

8 Fallstudien

Jeweils ein Gegenstandsbereich steht im Vordergrund

Seine Strukturen, Eigenschaften, Zusammenhänge werden mit verschiedenen Kalkülen modelliert.

Verschiedene Kalküle werden eingesetzt, um

- unterschiedliche Aspekte zu beschreiben
- Beschreibungen derselben Aspekte zu vergleichen.

Fallstudie 1: Autowerkstatt

Fallstudie 2: Monopoly - Spiel

Fallstudie 3: Getränkeautomat (siehe Übungen)

Fallstudie 1: Autowerkstatt

Wir modellieren die **Auftragsabwicklung in einer Autowerkstatt**.

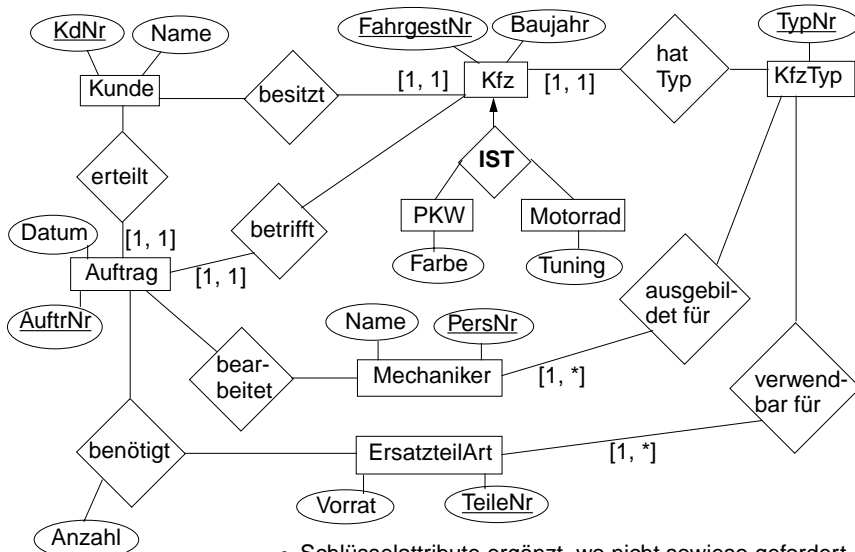
Ziel: Datenbank entwerfen, Abläufe analysieren und verbessern

- Teilaufgaben:**
1. Informationen und Zusammenhänge
 2. Bedingungen und Regeln
 3. Abläufe bei der Auftragsabwicklung

Kurzbeschreibung der Informationsstruktur:

1. **Kunde:** hat einen Namen, besitzt Kraftfahrzeuge, erteilt Aufträge
2. **Auftrag:** hat Eingangsdatum, betrifft ein Kraftfahrzeug, wird von Mechanikern bearbeitet, benötigt Ersatzteile bestimmter Arten und Mengen
3. **Kraftfahrzeug:** hat Fahrgestellnummer und Baujahr, ist entweder ein PKW oder ein Motorrad; zu PKWs interessiert ihre Farbe, zu Motorrädern der Tuningsatz
4. **Typ:** Kraftfahrzeug hat einen Typ, Mechaniker ist für einige Typen ausgebildet, Ersatzteil ist für bestimmte Typen verwendbar

8.1.a Informationsstruktur als ER-Modell



- Schlüsselattribute ergänzt, wo nicht sowieso gefordert
- Kardinalitäten: plausibel ohne unnötig einzuschränken

Vergleich ER-Modell und Wertebereiche

Attribut	Vorrat	Vorrat := \mathbb{N}_0	Wertemenge
Schlüsselattribut	KdNr	KdNr := \mathbb{N}_0	Indexmenge
Entity-Typ	Kunde	Kunde := $\text{KdNr} \times \text{Name}$	kartesisches Produkt ohne Identität der Entities
Relation	Kunde - erteilt - Auftrag	erteilt := $\text{Pow}(\text{Kunde} \times \text{Auftrag})$	Relation
Kardinalität	[1, 3]	Prädikatenlogik: $\forall a \in \text{Auftrag}: 1 \leq \{(x,y) \mid (x,y) \in \text{erteilt} \wedge y=a\} \leq 3$	
	[1, 1]	erteilt: Auftrag -> Kunde	Funktion (hier: total)
IST-Beziehung	Kfz	Kfz := FahrgestNr × KfzVarianten	kartesisches Produkt und disjunkte Vereinigung:
	PKW	KfzVarianten := { istPKW, p } p ∈ Farbe ∪ { istMotorrad, m } m ∈ Tuning	

8.1.b Bedingungen

Ein Auftrag soll von höchstens 3 Mechanikern bearbeitet werden:

ER Kardinalität:



