

# Kapitel 9: Frames

(Dieser Foliensatz basiert auf Material von Mirjam Minor, Humboldt-Universität Berlin, WS 2000/01)

## Frames

Ein **Frame** ist eine Musterbeschreibung für ein Objekt, in der unterschiedliche Rollen realisiert sind.

Zu den **Rollen** eines Konzeptes gehören auch solche, die Verweise zu Ober- und Unter-Konzepten widerspiegeln.

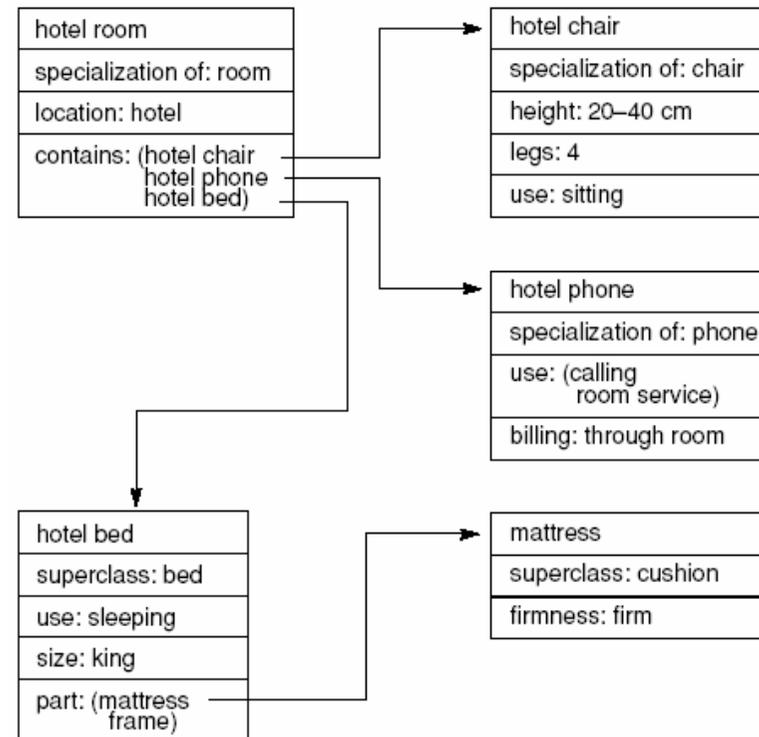
Entsprechend ist auch Vererbung möglich.

Analogie zu objektorientierter Programmierung:

- dort aber mehr die dynamische, prozedurale Sicht der Interaktion von Objekten (messages),
- hier mehr die deklarative, statische Sicht der Wissens-Repräsentation

Bezug zu semantischen Netzen:

- Gegenüber den einfachen Konzepttheorien in Semantischen Netzen wird in den Schemaansätzen eine reichere interne Struktur des Gedächtnisses angenommen.



## Erwartungsgesteuerte Verarbeitung

Frames orientieren sich an einer erwartungsgesteuerten Verarbeitung neuer Informationen, die der menschlichen Betrachtungsweise entspricht.

Beispiel: Der Begriff "Fahrrad" aktiviert Vorstellungen, in deren Rahmen problemlos Konkretisierungen (z.B. 5-Gang-Kettenschaltung von Shimano) für "mein Fahrrad" angegeben werden können.

Das Konzept der Frames enthält die Möglichkeit zur Angabe von **Default-Werten**, die später überschrieben werden können (z.B. Anzahl der Räder = 2). Auch **geerbte (generische) Werte** können überschrieben werden.

## Problemlose Behandlung von **Ausnahmen**:

Reguläre Werte:

- Standardwerte/Defaults,
- geerbte (generische) Werte

Bei Ausnahme:

- Überschreiben des regulären Wertes

## Faktorisierung von Wissen

Die Idee framebasierter Systeme geht davon aus, daß man Wissen innerhalb vorgegebener Strukturen sukzessive erweitern und aktualisieren kann.

Es gibt "generische" und konkrete Frames, die zwar die gleiche Struktur, aber verschiedene Werte haben:

- was gehört dazu → Schema
- was ist normal → defaults
- wie berechnen → prozedurales Wissen
- allgemeine Eigenschaften → Vererbung.

## Begriffe zur Darstellung eines Frames

**Frame**: Liste von Slots.

**Slot(Schlitz)** sind gegliedert in Facetten.

Eine **Facette** nimmt Füllsel (fillers) auf.

Ein Slot hat mindestens zwei Facetten: Slot-Name und Slot-Wert, ggf. weitere, z.B. für Defaultwert.

## Gebräuchliche Facetten (I)

**Value:** eigentlicher Slotwert, speziell auch als Verweis(e) auf andere Frames:

- instance-of, is-a
- ako
- part-of bzw. has-part
- supersets (instance-of, is-a, ako)
- subsets (Nachfolger in der Hierarchie)
- Parallelverweise (Verheiratet mit ...)

**Defaultwerte** (des Frames)

**Generische Werte** (für Vererbung)

## Gebräuchliche Facetten (II)

**Slotbedingungen:** zulässige Wertebereiche

**Prozeduren** (Dämonen):

- if-needed (Berechnungen, „Lese-Prozeduren“, z.B. Alter aus Geburtsdatum und aktueller Zeit)
- if-added (Aktionen, „Schreib-Prozeduren“) in Verbindung mit belegen/ändern eines Slotwertes, z.B. bei Änderung des Familienstandes auch Steuerklasse ändern („aktive Datenbank“)

**Vererbungshinweise** (Vererbungsrolle)

## Vergleich mit Semantischen Netzen

*Relationen* zwischen Objekt und Slotwert: Slotname ist Name der Relation.

*Relationen* zwischen Frames: über die entsprechenden Einträge (is-a usw.)

**Organisation frame-basierter Systeme:**

- Frames analog Records, Pointer für Beziehungen zw. Frames
- Arbeitsablauf:
  - Auswahl (Vorgabe) des zu bearbeitenden Frames
  - Ausfüllen der Slots (Vererbung, Berechnung),
  - dabei ggf. weitere Aktionen (Dämonen) aktivieren.

## Beispiele für framebasierte Systeme

**Medizinische Diagnose**

Die Diagnose zielt auf Ausfüllen der Slots, um eine Diagnose und Therapie zu bekommen. Dazu notwendig ist der Eintrag von Symptomwerten.

**Lagerhaltung**

Ein beständiger Prozeß der Aktualisierung löst Bestellungen aus etc.