Grundlagen

Die Beschreibungslogik $\mathcal{ALC}$

ALC

Von nun an sei

- $\mathbf{N_C}$ eine unendliche Menge von Konzeptnamen
  Diese entsprechen Klassen / unären Prädikaten
  z.B. Person, Kurs, Universität, Tafel, Student, etc.
- $\mathbf{N_R}$ eine unendliche Menge von Rollennamen
  Diese entsprechen Relationen / binären Prädikaten
  z.B. hört, lehrt, istTeilVon, etc.

Wir nehmen $\mathbf{N_C}$ und $\mathbf{N_R}$ als disjunkt und abzählbar unendlich an

ALC

Konzepte sind zusammengesetzt aus Konzeptnamen, Rollennamen und Konzeptkonstruktoren

Es gibt viele verschiedene Konstruktoren (manchmal auch für Rollen)

Versch. Konstruktormengen ergeben versch. Beschreibungslogiken

Hier zunächst $\mathcal{ALC}$:

- die einfachste BL mit allen Boolschen Konstruktoren
- eingeführt 1991 von Schmidt-Schauß und Smolka
- steht für *Attributive (Concept) Language with Complement*

ALC Syntax

Definition 2.1 ($\mathcal{ALC}$-Konzepte).
Die Menge der $\mathcal{ALC}$-*Konzepte* ist induktiv definiert:

- Jeder Konzeptname ist $\mathcal{ALC}$-Konzept
- Wenn C, D $\mathcal{ALC}$-Konzepte, so auch
  - $\neg C$ (Negation)
  - $C \sqcap D$ (Konjunktion)
  - $C \sqcup D$ (Disjunktion)
- Wenn C $\mathcal{ALC}$-Konzept und r Rollenname, so sind
  - $\exists r.C$ (Existenzrestriktion)
  - $\forall r.C$ (Werterestriktion)

  $\mathcal{ALC}$-Konzepte.

T2.1

ALC Syntax

Verwendete Symbole:

- A, B für Konzeptnamen
- C, D für zusammengesetzte Konzeptterme
- r, s für Rollennamen

Abkürzungen:

- $A \rightarrow B$ für $\neg A \sqcup B$
- $A \leftrightarrow B$ für $(A \rightarrow B) \sqcap (B \rightarrow A)$
- $\top$ für $A \sqcup \neg A$ (Top Konzept)
- $\bot$ für $\neg\top$ (Bottom Konzept)

ALC Semantik

Definition 2.2 ($\mathcal{ALC}$ Semantik).
Eine *Interpretation* $\mathcal{I}$ ist Paar $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ mit

- $\Delta^{\mathcal{I}}$ nicht-leere Menge (*Domäne*)
- $\cdot^{\mathcal{I}}$ *Interpretationsfunktion* bildet ab:
  - jeden Konzeptnamen $A \in \mathsf{N_C}$ auf Menge $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
  - jeden Rollennamen $r \in \mathsf{N_R}$ auf Relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Abbildung $\cdot^{\mathcal{I}}$ wird induktiv auf Konzeptterme erweitert:

- $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$,
- $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$, $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$,
- $(\exists r.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \text{es gibt } e \in \Delta^{\mathcal{I}} \text{ mit } (d,e) \in r^{\mathcal{I}} \text{ und } e \in C^{\mathcal{I}}\}$,
- $(\forall r.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \text{für alle } e \in \Delta^{\mathcal{I}}, (d,e) \in r^{\mathcal{I}} \text{ impliziert } e \in C^{\mathcal{I}}\}$

T2.2

ALC Semantik

Verwendete Symbole:

- $\mathcal{I}, \mathcal{J}$ für Interpretationen
- d, e für Elemente der Domäne

Für Interpretationen verwenden wir übliche Terminologie für Graphen:

- e ist r-*Nachfolger* von d (in $\mathcal{I}$) wenn $(d, e) \in r^{\mathcal{I}}$;
- e ist r-*Vorgänger* von d (in $\mathcal{I}$) wenn $(e, d) \in r^{\mathcal{I}}$;
- wenn r unwichtig, lassen wir es weg;
- e ist *erreichbar* von d wenn es $d_0, \ldots, d_n$ gibt $(n \geq 0)$ mit $d_0 = d$, $d_n = e$ und d_{i+1} ist Nachfolger von d_i für alle $i < n$;
- $\mathcal{I}$ is *endlich* gdw. $\Delta^{\mathcal{I}}$ endlich ist;
- $\mathcal{I}$ ist eine *Bauminterpretation* gdw. $(\Delta^{\mathcal{I}}, \bigcup_{r \in \mathsf{N_R}} r^{\mathcal{I}})$ ein (endlicher oder unendlicher) Baum ist.