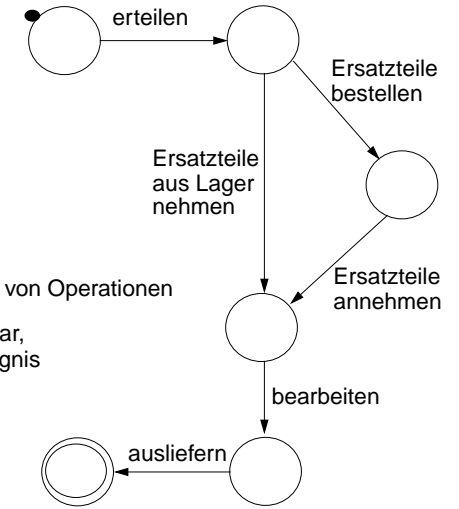
Prädikatenlogik: $\forall a \in \text{Auftrag}: 0 \leq |\{(x,y) \mid (x,y) \in \text{bearb.} \wedge x=a\}| \leq 3$

Ein Auftrag soll nur dann angenommen werden, wenn für den betreffenden KfzTyp auch Mechaniker ausgebildet sind.

Prädikatenlogik: $\forall a \in \text{Auftrag}: \forall k \in \text{Kfz}: \forall t \in \text{KfzTyp}: ((a, k) \in \text{betrifft} \wedge (k, t) \in \text{hatTyp}) \rightarrow \exists m \in \text{Mechaniker}: (m, t) \in \text{ausgebildet}$

8.1.c Ablauf der Auftragsbearbeitung (DEA)

- Auftrag wird **erteilt**,
- Verfügbarkeit der Ersatzteile **geprüft**,
- ggf. **bestellt**,
- von einem Mechaniker **bearbeitet**,
- Kraftfahrzeug wird dem Kunden **ausgeliefert**.



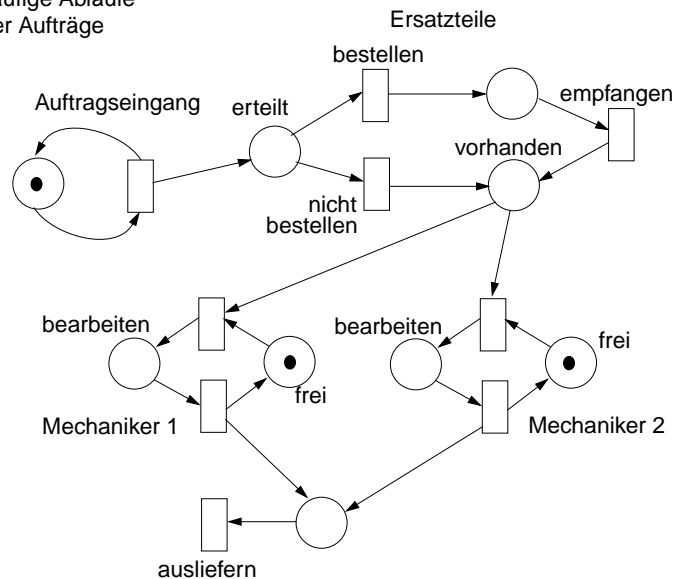
Deterministischer, endlicher Automat beschreibt streng **sequentielle Abfolge** von Operationen

Auch als **Abhängigkeitsgraph** interpretierbar, hier: Kante ist Operation, Knoten ist Ereignis

1.c Ablauf der Auftragsbearbeitung (Petri-Netz)

Petri-Netz

modelliert nebenläufige Abläufe
Durchlauf mehrerer Aufträge



Mechaniker konkurrieren um Aufträge:

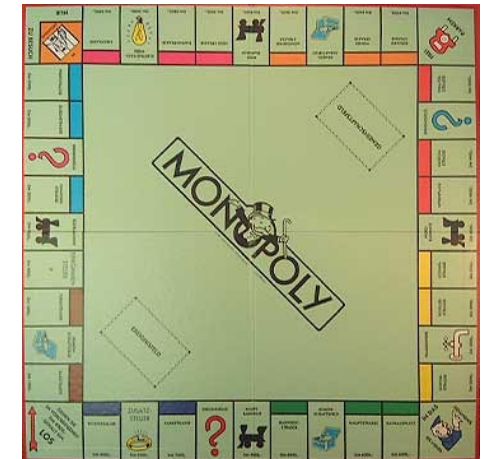
Fallstudie 2: Monopoly-Spiel

Wir modellieren Struktur und Ablauf des Monopoly-Spiels.

