Wirtschaftsinformatik II – Meilicke/Stuckenschmidt

Inferenz für Beschreibungslogik

Tableauverfahren

ONTOLOGIEN

INFERENZ

Inferenz - Wozu?

1. Überprüfen von Kohärenz
   – Sind alle Konzepte in meiner T-Box erfüllbar? Habe ich vielleicht einen Fehler bei der Modellierung gemacht?
   – Siehe Library Beispiel letzter Foliensatz

2. Überprüfen von Konsistenz
   – Werden die Konzepte und Rollen korrekt verwendet? Gibt es Daten (Instanzen), die in einer widersprüchlichen Weise beschrieben sind?

3. Ableiten von neuem Wissen
   – Folgt eine Behauptung aus meiner KB bzw. ist diese bereits „implizit“ in meiner KB enthalten?

Inferenz - Wie?

1. Überprüfen von Kohärenz
   - Ob Konzept C erfüllbar ist, kann geprüft werden durch den Versuch ein Modell zu konstruieren für $KB \cup C(a)$
     - Wobei a eine Instanz ist, die nicht in KB vorkommt

2. Überprüfen von Konsistenz
   - Gibt es ein (beliebiges) Modell für KB?

3. Ableiten von neuem Wissen
   - $C(a)$ folgt aus KB genau dann, wenn $KB \cup \neg C(a)$ unerfüllbar ist
   - $C \sqsubseteq D$ folgt aus KB genau dann, wenn $KB \cup C(b) \cup \neg D(b)$ unerfüllbar ist
     - Wobei b eine Instanz ist, die nicht in KB vorkommt

Inferenz - Wie genau?

- Wir müssen also ein Verfahren anwenden, was uns ein Modell konstruiert, wenn es ein solches Modell gibt

- Aussagenlogik => Tableauverfahren
  - Benötigt Negationsnormalform
  - Durch wiederholte Anwendung einfacher Regeln wird ein Baum aufgebaut
  - Jeder Zweig entspricht einer Interpretation
  - Alle Blätter widersprüchlich, dann gibt es kein Modell
  - Ein Blatt widerspruchsfrei, dann gibt es ein Modell

- Wir benötigen ein ähnliches Verfahren für Beschreibungslogik!

Tableau für Aussagenlogik

$$((a \land b) \lor (\neg b \land c)) \land (\neg a \land \neg c)$$

$$(a \land b) \lor (\neg b \land c)$$
$$\neg a \land \neg c$$

$$(a \land b) \lor (\neg b \land c)$$
$$\neg a$$
$$\neg c$$

$$a \land b$$
$$\neg a$$
$$\neg c$$

$$a$$
$$b$$
$$\neg a$$
$$\neg c$$

$$\neg b \land c$$
$$\neg a$$
$$\neg c$$

$$\neg b$$
$$c$$
$$\neg a$$
$$\neg c$$

Also gibt es kein Modell für die Formel an der Wurzel des Baums

Also folgt $a \lor c$ aus $(a \land b) \lor (\neg b \land c)$!

Tableau für Beschreibungslogik

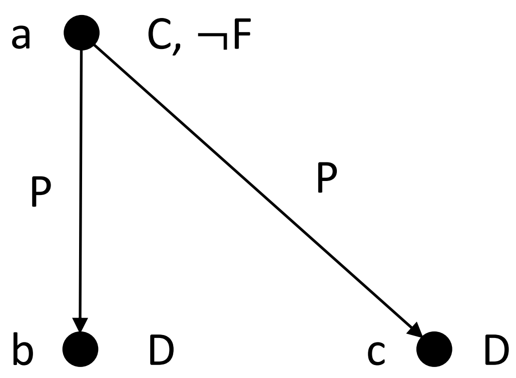
- Alle Regeln basieren auf der Idee, die ABox A systematisch durch die Axiome der TBox zu erweitern