Extension/ Modell

Wir nennen

- $C^{\mathcal{I}}$ die *Extension* des Konzeptes C;
- jedes $d \in C^{\mathcal{I}}$ eine *Instanz* des Konzeptes C;
- $r^{\mathcal{I}}$ die *Extension* der Rolle r.

Interpretation $\mathcal{I}$ ist *Modell* von Konzept C gdw. $C^{\mathcal{I}} \neq \emptyset$.

Grundlagen

TBoxen und Ontologien

TBoxen

Wir führen zwei Arten von TBoxen ein:

- Azyklische TBoxen.
  Ziel ist Definition von Konzepten, für viele Anwendungen ausreichend (z.B. SNOMED, NCI Thesaurus)
- Generelle TBoxen
  Ausdrucksstärker, erlauben allgemeine Constraints, führen oft zu höherer Komplexität beim Schlussfolgern

Wir setzen TBox = Ontologie.

Folgende Definitionen sind unabhängig von $\mathcal{ALC}$.

Azyklische TBox - Syntax

Definition 2.3 (Azyklische TBox)

- *Konzeptdefinition* ist Ausdruck $A \equiv C$
- *Primitive Konzeptinklusion* ist Ausdruck $A \sqsubseteq C$.

mit A Konzeptname und C Konzept.

Azyklische TBox ist endliche Menge von Konzeptdefinitionen und primitiven Konzeptinklusionen so dass

- linke Seiten eindeutig
- keine Zyklen vorkommen, also keine Ausdrücke

$$A_0 \equiv C_0 \ / \ A_0 \sqsubseteq C_0 \quad , \quad \ldots \quad , \quad A_{n-1} \equiv C_0 \ / \ A_{n-1} \sqsubseteq C_{n-1},$$

so dass A_i in C_{i+1} vorkommt, für alle $i < n$ (modulo n)

T2.3

Azyklische TBox - Syntax

Intuition:

- Definition $A \equiv C$ gibt notwendige und hinreichende Bedingungen dafür, eine Instanz des Konzeptes A zu sein.
- Inklusion $A \sqsubseteq C$ gibt nur notwendige Bedingungen.

Für TBox $\mathcal{T}$:

- Konzeptname A is *definiert* in $\mathcal{T}$ gdw. $\mathcal{T}$ enthält $A \equiv C$
- sonst ist A *primitiv* in $\mathcal{T}$.

T2.3 cont

Azyklische TBox - Semantik

Definition 2.4 (TBox Semantik) Interpretation $\mathcal{I}$ *erfüllt*

- $A \equiv C$ gdw. $A^{\mathcal{I}} = C^{\mathcal{I}}$;
- $A \sqsubseteq C$ gdw. $A^{\mathcal{I}} \subseteq C^{\mathcal{I}}$.

$\mathcal{I}$ ist *Modell* von azyklischer TBox $\mathcal{T}$ gdw. $\mathcal{I}$ alle Definitionen und Inklusionen in $\mathcal{T}$ erfüllt.

Wir schreiben $\mathcal{I} \models A \equiv C, \quad \mathcal{I} \models A \sqsubseteq C, \quad \mathcal{I} \models \mathcal{T}$.

T2.4

Generelle TBox - Syntax und Semantik

Definition 2.5 (Generelle TBox)
Konzeptinklusion ist Ausdruck $C \sqsubseteq D$, mit C, D Konzepten.
Generelle TBox ist endliche Menge von Konzeptinklusionen.

Interpretation $\mathcal{I}$

- *erfüllt* Konzeptinklusion $C \sqsubseteq D$ gdw. $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$;
- ist *Modell* von TBox $\mathcal{T}$ gdw. $\mathcal{I}$ alle Konzeptinklusionen in $\mathcal{T}$ erfüllt.

$A \equiv C$ ist äquivalent zu $A \sqsubseteq C$, $C \sqsubseteq A$,

also ist ist jede azyklische TBox auch generelle TBox.

Unterscheidung definierte/primitive Konzeptnamen nicht sinnvoll.

