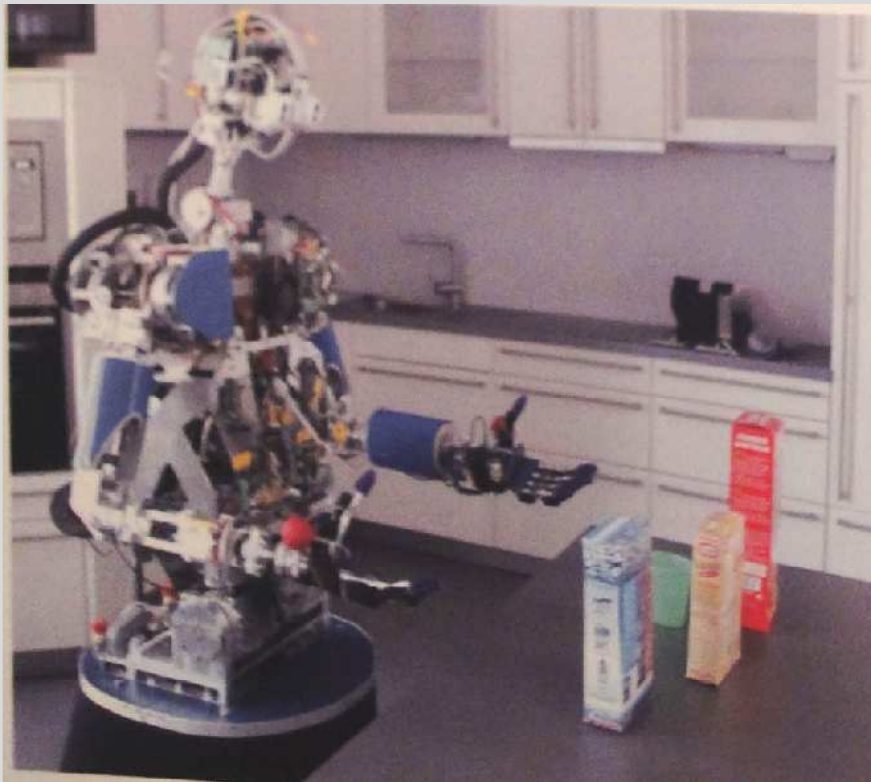


# Conceptual Modelling of Task Pattern & Analysis

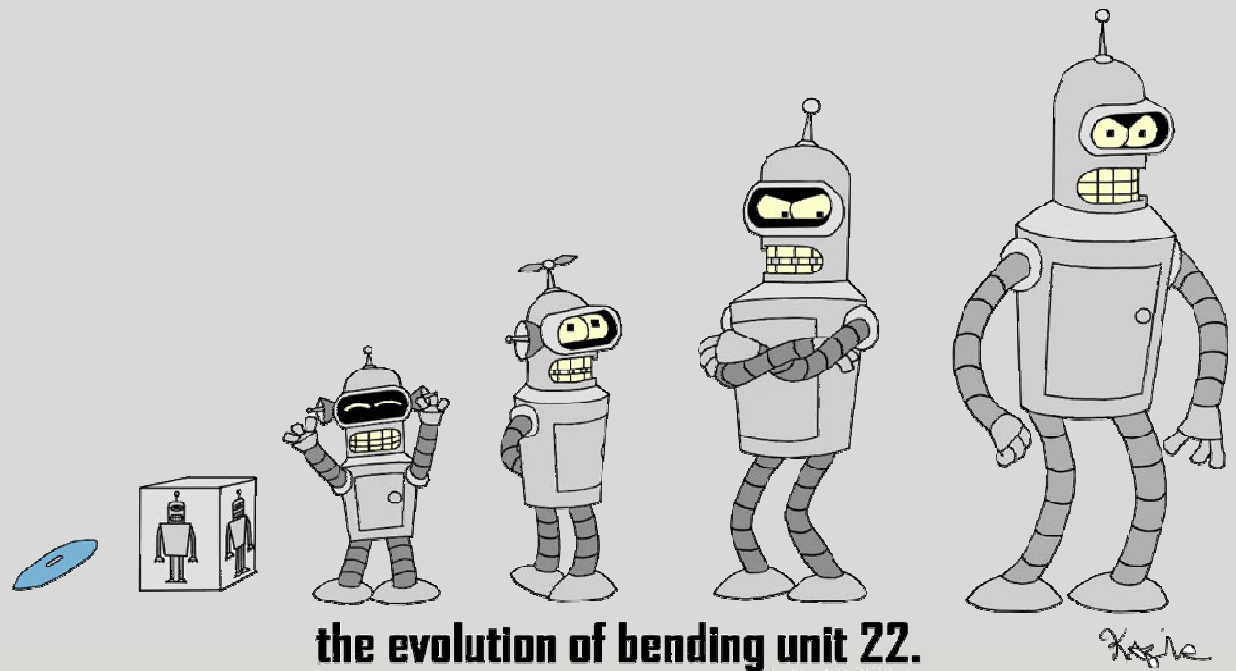


Pierre Babeck & Tom Tschernack

# Conceptual Modelling of Task: Pattern & Analysis

## Gliederung

1. Einleitung
2. Bisherige Ansätze
3. Robot Task – Grundlagen
  1. Charakteristik
  2. Komplexität
  3. Task Pattern
  4. Task Fragmente
  5. Task-Pattern-Sprache
4. Task Pattern Analysis
5. Bewertung



# 1. Einleitung

# Conceptual Modelling of Task: Pattern & Analysis

## 1. Einleitung



# Conceptual Modelling of Task: Pattern & Analysis

## 1. Einleitung

- Service-Roboter als Haushaltshilfe
  - leistungsfähig und flexibel (viele verschiedene Aufgaben in verschiedenen Zusammenhängen)
- Einfache Instruierung nötig
- Problem: bei großer Anzahl an Objekten und Räumen größere Anzahl an möglichen Instruktionen
- Aber: Gemeinsamkeiten bei vielen Aufgaben
  - Bringe **Tasse** in die **Küche** – Bringe **Buch** ins **Wohnzimmer**
- Ziel: Finden der Gemeinsamkeiten!

