Vorlesung Semantic Web

TECHNISCHE UNIVERSITÄT DARMSTADT

Vorlesung im Wintersemester 2011/2012

Dr. Heiko Paulheim

Fachgebiet Knowledge Engineering

Aufgabe 1

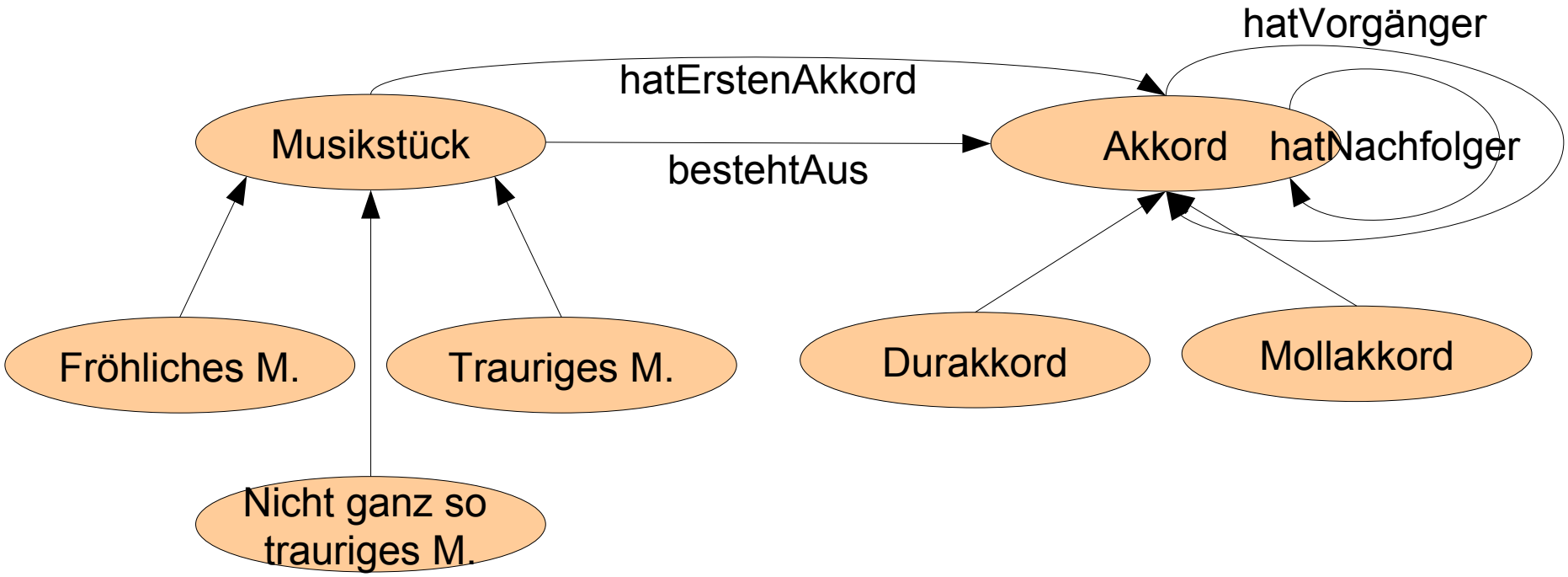
TECHNISCHE UNIVERSITÄT DARMSTADT

- Modellieren Sie die folgende Ontologie mit Hilfe von OWL und SWRL:
  - Musikstücke bestehen aus Akkorden, die aufeinander folgen (jedes Stück hat einen ersten Akkord, und jeder Akkord ein einen eindeutigen Vorgänger und Nachfolger). Es gibt Dur- und Mollakkorde (und nur diese). Ein fröhliches Stück hat nur Durakkorde, ein trauriges Stück hat nur Mollakkorde. Bei einem nicht ganz so traurigen Stück gibt es nie zwei Mollakkorde hintereinander.