T2.5

Grundlagen

Schlussfolgerungsprobleme

Erfüllbarkeit, Subsumtion

Definition 2.6 (Erfüllbar, subsumiert, äquivalent)
Seien C, D $\mathcal{ALC}$-Konzepte und $\mathcal{T}$ TBox. Dann

- ist C *erfüllbar bzgl.* $\mathcal{T}$ gdw. C und $\mathcal{T}$ gemeinsames Modell haben;
- wird C *von* D *subsumiert bzgl.* $\mathcal{T}$ ($\mathcal{T} \models C \sqsubseteq D$) gdw $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ in allen Modellen $\mathcal{I}$ von $\mathcal{T}$.
- sind C und D *äquivalent bzgl.* $\mathcal{T}$ ($\mathcal{T} \models C \equiv D$) gdw $C^{\mathcal{I}} = D^{\mathcal{I}}$ in allen Modellen $\mathcal{I}$ von $\mathcal{T}$.

Diese Begriffe existieren auch ohne Bezug auf TBox (i.e., TBox leer)

T2.6

Schlussfolgerungsprobleme

Erfüllbarkeitsproblem:

gegeben C und $\mathcal{T}$, entscheide ob C erfüllbar bzgl. $\mathcal{T}$;

Subsumtionsproblem:

gegeben C, D und $\mathcal{T}$, entscheide ob $\mathcal{T} \models C \sqsubseteq D$;

Äquivalenzproblem:

gegeben C, D und $\mathcal{T}$, entscheide ob $\mathcal{T} \models C \equiv D$;

Alle drei Probleme auch ohne (mit leerer) TBox.

Untersch. TBox-Formalismen geben untersch. Entscheidungsprobleme.

Motivation

Anwendung:

- Modellierungsfehler finden:
  unerfüllbare Konzepte im allg. unerwünscht;
- die Struktur der TBox explizit machen, z.B. für Browsing:
  Subsumption (= Is-A Relation) haupt-Stukturierungsmittel für TBoxen
- Redundanzen finden:
  zwei äquivalente Konzepte sind u.U. unerwünscht.

Subsumtion als Ordnungsrelation

Lemma 2.7
Seit $\mathcal{T}$ TBox mit mind. einem Modell. Subsumtionsrelation $\sqsubseteq$ bzgl. $\mathcal{T}$ ist

- reflexiv ($\mathcal{T} \models C \sqsubseteq C$),
- transitiv ($\mathcal{T} \models C \sqsubseteq D$ und $\mathcal{T} \models D \sqsubseteq E$ impliziert $\mathcal{T} \models C \sqsubseteq E$),
- nicht antisymmetrisch (es gibt C, D mit $\mathcal{T} \models C \sqsubseteq D$ und $\mathcal{T} \models D \sqsubseteq C$),
- nicht symmetrisch (es gibt C, D mit $\mathcal{T} \models C \sqsubseteq D$ und $\mathcal{T} \not\models D \sqsubseteq C$).

Bis auf fehlende Antisymmetrie ist $\sqsubseteq$ also partielle Ordnung.

Man kann $\sqsubseteq$ als Hasse-Diagram darstellen, dessen Knoten mit Mengen von Konzepten beschriftet sind.