## 2. Bisherige Ansätze

# Conceptual Modelling of Task: Pattern & Analysis

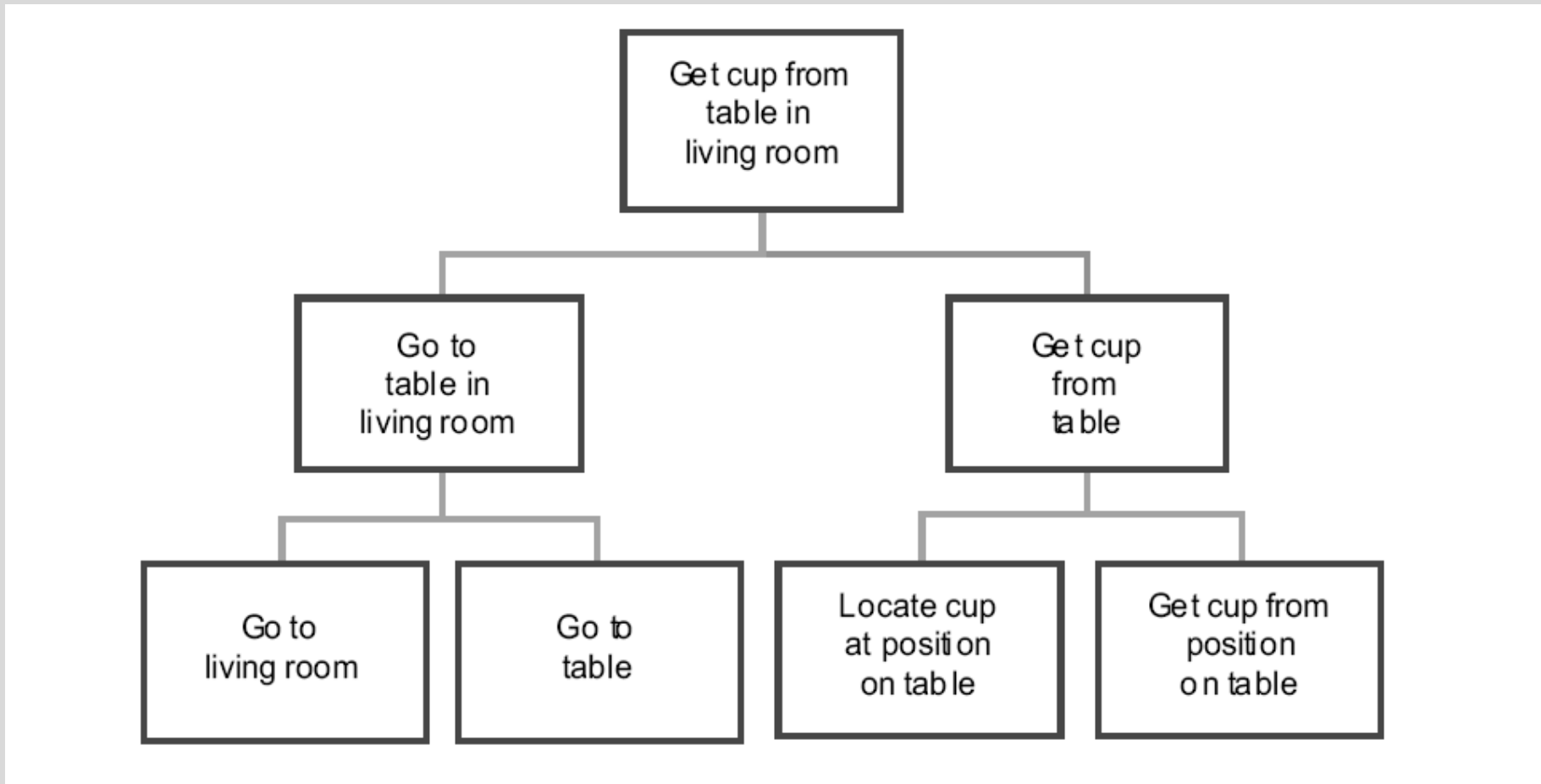
## 2. Bisherige Ansätze

- Hierarchische Struktur von Aktivitäten
  - wie in anderen Task-Analyse-Methoden
    - HTA, TAKD, TKS ...
- Entwurfsmuster
  - z.B. aus der Softwareentwicklung (MVC-Modell, Beobachter,... )
- Inspiration aus anderen Arbeiten zu Robotern