Ziel: Spielregeln präzisieren und formalisieren

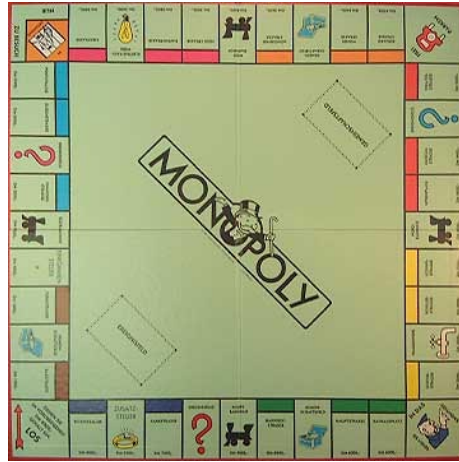
Teilaufgaben:

1. Informationen und Zusammenhänge
2. Bedingungen und Regeln
3. Spielabläufe

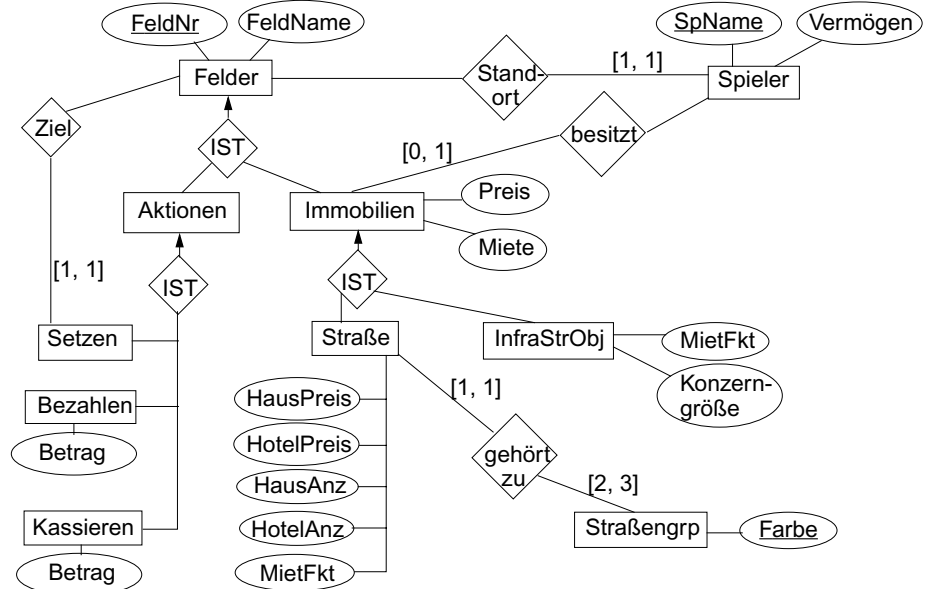


Kurzbeschreibung der Informationsstruktur

- Spieler:**
hat einen Namen, steht auf einem Spielfeld, hat Vermögen, besitzt Immobilien
- Feld:**
hat Nummer und Namen, ist entweder ein Aktionsfeld oder eine Immobilie
- Immobilie:**
hat einen Preis und kostet Miete, ist entweder eine Straße oder ein Infrastrukturobjekt
- Straße:**
hat Preise und Anzahl für Häuser und Hotels sowie Funktion zur Berechnung der Miete
- Infrastrukturobjekt:**
hat Konzerngröße und eine Funktion zur Berechnung der Miete
- Aktionsfeld:**
fordert auf zum Bezahlen oder Kassieren eines Betrages oder zum Setzen auf ein Feld
- Straßengruppe:**
2 oder 3 Straßen werden zu einer Gruppe mit gleicher Farbe zusammengefasst



8.2.a Informationsstruktur als ER-Modell



Einige Wertebereiche zur Informationsstruktur

- FeldNr := { 1, 2, ..., 40 }
- FeldArten := { istAktion, istImmobilie }
- Felder := FeldNr × FeldName × FeldVarianten
- FeldVarianten := { (istAktion, a) | a ∈ Aktionen } ∪ { (istImmobilie, i) | i ∈ Immobilien }
- AktionsArten := { istSetzen, istBezahlen, istKassieren }
- Aktionen := { (istSetzen) } ∪ { (istBezahlen, b) | b ∈ Betrag } ∪ { (istKassieren, b) | b ∈ Betrag }
- Betrag := \mathbb{N}_0
- ImmobilienArten := { istStraße, istInfraStrObj }
- Immobilien := Preis × Miete × ImmobilienVarianten
- ImmobilienVarianten := { (istStraße, s) | s ∈ Straße } ∪ { (istInfraStrObj, i) | i ∈ InfraStrObj }
- Straße := HausPreis × HotelPreis × HausAnzahl × HotelAnzahl × MietFkt
- besitzt := FeldNr → SpName

Beispiele für Felder:

- (1, Los, (istAktion, (istKassieren, 4000))) ∈ Felder
- (2, BadStraße, (istImmobilie, 1200, 40, (istStraße, 1000, 1000, 0, 0, MFkt2))) ∈ Felder
- (6, Südbahnhof, (istImmobilie, 4000, 1000, (istInfraStrObj, 2, MFktBhf))) ∈ Felder

8.2.b Bedingungen

Die **Miete einer Straße steigt** je intensiver sie **bebaut** ist;
die **Miete eines Infrastrukturobjektes steigt** je mehr **gleichartige Objekte** ein Spieler besitzt.

$$\forall x \in \text{Immobilien: } \forall p \forall m \forall hap \forall hop \forall haanz \forall hoanz \forall n \forall g$$

$$[x = (p, m, (\text{istStraße}, hap, hop, haanz, hoanz, f)) \rightarrow m = f(haanz, hoanz)] \wedge$$

$$[x = (p, m, (\text{istInfraStrObj}, n, g)) \rightarrow m = g(n)]$$

Eine Straße darf nur dann **bebaut** werden, wenn der Besitzer **alle Straßen dieser Gruppe** besitzt.

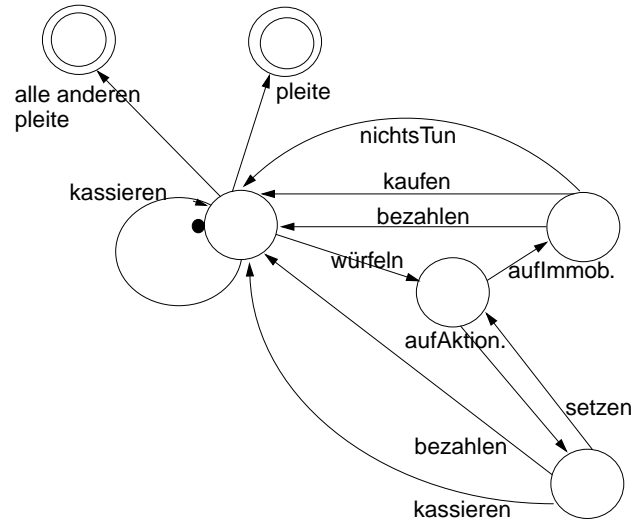
$$\forall x \in \text{Felder: } \forall nr \forall name \forall p \forall m \forall hap \forall hop \forall haanz \forall hoanz \forall h$$

$$x = (nr, name, (\text{istImmobilie}, p, m, (\text{istStraße}, hap, hop, haanz, hoanz, h))) \rightarrow$$

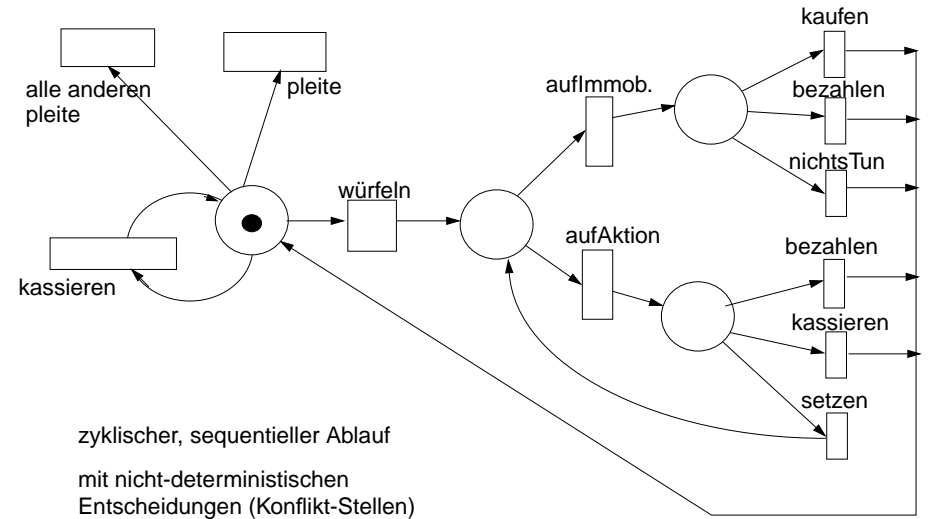
$$(haanz + hoanz > 0 \wedge \exists f \in \text{Farbe: } (x, f) \in \text{gehörtZu} \wedge \exists s \in \text{Spieler: } (s, x) \in \text{besitzt}$$

$$\rightarrow \forall g \in \text{Felder: } (g, f) \in \text{gehörtZu} \rightarrow (s, g) \in \text{besitzt}$$

2.c Aktionsfolgen eines Spielers (DEA)



8.2.c Aktionsfolgen eines Spielers (Petri-Netz)



zyklischer, sequentieller Ablauf
mit nicht-deterministischen
Entscheidungen (Konflikt-Stellen)