Aufgabe 1

TECHNISCHE UNIVERSITÄT DARMSTADT

- Grundontologie – Klassen und Relationen



Aufgabe 1

TECHNISCHE UNIVERSITÄT DARMSTADT

- Vorgänger und Nachfolger von Akkorden

```
:hatVorgänger owl:inversePropertyOf :hatNachfolger .
:hatErstenAkkord rdfs:subPropertyOf :bestehtAus .
:hatVorgänger a owl:FunctionalProperty,
                owl:InverseFunctionalProperty .
:hatNachfolger a owl:FunctionalProperty,
                 owl:InverseFunctionalProperty .
bestehtAus(?m,?a) ∧ hatNachfolger(?a,?an)
→ bestehtAus(?m,?an) .
```

- Akkordarten

```
:Akkord owl:disjointUnionOf (:Durakkord :Mollakkord) .
```

Aufgabe 1

TECHNISCHE UNIVERSITÄT DARMSTADT

- Fröhliche Stücke

```
:FröhlichesStück owl:equivalentClass [
    owl:intersectionOf (:Musikstück [
            a owl:Restriction;
            owl:onProperty :bestehtAus ;
            owl:allValuesFrom :Durakkord ]) ].
```

Aufgabe 1

TECHNISCHE UNIVERSITÄT DARMSTADT

- Folgt damit aus

  :Musikstück1 :bestehtAus :CDur, FDur, GDur .

  dass :Musikstück1 ein fröhliches Musikstück ist?

- Nein, denn wir haben Open World Assumption
  - unser Modell von Musikstück1 ist also unvollständig!

- So geht's:

```
:Musikstück1 a [
  a owl:Restriction ;
  owl:onProperty :bestehtAus ;
  owl:allValuesFrom [
    owl:oneOf (:CDur :FDur :GDur) ] ] .
```

Aufgabe 1

TECHNISCHE UNIVERSITÄT DARMSTADT

- Traurige Stücke gehen genauso
- Jetzt noch die nicht ganz so traurigen
- Richtung 1:

```
NichtSoTraurigesMusikstück(?m) ∧ bestehtAus(?m,?a)
∧ folgtAuf(?m,?an) ∧ Mollakkord(?a)
→ Durakkord(?an)
```

- Richtung 2: Hier brauchen wir eine zweite Klasse

```
:Musikstück owl:disjointUnionOf
  ( :NichtSoTraurigesMusikstück,
    :EinigermaßenTraurigesMusikstück) .
Musikstück(?m) ∧ bestehtAus(?m,?a) ∧ bestehtAus(?m,?a1)
∧ folgtAuf(?a,?a1) ∧ Mollakkord(?a) ∧ Mollakkord(?a1)
→ EinigermaßenTraurigesMusikstück(?m)
```

Aufgabe 2

TECHNISCHE UNIVERSITÄT DARMSTADT

- Gegeben ist folgende F-Logic-Ontologie:

```
Zelt::Unterkunft .
Hotel::Unterkunft .
DubioseUnterkunft::Unterkunft .
LuxusUnterkunft::Unterkunft.
Unterkunft[hatSterne{0,1} *=>integer].
```

- Definieren Sie nun folgende Regeln:
  - Alle Zelte haben nur einen Stern.
  - Luxusunterkünfte sind Unterkünfte mit mindestens vier Sternen.
  - Dubiose Unterkünfte sind solche, bei denen die Angabe der Sterne fehlt.

Aufgabe 2

TECHNISCHE UNIVERSITÄT DARMSTADT

- Alle Zelte haben nur einen Stern

```
?X[hatSterne->1] :- ?X:Zelt.
```

- Luxusunterkünfte sind Unterkünfte mit mindestens vier Sternen.

```
?X:Luxusunterkunft :- ?X:Unterkunft[hatSterne->?Y]
                      AND (?Y>=4).
```

- Dubiose Unterkünfte sind solche, bei denen die Angabe der Sterne fehlt.

```
?X:DubioseUnterkunft :- ?X:Unterkunft
                        AND NOT (EXIST ?Y ?X[hatSterne->?Y]).
```

- Jetzt live in OntoStudio...