- Beispiel $C(a), C \sqsubseteq \mathrm{D} \Rightarrow C(a), D(a)$

- Unterschiede zum Tableau bei Ausagenlogik:
  - Baum wird unübersichtlich, man beginnt mit der ABox A, bei einer Verzweigung macht man z.B. mit A_1 und A_2 weiter
  - Für einen Knoten (= Kasten) im aussagenlogischen Tableau zeichnet man einen Graph dessen Knoten Instanzen sind

- Wir führen das Verfahren an einem Beispiel ein, bevor wir die Regeln im allgemeinen notieren

Notation im Tableauverfahren



ABox *A*

$C(a)$
$\neg F(a)$
$P(a,b)$
$D(b)$
$P(a,c)$
$D(c)$

Prinzipielle Vorgehensweise

- Es seien C und D Konzepte: Für jedes Axiom der Form $C \sqsubseteq D$ füge jedem Knoten, der das Label C hat zusätzlich das Label D hinzu
  - Dabei kann D auch ein komplexes Konzept sein
- Es seien P und Q Rollen: Für jedes Axiom der Form $P \sqsubseteq Q$ füge jeder Kante, die das Label P hat zusätzlich das Label Q hinzu
- Für jede komplexe Konzeptbeschreibung, die als Label an einem Knoten steht, führe die auf den folgenden Folien genannten Regeln aus
- Tue dies solange bis keine der Regeln mehr angewendet werden kann oder bis ein Widerspruch in jedem der Äste aufgetreten ist
  - Was ein Widerspruch ist, siehe nächste Folien

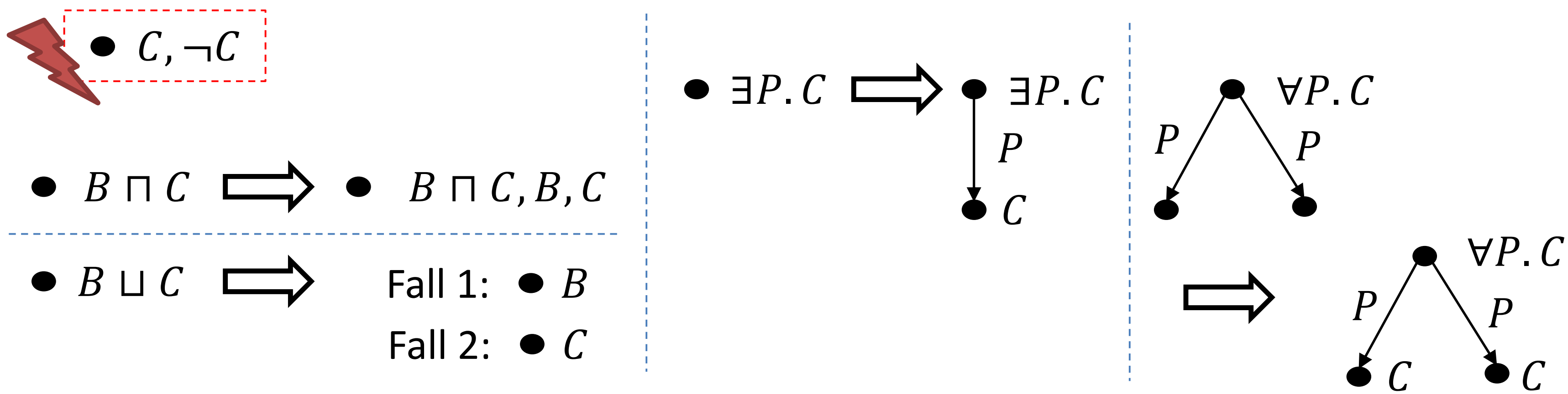
Erinnerung I

Konzept / Rolle	Interpretation	Bedeutung
$\neg C^I$	$\Delta \setminus C^I$	Negation
$(B \sqcap C)^I$	$B^I \cap C^I$	Konjunktion
$(B \sqcup C)^I$	$B^I \cup C^I$	Disjunktion
$(\exists P.C)^I$	$\{x \mid \exists y \langle x,y\rangle \in P^I \wedge y \in C^I\}$	**Existenz Restriktion**
$(\forall P.C)^I$	$\{x \mid \forall y \langle x,y\rangle \in P^I \rightarrow y \in C^I\}$	Value Restriktion