# Conceptual Modelling of Task: Pattern & Analysis

## 2. Bisherige Ansätze

### HTA

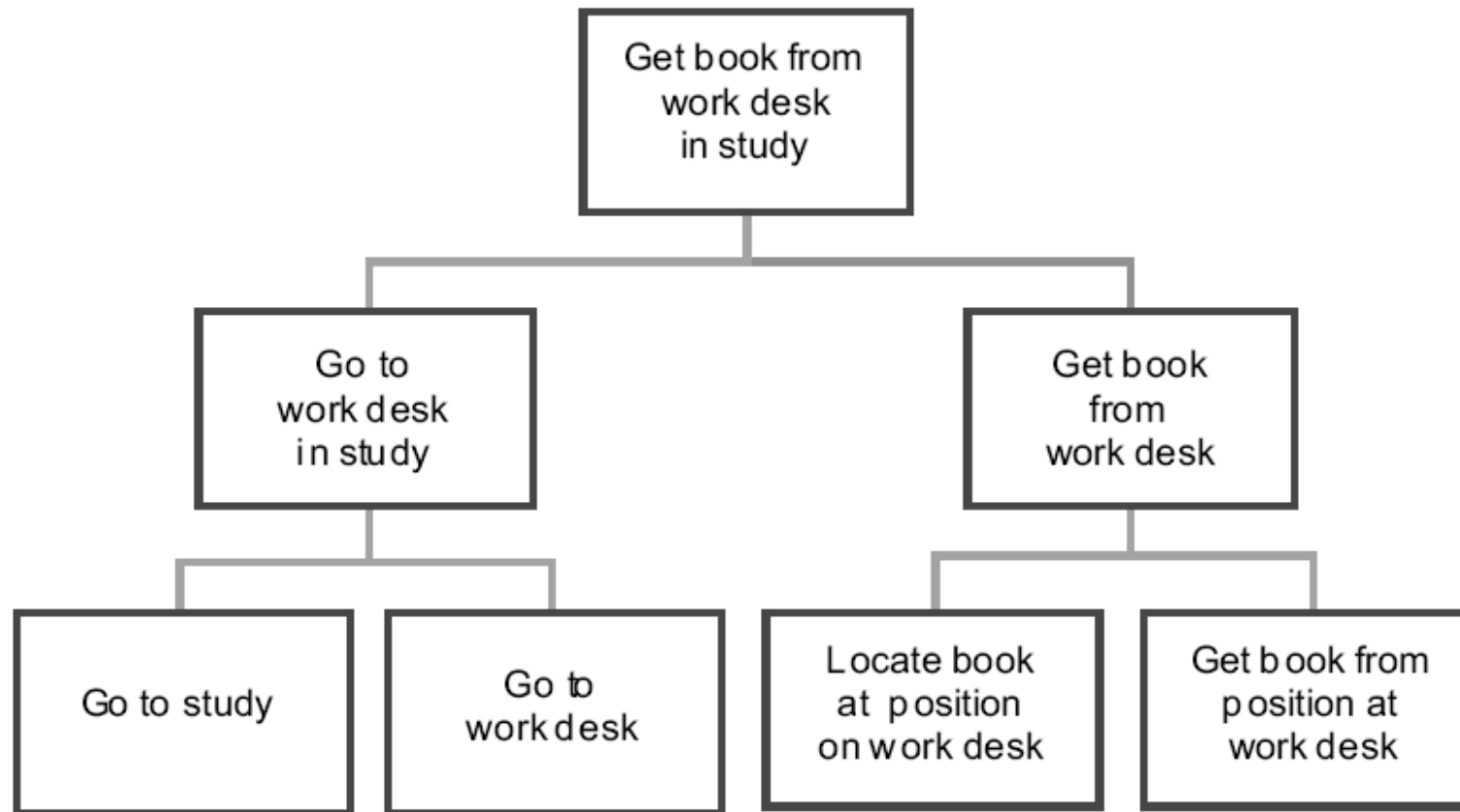




# Conceptual Modelling of Task: Pattern & Analysis

## 2. Bisherige Ansätze

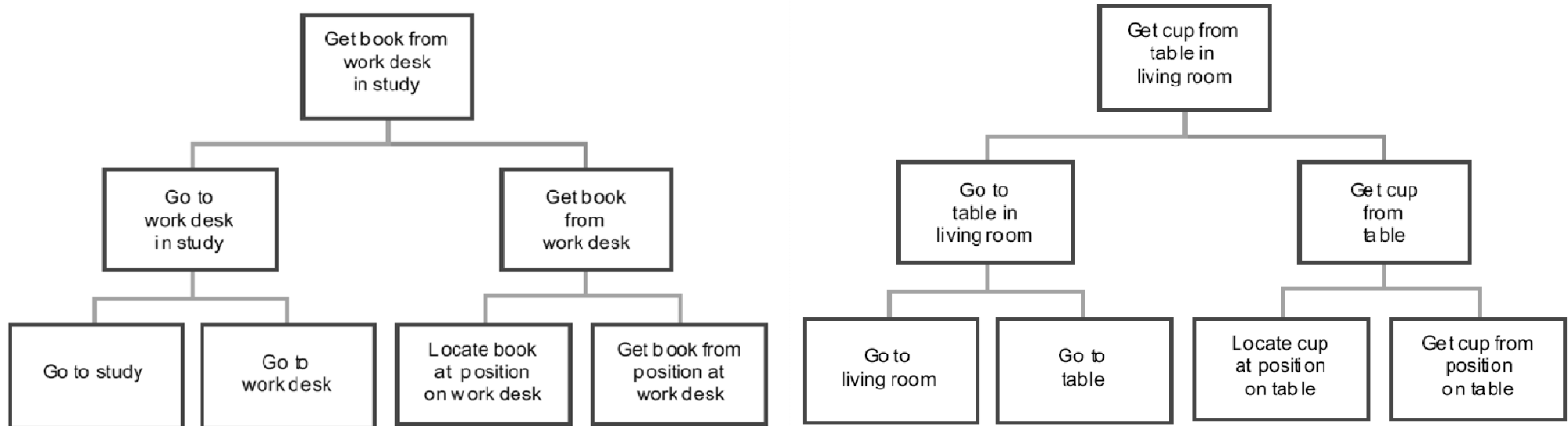
### HTA



# Conceptual Modelling of Task: Pattern & Analysis

## 2. Bisherige Ansätze

### HTA



- **Nachteil:**

Fehlende Unterscheidung von Eigenschaften der beteiligten Objekte:

Buch != gefüllte Tasse

# 3. Robot Task

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Grundlagen

### Robot Task?

- Aufgabe die ein Roboter erfüllen soll
- Aufgaben sind umfangreich und lassen sich meiste in viele kleine Aufgaben unterteilen
- Kleine Aufgaben haben meist eine gemeinsame Struktur
- Über diese einfachen Strukturen lässt sich ein Framework oder eine Toolbox entwickeln
- Grundlegende Idee stammt aus der Softwarearchitektur, den Entwurfsmustern (Design Patterns)

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | **Charakteristik**

### Charakteristik von Robot-Tasks:

- das Bewegen von Objekten in seiner Umgebung
- das Verändern von Objekten
- das Bewegen des Roboters in einer bekannten Umgebung (navigations maps)

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | **Komplexität**

- Grundlegende Task:

- einfachste Aufgaben, Baukastenprinzip

z.B.:           Suche ein Objekt auf einem Tisch.

                  Bewege dich zu einem Raum.

                  Identifiziere ein Objekt.

- Einfache Task

- nicht wiederholte einfache Aufgaben

z.B.:           Greif das Buch.

                  Bewege dich zu dem Tisch in diesem Raum.

                  Finde das Buch auf dem Tisch.