Aufgabe 2

TECHNISCHE UNIVERSITÄT DARMSTADT

- Geben Sie eine Stratifizierung der Ontologie an.
- Folgende Ungleichungen müssen gelten
- aus der Ontologie:
  - S(Zelt) ≤ S(Unterkunft) – wegen Unterkunft(x) ← Zelt(x)
  - S(Hotel) ≤ S(Unterkunft)
  - S(LuxusUnterkunft) ≤ S(Unterkunft)
  - S(DubioseUnterkunft) ≤ S(Unterkunft)
- aus den Regeln:
  - S(Zelt) ≤ S(hatSterne)
  - S(hatSterne) ≤ S(Luxusunterkunft)
  - S(hatSterne) < S(DubioseUnterkunft)

Aufgabe 2

TECHNISCHE UNIVERSITÄT DARMSTADT

- Damit ergibt sich als **eine** mögliche Stratifikation
  - Schicht 0: hatSterne, Zelt
  - Schicht 1: Unterkunft, Hotel, LuxusUnterkunft, DubioseUnterkunft

Aufgabe 2

TECHNISCHE UNIVERSITÄT DARMSTADT

- Welche der Regeln könnten Sie auch in OWL modellieren, welche in SWRL?

- Alle Zelte haben nur einen Stern
  - das geht mit OWL:

```
:Zelt rdfs:subClassOf [
  a owl:Restriction;
  owl:onProperty hatSterne;
  owl:hasValue 1^^xsd:integer ] .
```

Aufgabe 2

TECHNISCHE UNIVERSITÄT DARMSTADT

- Welche der Regeln könnten Sie auch in OWL modellieren, welche in SWRL?

- Luxusunterkünfte sind Unterkünfte mit mindestens vier Sternen
  - dafür brauchen wir SWRL (für integer-Vergleich)

    `Unterkunft(?u) ∧ hatSterne(?u,?s)`
    `∧ swrlb:greaterThanOrEqual(?s,4) → LuxusUnterkunft(?u)`

- Dubiose Unterkünfte sind solche, bei denen die Angabe der Sterne fehlt
  - das bekommen wir in OWL/SWRL so nicht modelliert
  - das Fehlen einer Angabe zu prüfen, hat unter Open World Assumption auch wenig Sinn

Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

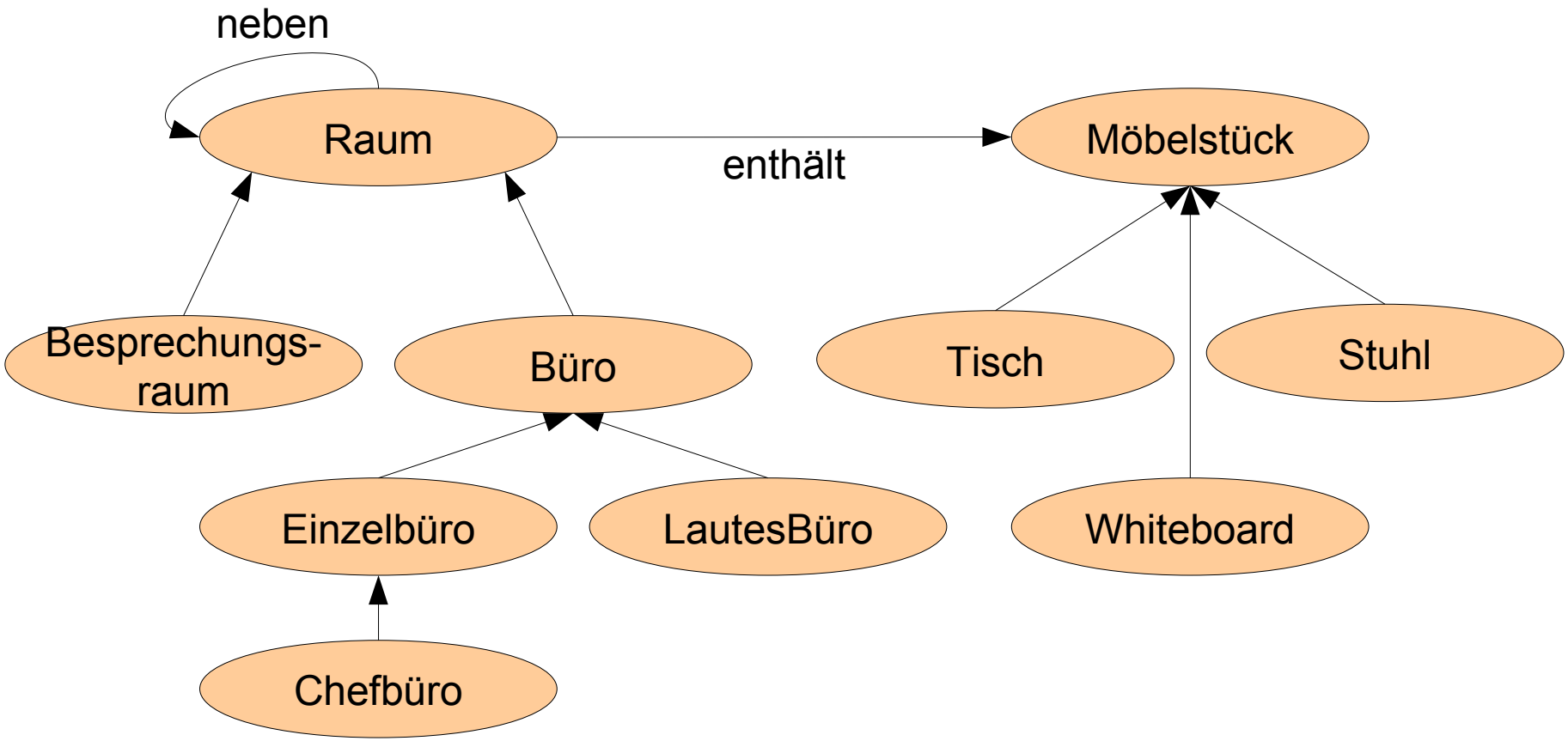
- Modellieren Sie folgende Ontologie für ein Hausverwaltungssystem in OWL:

Räume können Büros und Besprechungsräume sein. Räume enthalten Möbel; Möbel sind Tische, Stühle und Whiteboards. Jedes Büro enthält mindestens einen Tisch und einen Stuhl; jedes Besprechungszimmer enthält mindestens ein Whiteboard. Einzelbüros enthalten genau einen Tisch und genau einen Stuhl. Büros neben Besprechungszimmern sind laute Büros. Chefbüros sind Einzelbüros, die keine lauten Büros sind.

Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

- Grundontologie – Klassen und Relationen



Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

- Jedes Büro enthält mindestens einen Tisch und einen Stuhl; jedes Besprechungszimmer enthält mindestens ein Whiteboard

```
:Büro rdfs:subClassOf [
  a owl:Restriction ;
  owl:onProperty :enthält ;
  owl:someValuesFrom :Tisch ] .


...
```

Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einzelbüros enthalten genau einen Tisch und genau einen Stuhl.
  - das geht nur in OWL2 (qualifizierte Restriktionen):

```
:Einzelbüro rdfs:subClassOf [
    a owl:Restriction ;
    owl:onProperty :enthält;
    owl:qualifiedCardinality "1"^^xsd:integer ;
    owl:onClass :Tisch ] .

...
```

Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

- Büros neben Besprechungszimmern sind laute Büros

```
:LautesBüro owl:equivalentClass [
    owl:intersectionOf ( :Büro [
            owl:Restriction ;
            owl:onProperty :neben ;
            owl:someValuesFrom :Besprechungsraum ] ) ] .
```

- Warum equivalentClass?
- Und warum die intersection-Konstruktion?

Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

- Chefbüros sind Einzelbüros, die keine lauten Büros sind.
- Einzelbüros sind sie schon per Definition
- keine lauten Büros:

```
:Chefbüro rdfs:subClassOf [
    owl:complementOf :LautesBüro ] .
```

- Was wäre die Semantik, wenn wir owl:equivalentClass verwenden würden?

Aufgabe 3

TECHNISCHE UNIVERSITÄT DARMSTADT

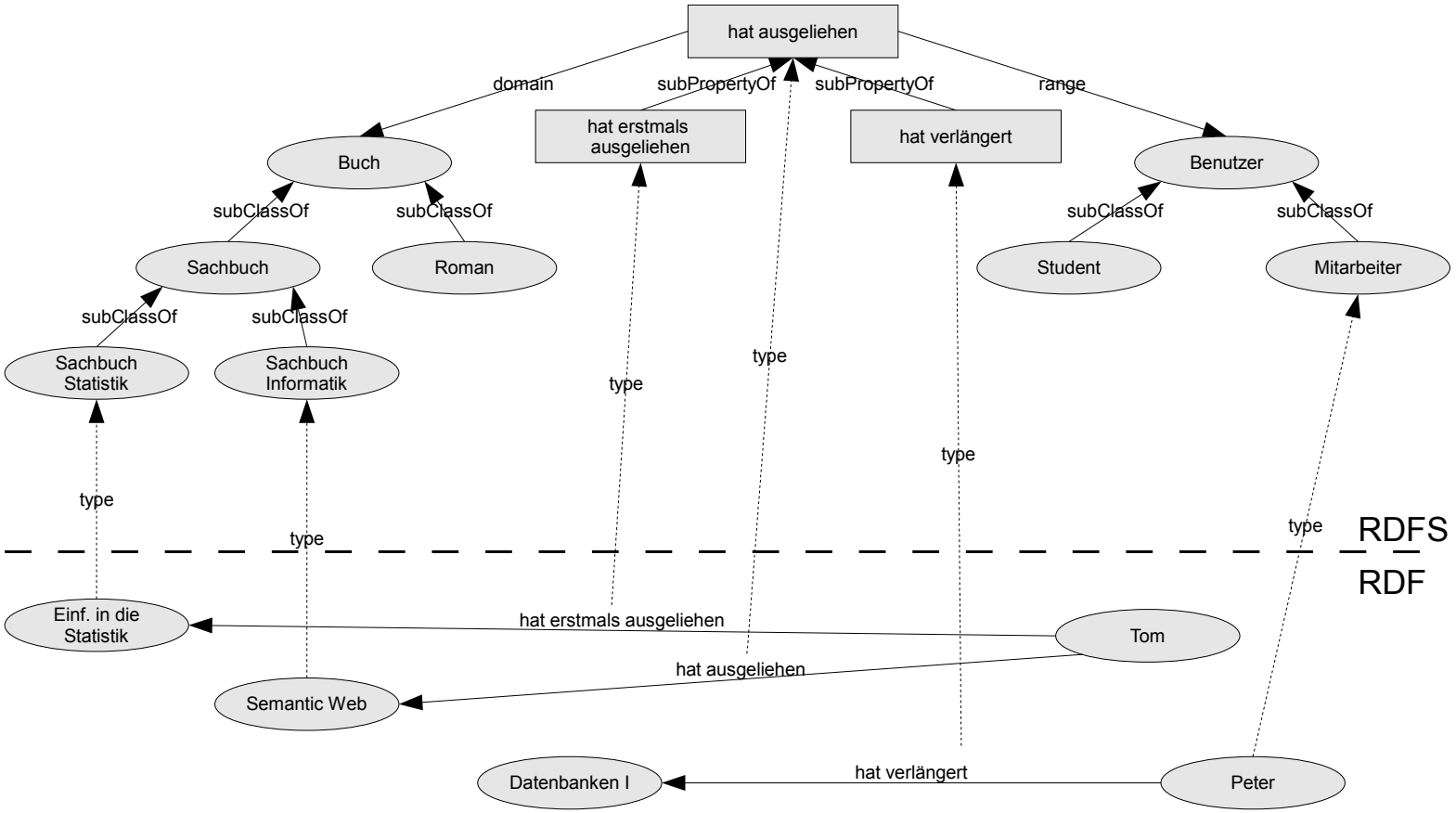
- Komplexität der Ontologie
  - Subklassen, Komplementklassen, Intersection, someValuesFrom → S
  - Qualifizierte Restriktionen → Q