- Wir betrachten nun nur die Ausdruckstärke, die in folgender Tabelle festgelegt ist
  - D.h. wir definieren das Verfahren nur für die Arten von komplexen Ausdrücken, die in der Tabelle auftauchen

Schlussfolgerungen

$\neg C^I$	$\Delta \setminus C^I$	Das, was nicht C ist
$(B \sqcap C)^I$	$B^I \cap C^I$	Das, was sowohl B als auch C ist
$(B \sqcup C)^I$	$B^I \cup C^I$	Das, was B oder C (oder beides) ist
$(\exists P.C)^I$	$\{x \mid \exists y\, \langle x,y \rangle \in P^I \wedge y \in C^I\}$	Das, was zu einem C in Relation R steht
$(\forall P.C)^I$	$\{x \mid \forall y\, \langle x,y \rangle \in P^I \rightarrow y \in C^I\}$	Das, was nur zu C-artigem in Relation P steht



Beispiel

- Ist das Konzept MadCow erfüllbar?

TBox T

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.\neg(Animal \sqcup PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

ABox A

$MadCow(c)$

- Füge $MadCow(c)$ zu A hinzu und versuche ein Modell für T und die erweiterte ABox A zu konstruieren!

Beispiel

c • $MadCow$ ⇐ $MadCow(c)$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c • $MadCow, Cow$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c • $MadCow, Cow, Vegetarian$

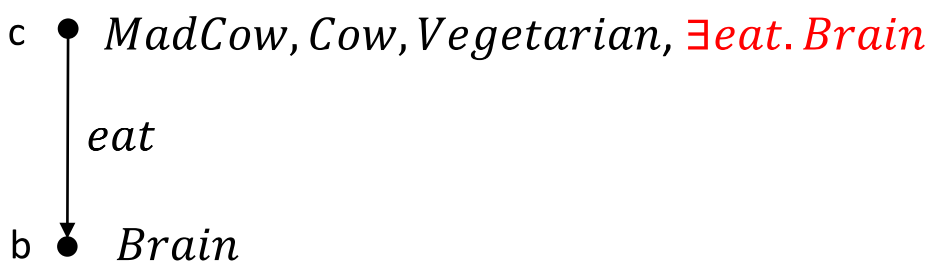
(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c • $MadCow, Cow, Vegetarian, \exists eat.Brain$

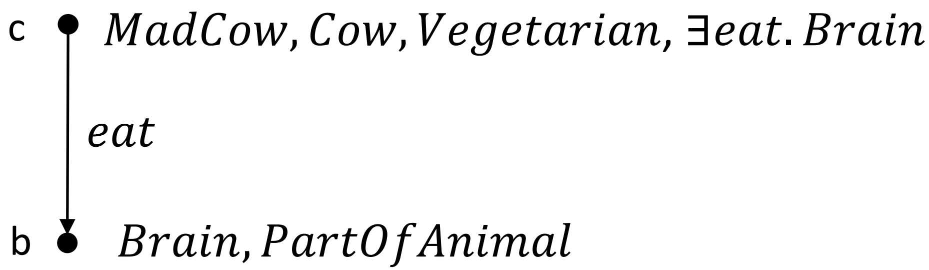
(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel



(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel



(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c $MadCow, Cow, Vegetarian, \exists eat.Brain, \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$

$$\Big\downarrow eat$$

b $Brain, PartOfAnimal$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c $MadCow, Cow, Vegetarian, \exists eat.Brain, \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$

$\downarrow eat$

b $Brain, PartOfAnimal, \neg Animal \sqcap \neg PartOfAnimal$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c: $MadCow, Cow, Vegetarian, \exists eat.Brain, \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$