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | **Komplexität**

- Zusammengesetzte Task
  - verfolgt einen Plan, dauert länger, beinhaltet mehrere Objekte, Position und Funktion

z.B.:            `Check that the table is clear and wiped`  
                  `If not, CLEAR TABLE`  
                  `Go to kitchen`  
                  `Pick one cover`  
                  `Pick fork`  
                  `Pick knife`  
                  `Go to dining room`  
                  `Place one cover`  
                  `Place fork`  
                  `Place knife`  
                  `Repeat from X for the proper number of diners`

Anmerkung: Wann endet eine Aufgabe?

Bewege dich zu einem Ort – ▲

Signal Error – ▲

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Pattern

Die Grundlegende Schablone zum erstellen von bestimmten Tasks

Zur Nutzung der Schablonen müssen die Platzhalter durch entsprechende Variablen ersetzt werden.

Formale Beschreibung:

[variable 1] static text 1 [variable 2]  
static text 2 ... [variable n]  
where “constraining and specifying  
Statements”  
is “action statements”



Spezifiziert:

<Actor> gets <object> from <place>  
where (<Actor> = [robot, user],  
<object> = [cup],  
<place> = [tabletop])  
is [Actions].





# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Pattern

Die nutzbare Funktion für Prolog

Spezifiziert:

**<Actor> gets <object> from <place>**  
**where (<Actor> = [robot, user],**  
**<object> = [cup],**  
**<place> = [tabletop])**  
**is [Actions].**

Prolog Funktion:

**get\_object\_from(Actor, Object, Place, T1, T2) :-**  
**( Actor = robot ;**  
**Actor = user ),**  
**Object = cup,**  
**Place = tabletop,**  
**Actions.**

*Andere Schreibweise*

**r1(X,Y,Z,T1,T2):- ( X = robot; X = user ), Y = cup, Z = tabletop.**

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Pattern

Bsp.

Ausgangs Pattern:

[variable 1] static text 1 [variable 2]  
static text 2 ... [variable n]  
where “constraining and specifying  
Statements”  
is “action statements”

<Agent> goes to <room> and delivers <object>  
where <room> = “kitchen”  
and <object> = “coffee cups”  
and <agent> = “robot”  
is ...

*“Robot goes to the kitchen and delivers the coffee cups”*

<Agent> goes to <room> and delivers <object>  
where <room> = “Ann’s room”  
and <object> = “book”  
and <agent> = “robot”  
is ...

*“Robot goes to Ann’s room and delivers the book”*

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Pattern

- Bedingungen

<Agent1> goes to <room 1> and delivers <object>

is

TP1: <Agent1> goes to <room 1>

TP2: <Agent1> waits in <room 1>

until <event>

where event is one of

{ <Agent 2> takes <object>, time out }

TP3: <Agent 2> goes to <room 2>

where room 2 is "home"

Conditions: TP1: <Agent 1> is not in <room1>

TP2: <Agent 1> is in <room1>

<Agent 2> is in <room1>

TP3: <Agent 1> is not in <room2>

Was abzarbeiten ist, unter der Bedingung, dass *er nicht in dem Raum ist*.

Warte in dem Raum bis, jemand das Objekt nimmt. Wobei, beide in dem gleichen Raum sein müssen.

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Fragmente

### Unterscheidung von Objekten

- physische Umwelt: Zimmer, Türen, Fenster ...
- Einrichtung: Schränke, Tische, Regale ...
- Elektrische Geräte: Kühlschrank, Herd, TV...
- Lose Gegenstände: Tassen, Vasen, Bücher ...

### Dynamische vs. Statische Objekte (veränderbar)

- Grundsätzlich sind dann Eigenschaften von Objekten als Default oder Initialwerte zu betrachten.

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Fragmente

### Eigenschaften von Objekten

Bowl	Container: yes Content: food Weight: light Size: small Solidity: hard Type: utensil
Plastic Bag	Container: yes Content: miscellaneous Weight: light Size: small Solidity: soft Type: utensil
Chest	Container: yes Content: miscellaneous Weight: heavy Size: large Solidity: hard Type: furniture
Book	Container: no Content: N/A Weight: light Size: small Solidity: hard Type: object