Normalerweise Einschränkung auf die in $\mathcal{T}$ verwendeten Konzeptnamen

T2.7

Klassifikation

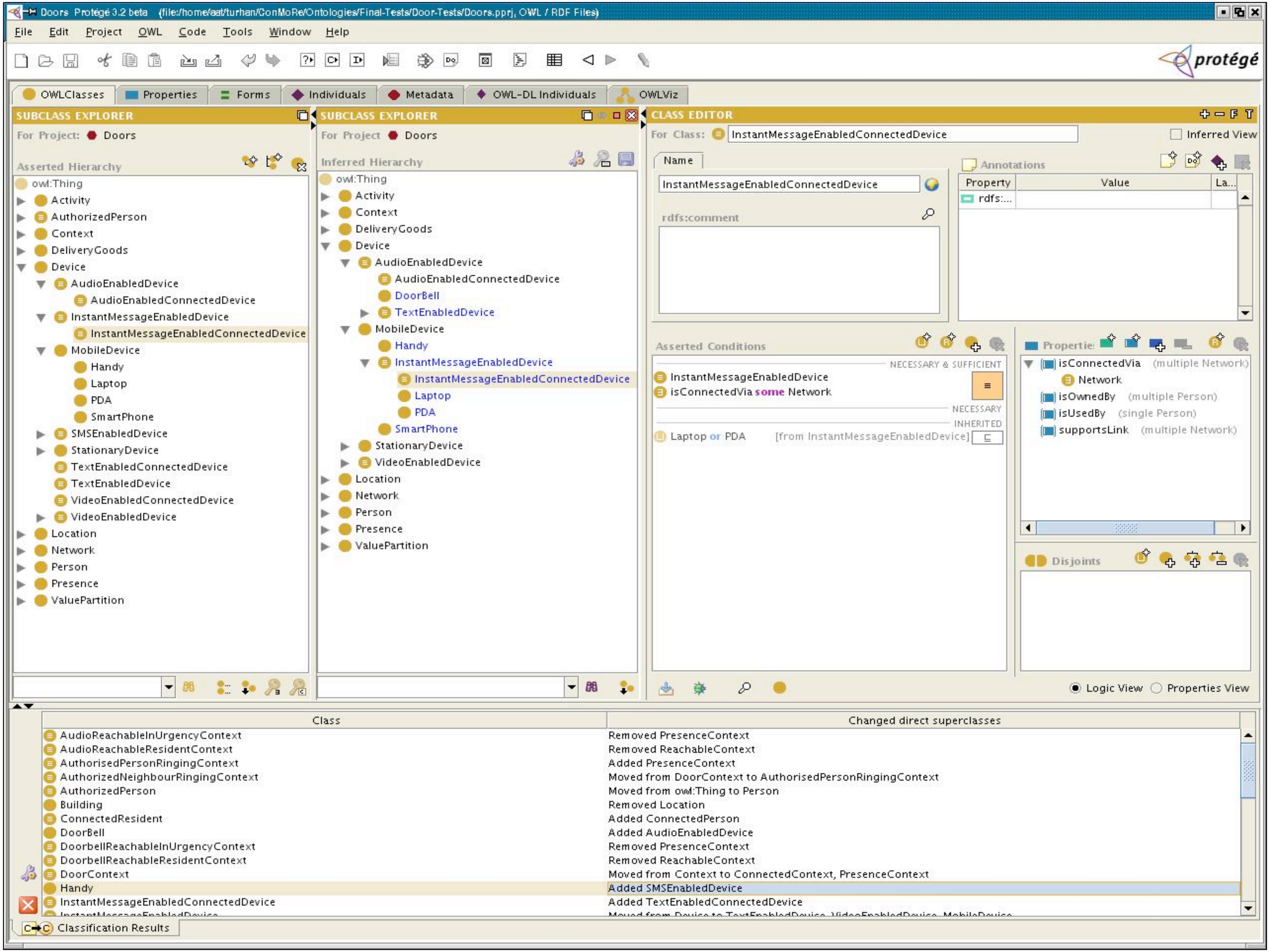
Ein weiteres Schlussfolgerungsproblem:

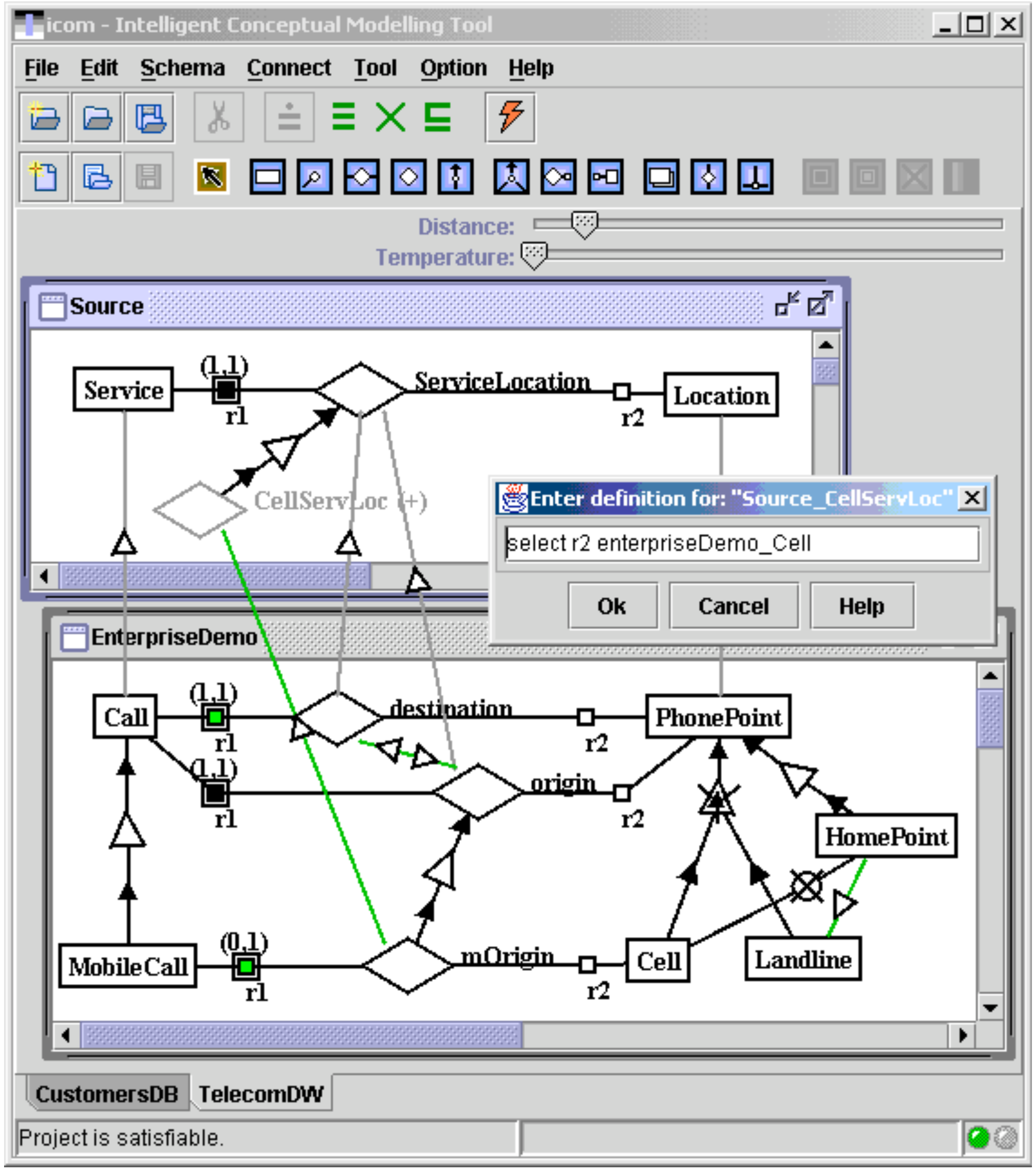
- *Klassifikation*: gegeben $\mathcal{T}$, berechne das Hasse-Diagramm für $\sqsubseteq$ bzgl. $\mathcal{T}$, eingeschränkt auf Konzeptnamen in $\mathcal{T}$.

Berechnungsproblem, kein Entscheidungsproblem.

In der Praxis:

- berechenbar durch n^2 Subsumtionsberechnungen;
- zahlreiche Optimierungen verfügbar.


Doors Protégé 3.2 beta (file:/home/aat/turhan/ConMoRe/Ontologies/Final-Tests/Door-Tests/Doors.pprj, OWL / RDF Files)
File Edit Project OWL Code Tools Window Help
protégé
OWLClasses Properties Forms Individuals Metadata OWL-DL Individuals OWLViz
SUBCLASS EXPLORER
For Project: Doors
Asserted Hierarchy
owl:Thing
Activity
AuthorizedPerson
Context
DeliveryGoods
Device
AudioEnabledDevice
AudioEnabledConnectedDevice
InstantMessageEnabledDevice
InstantMessageEnabledConnectedDevice
MobileDevice
Handy
Laptop
PDA
SmartPhone
SMSEnabledDevice
StationaryDevice
TextEnabledConnectedDevice
TextEnabledDevice
VideoEnabledConnectedDevice
VideoEnabledDevice
Location
Network
Person
Presence
ValuePartition
SUBCLASS EXPLORER
For Project: Doors
Inferred Hierarchy
owl:Thing
Activity
Context
DeliveryGoods
Device
AudioEnabledDevice
AudioEnabledConnectedDevice
DoorBell
TextEnabledDevice
MobileDevice
Handy
InstantMessageEnabledDevice
InstantMessageEnabledConnectedDevice
Laptop
PDA
SmartPhone
StationaryDevice
VideoEnabledDevice
Location
Network
Person
Presence
ValuePartition
CLASS EDITOR
For Class: InstantMessageEnabledConnectedDevice
Inferred View
Name
InstantMessageEnabledConnectedDevice
rdfs:comment
Annotations
Property Value La...
rdfs:...
Asserted Conditions
NECESSARY & SUFFICIENT
InstantMessageEnabledDevice
isConnectedVia some Network
NECESSARY
INHERITED
Laptop or PDA [from InstantMessageEnabledDevice]
Propertie
isConnectedVia (multiple Network)
Network
isOwnedBy (multiple Person)
isUsedBy (single Person)
supportsLink (multiple Network)
Disjoints
Logic View Properties View
Class Changed direct superclasses
AudioReachableInUrgencyContext Removed PresenceContext
AudioReachableResidentContext Removed ReachableContext
AuthorisedPersonRingingContext Added PresenceContext
AuthorizedNeighbourRingingContext Moved from DoorContext to AuthorisedPersonRingingContext
AuthorizedPerson Moved from owl:Thing to Person
Building Removed Location
ConnectedResident Added ConnectedPerson
DoorBell Added AudioEnabledDevice
DoorbellReachableInUrgencyContext Removed PresenceContext
DoorbellReachableResidentContext Removed ReachableContext
DoorContext Moved from Context to ConnectedContext, PresenceContext
Handy Added SMSEnabledDevice
InstantMessageEnabledConnectedDevice Added TextEnabledConnectedDevice
Classification Results