$\downarrow$ eat

b: $Brain, PartOfAnimal, \neg Animal \sqcap \neg PartOfAnimal, \neg Animal, \neg PartOfAnimal$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Beispiel

c $\bullet$ $MadCow, Cow, Vegetarian, \exists eat.Brain, \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$

$\downarrow$ eat

b $\bullet$ $Brain, PartOfAnimal, \neg Animal \sqcap \neg PartOfAnimal, \neg Animal, \neg PartOfAnimal$

$PartOfAnimal$
$\neg PartOfAnimal$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Also: MadCow ist nicht erfüllbar, die KB ist inkohärent

Allgemeine Regeln

- Vereinfachende Notation
  - $x: C$ steht für $C(x)$
  - der Pfeil $\Rightarrow$ steht für „füge hinzu“
- Subsumptionsregel
  - $x: C, C \sqsubseteq D \Rightarrow x: D$
- „Und“-Regel
  - $x: C \sqcap D \Rightarrow x: C, x: D$
- „Oder“-Regel
  - $x: C \sqcup D \Rightarrow$
    - Fall 1: $x: C$
    - Fall 2: $x: D$
- $\exists$-Regel
  - $x: \exists r. C \Rightarrow r(x, y), y: C$
  - wobei y eine neue Instanz ist
- $\forall$-Regel
  - $x: \forall r. C, r(x, y) \Rightarrow y: C$

Diese Regeln sind nur für die Beschreibungslogik ALC vollständig. Das ist die Variante der Beschreibungslogik, bei der nur die Konstruktoren erlaubt sind, die in der Tabelle in diesem Foliensatz aufgeführt sind. Für ausdruckstärkere Varianten gibt es erweiterte Regelsätze. Das Prinzip bleibt aber das gleiche.

Normalisierung

- Die genannten Regeln sind nur dann anwendbar, wenn Negationsnormalform (NNF)vorliegt
  - NNF liegt vor, wenn Negation nur vor atomaren Konzepten steht
  - Dies war der Fall in unserem Beispiel
- Ausserdem müssen $\equiv$ Axiome in $\sqsubseteq$ Axiome zerlegt werden

- Äquivalenzumformungen, um NNF zu erzeugen:
  - $\neg\neg C \equiv C$
  - $\neg(C \sqcup D) \equiv \neg C \sqcap \neg D$
  - $\neg(C \sqcap D) \equiv \neg C \sqcup \neg D$
  - $\neg\forall r.C \equiv \exists r.\neg C$
  - $\neg\exists r.C \equiv \forall r.\neg C$

Normalisierung - Beispiel

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \neg\exists\, eat.(Animal \sqcup PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

i. $Vegetarian \sqsubseteq \neg\exists eat.(Animal \sqcup PartOfAnimal)$
ii. $Vegetarian \sqsubseteq \forall eat.\neg(Animal \sqcup PartOfAnimal)$
iii. $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$

(1) $MadCow \sqsubseteq Cow$
(2) $Cow \sqsubseteq Vegetarian$
(3) $Vegetarian \sqsubseteq \forall eat.(\neg Animal \sqcap \neg PartOfAnimal)$
(4) $Brain \sqsubseteq PartOfAnimal$
(5) $MadCow \sqsubseteq \exists eat.Brain$

Endlose Modelle

- Es bleibt ein Problem zu lösen, dass unter anderem auftritt, wenn man das Tableauverfahren auf folgendes Beispiel anwendet

(1) $Human \sqsubseteq \exists father.Man$
(2) $Human \sqsubseteq \exists mother.Woman$
(3) $Man \sqcup Women \sqsubseteq Human$
(4) $Man \sqsubseteq \neg Women$

(1) $Women(a)$

Endlose Modelle

(1) $Human \sqsubseteq \exists father.Man$
(2) $Human \sqsubseteq \exists mother.Woman$
(3) $Man \sqcup Women \sqsubseteq Human$
(4) $Man \sqsubseteq \neg Women$