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task Fragmente

### Strukturierung von Objekten

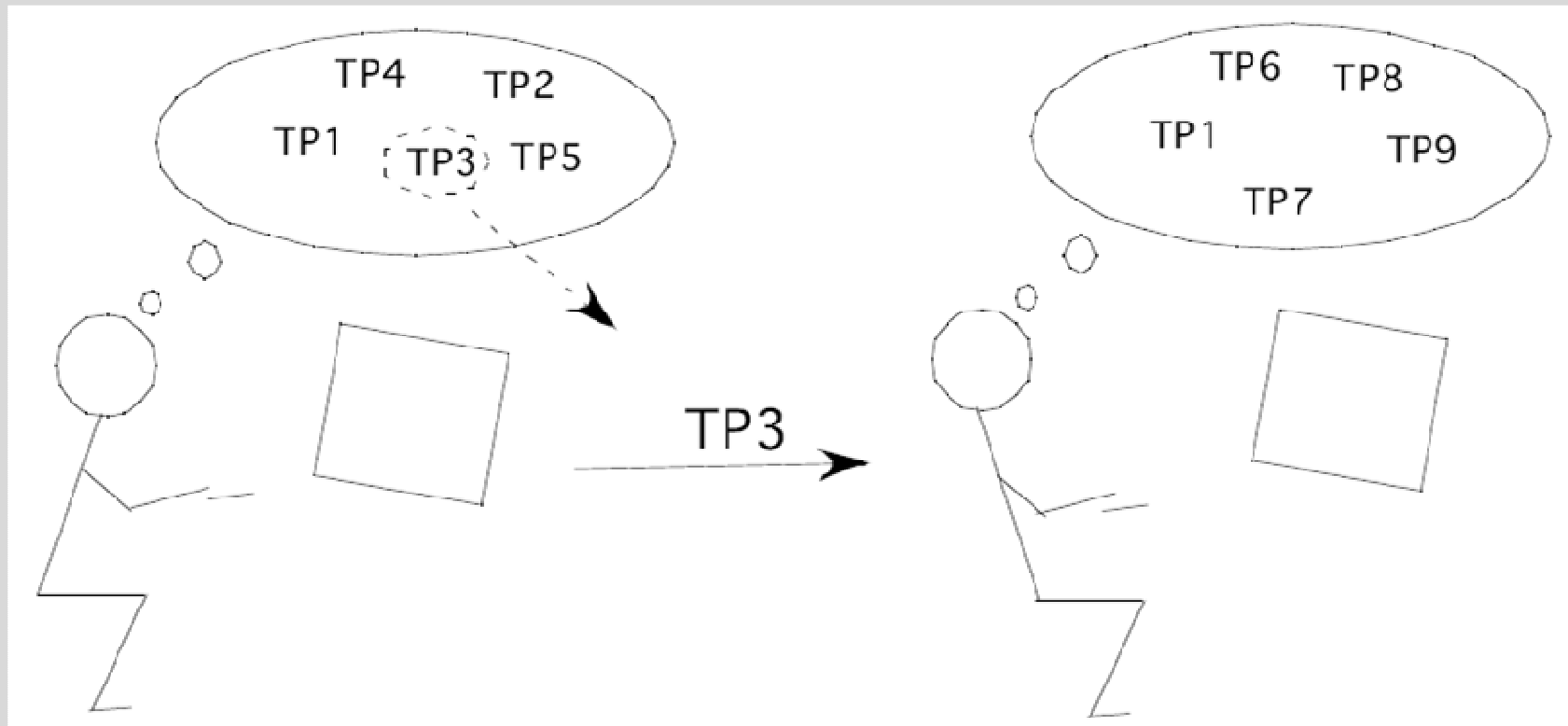
- Unterteilung von Objekten in Unterobjekte (in hierarchischen Strukturen)  
z.B.: ein Raum setzt sich aus vier Wänden zusammen, Fenstern, Türen usw.

Vorteil: bei der Beschreibung von Aufgaben muss nicht die Wand xy als Ziel angeben werden sondern nur das „Haupt“-Element also die größte Einheit z.B.: Raum xy.

# Conceptual Modelling of Task: Pattern & Analysis

## 3. Robot Task | Task-Pattern-Sprache

Bietet dem Roboter in einer bestimmten Situation eine Menge an Patterns zur Problemlösung an