- Mehr haben wir nicht verwendet, also ist die Komplexität SQ

Aufgabe 4

TECHNISCHE UNIVERSITÄT DARMSTADT

- Gegeben folgende Ontologie:



Aufgabe 4

TECHNISCHE UNIVERSITÄT DARMSTADT

- Welche der folgenden Aussagen erhält man...
  - allein aus den RDF-Fakten
  - allein aus dem RDF-Schema
  - nur aus der Kombination von RDF-Fakten und RDF-Schema
  - gar nicht

  - Tom hat "Semantic Web" ausgeliehen.
  - Tom hat "Einführung in die Statistik" ausgeliehen.
  - Jedes Sachbuch ist auch ein Buch.
  - Jedes Buch kann nur von einem Benutzer ausgeliehen werden.
  - Tom ist ein Benutzer.
  - Peter ist kein Student.
  - Ein Sachbuch ist kein Roman.
  - Wer ein Buch verlängert hat, hat es auch ausgeliehen.

Aufgabe 4

TECHNISCHE UNIVERSITÄT DARMSTADT

- Allein aus RDF-Fakten:
  - Tom hat "Semantic Web" ausgeliehen.
- Allein aus RDF-Schema:
  - Jedes Sachbuch ist auch ein Buch.
  - Wer ein Buch verlängert hat, hat es auch ausgeliehen.
- Aus Kombination von RDF-Fakten und -Schema:
  - Tom hat "Einführung in die Statistik" ausgeliehen.
  - Tom ist ein Benutzer.

Aufgabe 4

TECHNISCHE UNIVERSITÄT DARMSTADT

- Wie folgen diese Aussagen?
- RDF-S-Ableitungsregeln!

```
:Tom :hatErstmalsAusgeliehen :EinführungInDieStatistik .
:hatErstmalsAusgeliehen rdfs:subPropertyOf
:hatAusgeliehen .
→ :Tom :hatAusgeliehen :EinführungInDieStatistik .

:Tom :hatAusgeliehen :SemanticWeb .
:hatAusgeliehen rdfs:range :Benutzer .
→ :Tom a :Benutzer .
```

Aufgabe 4

TECHNISCHE UNIVERSITÄT DARMSTADT

- Geben Sie für alle Aussagen, für die Sie "gar nicht" geantwortet haben, OWL-Axiome an, die diese Schlussfolgerung ermöglichen würden.

- Jedes Buch kann nur von einem Benutzer ausgeliehen werden.

```
:hatAusgeliehen a owl:InverseFunctionalProperty .
```

- Peter ist kein Student.

```
:Mitarbeiter owl:disjointWith :Student .
```

- Ein Sachbuch ist kein Roman.

```
:Sachbuch owl:disjointWith :Roman .
```

Vorlesung Semantic Web

TECHNISCHE UNIVERSITÄT DARMSTADT

Vorlesung im Wintersemester 2011/2012

Dr. Heiko Paulheim

Fachgebiet Knowledge Engineering