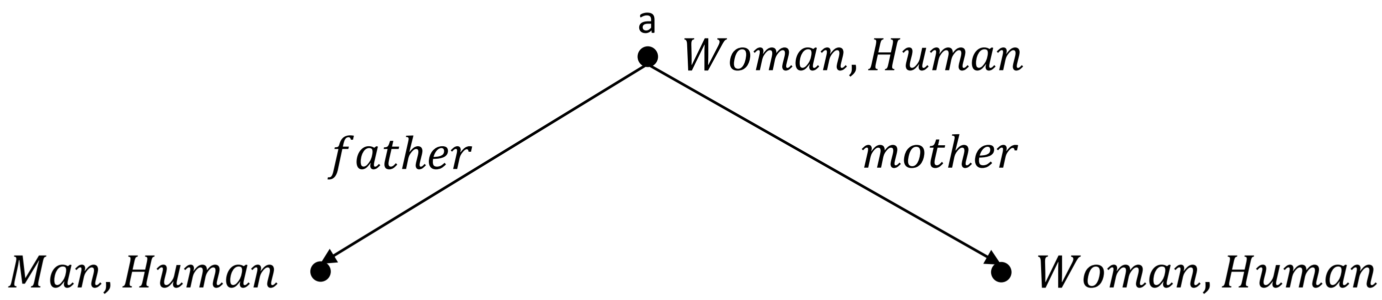
(1) $Women(a)$

a
● $Women, Human$

Endlose Modelle

(1) $Human \sqsubseteq \exists father.Man$
(2) $Human \sqsubseteq \exists mother.Woman$
(3) $Man \sqcup Women \sqsubseteq Human$
(4) $Man \sqsubseteq \neg Women$