icom - Intelligent Conceptual Modelling Tool
File Edit Schema Connect Tool Option Help
Distance:
Temperature:
Source
Service
(1,1)
r1
ServiceLocation
r2
Location
CellServLoc (+)
Enter definition for: "Source_CellServLoc"
select r2 enterpriseDemo_Cell
Ok
Cancel
Help
EnterpriseDemo
Call
(1,1)
r1
destination
r2
PhonePoint
(1,1)
r1
origin
r2
HomePoint
MobileCall
(0,1)
r1
mOrigin
r2
Cell
Landline
CustomersDB
TelecomDW
Project is satisfiable.

Reduktionen

Erfüllbarkeit, Subsumtion, Äquivalenz wechselseitig polynomiall reduzierbar:

Lemma 2.8.

- C erfüllbar bzgl. $\mathcal{T}$ gdw. $\mathcal{T} \not\models C \sqsubseteq \bot$ bzgl. $\mathcal{T}$
- $\mathcal{T} \models C \sqsubseteq D$ gdw. $\mathcal{T} \models C \equiv C \sqcap D$
- $\mathcal{T} \models C \equiv D$ gdw. $\neg(C \leftrightarrow D)$ unerfüllbar bzgl. $\mathcal{T}$.

Natürlich auch korrekt wenn $\mathcal{T} = \emptyset$, erhält Azyklizität von $\mathcal{T}$.

T2.8

Reduktionen

Wir wollen zeigen:

Theorem 2.9
Erfüllbarkeit bzgl. azyklischer TBoxen kann reduziert werden auf Erfüllbarkeit ohne TBoxen.

Mit Lemma 2.8 folgt dasselbe für Subsumtion und Äquivalenz.

Wir gehen in drei Schritten vor.

Beweis von Theorem 3.9

Schritt 1:

Wir können o.B.d.A annehmen, dass es in azyklischen TBoxen nur Konzept-definitionen, aber keine Implikationen gibt.

Definitorische TBox

Definition 2.10 (Definitorische TBox)
Eine azyklische TBox heisst *definitorisch* wenn sie nur Konzept-definitionen aber keine Inklusionen enthält.

Sei $\mathcal{T}$ azyklische TBox.

Definitorische TBox $\mathcal{T}'$: ersetze jede Inklusion
$A \sqsubseteq C \in \mathcal{T}$ durch

$$A \equiv X_A \sqcap C, \text{ mit } X_A \text{ neuer Konzeptname}$$

Lemma 2.11
Für alle $\mathcal{ALC}$-Konzepte C, die nicht die neuen Konzeptnamen verwenden, gilt:

$$C \text{ erfüllbar bzgl. } \mathcal{T} \text{ gdw. } C \text{ erfüllbar bzgl. } \mathcal{T}'$$

T2.9

Beweis von Theorem 3.9

Schritt 2:

Wir können o.B.d.A annehmen, dass auf den rechten Seiten von Konzeptdefinitionen keine definierten Konzeptnamen vorkommen.

Azyklische TBox - Expandieren

Definition 2.12
Sei $\mathcal{T}$ definitorische TBox. Definiere Folge von definitorischen TBoxen $\mathcal{T}_0, \mathcal{T}_1, \ldots$ wie folgt:

- $\mathcal{T}_0 = \mathcal{T}$;
- $\mathcal{T}_{i+1}$ entsteht aus $\mathcal{T}_i$ wie folgt:

  für alle $A \equiv C \in \mathcal{T}_i$: jeder Konzeptname B in C mit $B \equiv D \in \mathcal{T}_i$ wird ersetzt durch D.