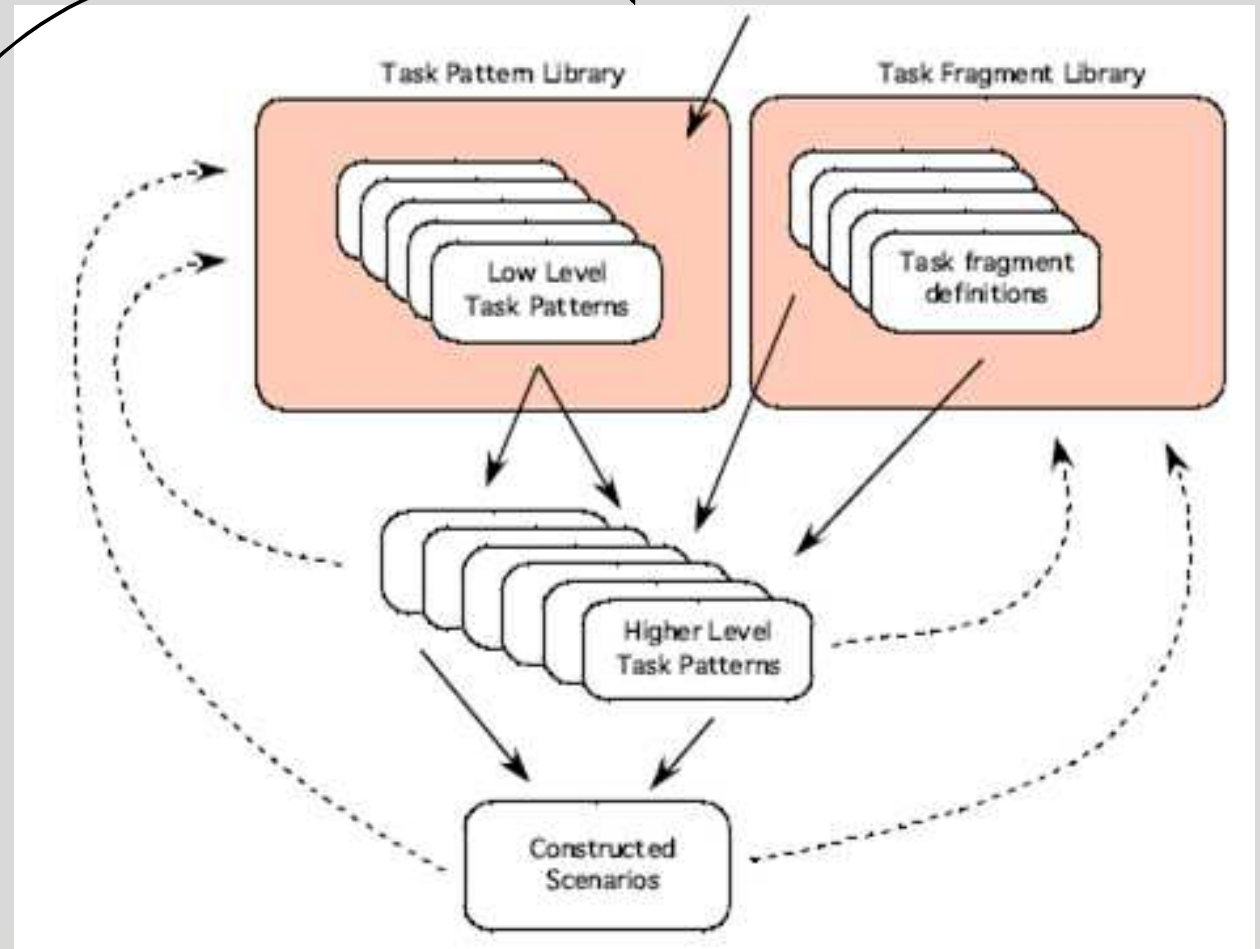


## 4. Task Pattern Analysis (TAPAS)



# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**



# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**

Bsp: Interview

„Peter is completely unable to move from the neck down. He uses an electric wheelchair to move around in the apartment. He wants to have a glass of water. He asks the robot to bring a glass of water to the living room, where he currently sits. The robot goes to the cupboard in the kitchen. The robot opens the cupboard and takes out one glass. The robot moves to the sink and pours water into the glass. The robot puts a straw in the glass. The robot goes into the living room and moves close to Peter's wheel chair. It moves the glass with the straw towards Peter's mouth and stops close enough for him to reach the straw. When Peter has finished drinking, he tells the robot to put away the glass. The robot goes into the kitchen, empties the remaining water in the sink, and puts the glass in the dishwasher.“

# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**

Bsp: Interview

„Peter is completely unable to move from the neck down. He uses an electric wheelchair to move around in the apartment. He wants to have a glass of water. He asks the robot to bring a glass of water to the living room, where he currently sits. The robot goes to the cupboard in the kitchen. The robot opens the cupboard and takes out one glass. The robot moves to the sink and pours water into the glass. The robot puts a straw in the glass. The robot goes into the living room and moves close to Peter's wheel chair. It moves the glass with the straw towards Peter's mouth and stops close enough for him to reach the straw. When Peter has finished drinking, he tells the robot to put away the glass. The robot goes into the kitchen, empties the remaining water in the sink, and puts the glass in the dishwasher.“

# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | Vorgehen

### Bsp: Aktivitäten

Peter asks the robot to bring him a glass of water.

The robot goes to the kitchen.

The robot goes to the cupboard (in the kitchen).

The robot opens the cupboard.

The robot takes out a glass from the cupboard.

The robot goes to the sink.

The robot pours water into the glass (extended description)

    The robot moves the glass under the tap.

    The robot opens faucet.

    The robot waits until glass is filled.

    The robot closes faucet.

The robot puts the glass on the carrying tray.