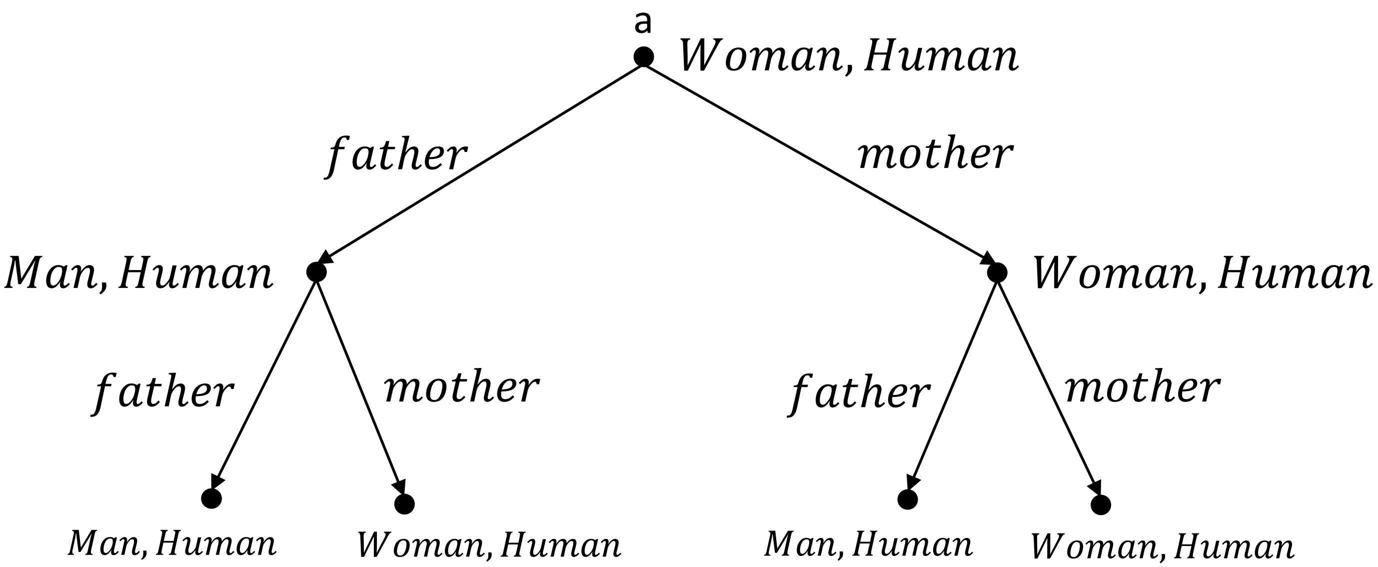
(1) $Women(a)$



Endlose Modelle

(1) $Human \sqsubseteq \exists father.Man$
(2) $Human \sqsubseteq \exists mother.Woman$
(3) $Man \sqcup Women \sqsubseteq Human$
(4) $Man \sqsubseteq \neg Women$

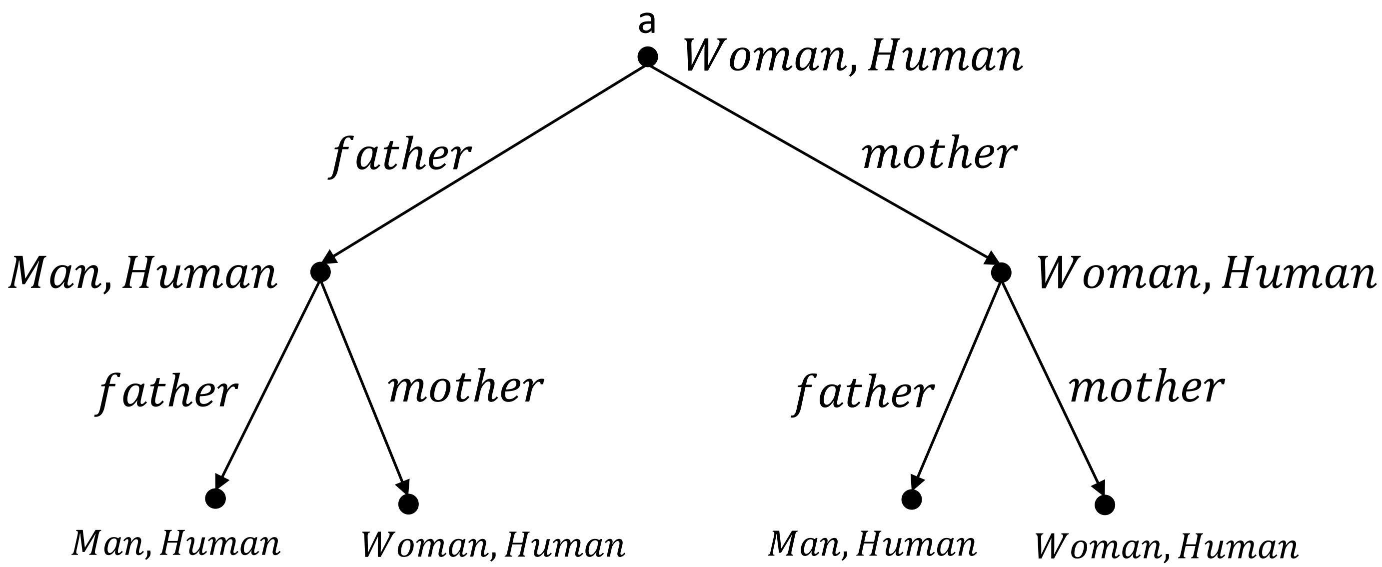
(1) $Women(a)$



Endlose Modelle

(1) $Human \sqsubseteq \exists father.Man$
(2) $Human \sqsubseteq \exists mother.Woman$
(3) $Man \sqcup Women \sqsubseteq Human$
(4) $Man \sqsubseteq \neg Women$

(1) $Women(a)$



Wann können wir aufhören?