Da $\mathcal{T}$ azyklisch gibt es $i \geq 0$ mit $\mathcal{T}_i = \mathcal{T}_{i+1}$.

$\mathcal{T}_i$ ist die *Expansion* von $\mathcal{T}$.

T2.10

Azyklische TBox - Expandieren

Zwei TBoxen sind *äquivalent* gdw. sie dieselben Modelle haben.

Lemma 2.13
Jede definitorische TBox ist äquivalent zu ihrer Expansion. T2.11

Bemerkungen:

Intuitiv zeigt Proposition 2.13, dass Konzeptdefinitionen nicht mehr sind als Makros

Azyklische TBox - Expandieren

Expansion kann zu einem exponentiellen Aufblasen der TBox führen:

Expandieren von

$$\begin{aligned} A_0 &\equiv \exists r.A_1 \sqcap \exists s.A_1 \\ A_1 &\equiv \exists r.A_2 \sqcap \exists s.A_2 \\ &\cdots \\ A_{n-1} &\equiv \exists r.A_n \sqcap \exists s.A_n \end{aligned}$$

gibt TBox der Grösse $> 2^n$.

T2.12

Es gibt keine äquivalente kleinere TBox ohne definierte Konzepte auf den rechten Seiten.

Beweis von Theorem 3.9

Schritt 3:

Expandieren der definierten Konzeptnamen im Eingabekonzept.

Zunächst etwas Vorbereitung.

Primitive Interpretation

Definition 2.14 (Primitive Interpretation)
Primitive Interpretation für azyklische TBox $\mathcal{T}$ ist Interpretation, die nur primitive Konzeptnamen und Rollennamen interpretiert, aber keine definierten Konzeptnamen aus $\mathcal{T}$.

Lemma 2.15
Sei $\mathcal{T}$ definitorische TBox. Jede primitive Interpretation kann in eindeutiger Weise zu Modell von $\mathcal{T}$ erweitert werden.

T2.13

Beweis von Theorem 2.9

Theorem 2.9
Erfüllbarkeit bzgl. azyklischer TBoxen kann reduziert werden auf Erfüllbarkeit ohne TBoxen.

T2.15

Grundlagen

Erweiterungen von $\mathcal{ALC}$

Erweiterungen von ALC

Wir betrachten exemplarisch zwei Erweiterungen von $\mathcal{ALC}$:

- $\mathcal{ALCI}$: $\mathcal{ALC}$ mit inversen Rollen (Rollenkonstruktor)
- $\mathcal{ALCQ}$: $\mathcal{ALC}$ mit Zahlenrestriktionen (Konzeptkonstruktor)

Beide Erweiterungen sind in OWL realisiert.

Namenschema:

- ein Buchstabe pro Erweiterung
- kann kombiniert werden, z.B. $\mathcal{ALCQI}$

Inverse Rollen

Häufig möchte man über Rollen “in beiden Richtungen” reden:

- SNOMED: hatTeil / istTeilVon (z.B. in der Anatomie)
- Universitätsbeispiel: hört / wirdGehörtVon ,
  gibt / wirdGegebenVon

Verwendet man diese Rollen einfach als Namen in $\mathcal{ALC}$,
gibt es unintuitive Konsequenzen.

T2.17

Inverse Rollen

Definition 2.16. (Inverse Rollen)
Für jedes $\boldsymbol{r} \in \mathsf{N_R}$ ist $\boldsymbol{r}^-$ die *inverse Rolle* zu $\boldsymbol{r}$. Wir definieren

$$(\boldsymbol{r}^-)^{\mathcal{I}} = \{(\boldsymbol{e}, \boldsymbol{d}) \mid (\boldsymbol{d}, \boldsymbol{e}) \in \boldsymbol{r}^{\mathcal{I}}\}.$$

Eine *Rolle* ist entweder ein Rollenname oder ein inverse Rolle.

$\mathcal{ALCI}$: $\mathcal{ALC}$ erweitert um die Möglichkeit, inverse Rollen in Existenz- und Werterestriktionen zu benutzen.

T2.17 cont.

Schachtelung von Inversen i. allg. nicht sinnvoll da $((\boldsymbol{r}^-)^-)^{\mathcal{I}} = \boldsymbol{r}^{\mathcal{I}}$.
Daher setzen wir $(\boldsymbol{r}^-)^- = \boldsymbol{r}$.