The robot gets a straw from the cupboard.

The robot puts the straw into the glass.

The robot goes into the living room.

The robot goes to Peter's wheel chair.

The robot moves the glass with the straw to Peter's mouth.

The robot waits.

Peter drinks water. (no-op activity)

Peter tells the robot that he has finished.

Peter tells the robot to put away the glass.

The robot goes to the sink in the kitchen

The robot empties the glass in the sink.

(The robot moves to the dishwasher) – only if needed

The robot opens the dishwasher.

The robot puts the glass in the dishwasher.

The robot closes the dishwasher.

# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**

Bsp: Aktivitäten

1. Peter asks the robot to bring him a glass of water.
2. The robot goes to the kitchen.
3. The robot goes to the cupboard (in the kitchen).
4. The robot opens the cupboard.
5. The robot takes out a glass from the cupboard.
6. The robot goes to the sink.



### Task Patterns

<Agent> asks <Agent1> for <Action> [1]  
<Agent> goes to <Room> [2]  
<Agent> goes to <Place> [3] [6]  
<Agent> opens <Object> [4]  
<Agent> gets <Object> from <Place> in <Object1> [5]



### Task Fragments

- Robot, Peter  
- glass, sink  
- kitchen,  
- cupboard

# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**

### Bsp: Task Patterns

<Agent> asks <Agent1> for <Action>  
<Agent> goes to <Room>  
<Agent> goes to <Place>  
<Agent> opens <Object>  
<Agent> gets <Object> from <Place> in <Object1>  
<Agent> fills <Object> with <Content>  
<Agent> moves <Object> to <Place>  
<Agent> operates <Control>  
<Agent> waits until <Event>  
<Agent> places <Object> on <Place>  
<Agent> puts <Object> into <Object2>  
<Agent> performs no\_op operation  
<Agent> tells <Agent1> <information>  
<Agent> goes to <Place> in <Room>  
<Agent> empties <Object> in <Place>  
<Agent> goes to <Object>  
<Agent> closes <Object>

# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**

### Bsp: Task Fragments

Acting Agents	Robot, Peter	<agent>
Rooms	kitchen, living room	<room>
Objects	glass, straw, cupboard, dishwasher, carrying tray, tap, faucet, wheel chair (peter's), cupboard door, dishwasher door	<object>
Containers (box)	dishwasher, cupboard	<object>, <location>
Containers (open)	glass, sink	<object>, <location>
Locations (fixed)	Cupboard	<location>
Locations (dynamic)	Peter's wheel chair	<location>, <object>

# Conceptual Modelling of Task: Pattern & Analysis

## 4. Task Pattern Analysis | **Vorgehen**

### Bsp: Synthese - Task Patterns Structures

```
<Agent> puts <Object> into <Object1> is  
TP1: <Agent> gets <Object>  
TP2: <Agent> moves to <Room>  
TP3: <Agent> moves to <Object1>  
TP4: <Agent> opens <Object1>  
TP5: <Agent> puts <Object> at <Place>  
      in <Object1>  
TP6: <Agent> closes <Object1>  
      where  
      <Agent> = {agent}  
      <Object> = [cup, bowl, plate]  
      <Object1> = [cupboard, dish washer]  
      <Place> = [shelf]
```



# 5. Bewertung

# Conceptual Modelling of Task: Pattern & Analysis

## 5. Bewertung

### Evaluations-Tests

- Studenten aus unterschiedlichen Richtungen
- 9 Szenarien mit 140 Aktivitäten und 23 TP
- Aufgabe: Anwenden der Analyse-Methode und Bewertung aus didaktischer Sicht
- ERGEBNIS:
  - Ähnliche Form der Beschreibungen, Unterschiede in Variationen der Syntax oder oder bei Komplexität
  - Bewertung: ansprechend und nützlich, aber Schwierigkeiten mit Abstraktionsstufen, bei großer Anzahl von Patterns und „Programmier-Denkart“

# 6. Quellen

# Conceptual Modelling of Task: Pattern & Analysis

## 6. Quellen

- Task Pattern for Human-Robot Interaction

Lars Oestricher: TAMODIA '02 Proceedings of the First International Workshop on Task Models and Diagrams for User Interface Design

- Conceptual Modelling of Tasks

Lars Oestricher, 2009, Seiten 129-149

Ende

# Conceptual Modelling of Task: Pattern & Analysis

## Gliederung

1. Einleitung
2. Bisherige Ansätze
3. Robot Task – Grundlagen
  1. Charakteristik
  2. Komplexität
  3. Task Pattern
  4. Task Fragmente
  5. Task-Pattern-Sprache
4. Task Pattern Analysis
5. Bewertung