Blocking

- Wenn ein Knoten auftaucht, der identisch „gelabelt“ ist, wie ein bereits existierender Knoten, dann muss dieser nicht weiter expandiert werden

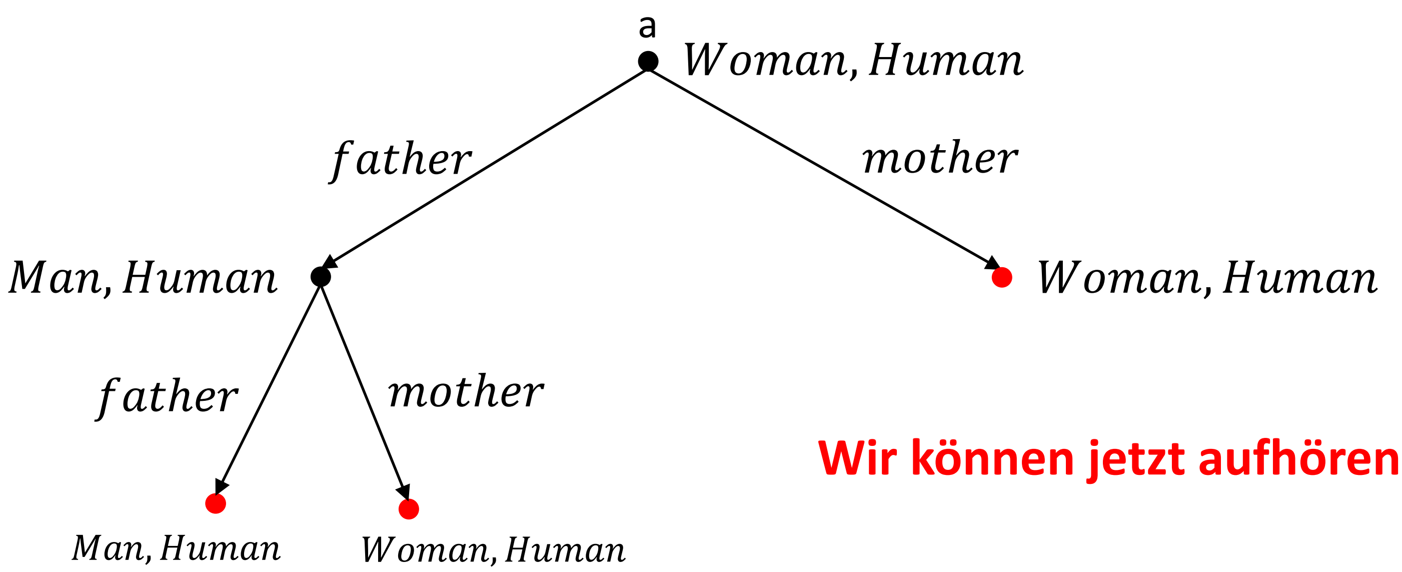
- Diese einfache Blocking Strategie gilt nur für das einfache hier vorgestellte Verfahren
  - Erweiterungen auf ausdruckstärkere Beschreibunsglogikdialekte benötigen deutlich kompliziertere Blocking Strategien

- In Bezug auf unser Beispiel werden im folgenden Knoten mit dieser Eigenschaft rot gefärbt
  - Siehe nächste Folie

Endlose Modelle

(1) $Human \sqsubseteq \exists father.Man$
(2) $Human \sqsubseteq \exists mother.Woman$
(3) $Man \sqcup Women \sqsubseteq Human$
(4) $Man \sqsubseteq \neg Women$

(1) $Women(a)$



Wir können jetzt aufhören!

Zusammenfassung / Ausblick

- Tableauverfahren analog zu ausagelogischem Verfahren
  - Versucht Modell (als Graph notiert) zu konstruieren
  - Andere Schreibweise, selbes Prinzip

- Besonderheit: Modelle können unendlich groß werden, wenn man keine „Blocking“ Strategien anwendet

Weiter geht es mit:

- Was sind typische Arten von Axiomen und zu welchem Zweck werden diese eingesetzt
  - Konzepthierarchie definieren
  - Domain und Range von Rollen spezifizieren
  - ...