Zahlenrestriktionen

Häufig möchte man Rollennachfolger “zählen” können:

- SNOMED: Eine Hand hat fünf Teile, die ein Finger sind.
- Universitätsbeispiel: wie viele Vorlesungen hört ein Student?

In $\mathcal{ALC}$ ist zählen nicht ohne weiteres möglich

T2.18

Zahlenrestriktionen

Definition 2.17. (Zahlenrestriktion)
Für jede natürliche Zahl $\boldsymbol{n}$, jeden Rollennamen $\boldsymbol{r}$ und jedes Konzept $\boldsymbol{C}$ definiere Konzepte wie folgt:

- $(\leqslant \boldsymbol{n}\ \boldsymbol{r}\ \boldsymbol{C})$ (Höchstens-Restriktion);
- $(\geqslant \boldsymbol{n}\ \boldsymbol{r}\ \boldsymbol{C})$ (Mindestens-Restriktion);

Die Semantik ist

- $(\leqslant \boldsymbol{n}\ \boldsymbol{r}\ \boldsymbol{C})^{\mathcal{I}} = \{\boldsymbol{d} \in \boldsymbol{\Delta}^{\mathcal{I}} \mid \#\{\boldsymbol{e} \mid (\boldsymbol{d}, \boldsymbol{e}) \in \boldsymbol{r}^{\mathcal{I}} \wedge \boldsymbol{e} \in \boldsymbol{C}^{\mathcal{I}}\} \leq \boldsymbol{n}\}$
- $(\geqslant \boldsymbol{n}\ \boldsymbol{r}\ \boldsymbol{C})^{\mathcal{I}} = \{\boldsymbol{d} \in \boldsymbol{\Delta}^{\mathcal{I}} \mid \#\{\boldsymbol{e} \mid (\boldsymbol{d}, \boldsymbol{e}) \in \boldsymbol{r}^{\mathcal{I}} \wedge \boldsymbol{e} \in \boldsymbol{C}^{\mathcal{I}}\} \geq \boldsymbol{n}\}$

T2.18 cont.

Beachte:

- $\exists \boldsymbol{r}.\boldsymbol{C}$ ist äquivalent zu $(\geqslant \mathbf{1}\ \boldsymbol{r}\ \boldsymbol{C})$
- $\forall \boldsymbol{r}.\boldsymbol{C}$ ist äquivalent zu $(\leqslant \mathbf{0}\ \boldsymbol{r}\ \neg\boldsymbol{C})$

Zahlenrestriktionen

$\mathcal{ALCQ}$: $\mathcal{ALC}$ erweitert mit Zahlenrestriktionen.

$\mathcal{ALCQI}$: $\mathcal{ALC}$ erweitert mit Zahlenrestriktionen und inversen Rollen.

inverse Rollen auch in Zahlenrestriktionen, z.B.

$$\text{Vorlesung} \sqsubseteq (\geqslant \mathbf{3}\ \text{hört}^{-}\ \text{Student})$$

Weitere Erweiterungen

Es gibt noch viel mehr Erweiterungen:

- Spezielle Rolleninterpretationen (transitiv, symmetrisch, reflexiv, etc)
- Konzepte, die nur eine einzige Instanz haben koennen (Nominale)
- Inklusionen zwischen Rollen oder Rollenketten
- Numerische Werte
- Fixpunktoperatoren
- temporale Operatoren
- ...

Grundlagen

Zusammenhang zur Logik erster Stufe

Zusammenhang zu FO

BL Interpretation ist FO Struktur

- mit unären und binären Prädikaten (Konzeptnamen und Rollennamen);
- ohne Prädikate höherer Arität;
- ohne Funktionssymbole.

Wir übersetzen

- $\mathcal{ALC}$-Konzepte C in FO-Formeln $C^{\#}$ mit einer freien Variable
- Generelle TBoxen $\mathcal{T}$ in FO-Sätze $\mathcal{T}^{\#}$.

Zusammenhang zu FO

Für FO-Formel φ mit einer freien Variable und Interpretation $\mathcal{I}$ definieren wir die *Extension*

$$\varphi^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} \mid \mathcal{I} \models \varphi[d]\}$$

Die Übersetzung soll erfüllen, für alle C, $\mathcal{T}$, $\mathcal{I}$:

- $C^{\mathcal{I}} = (C^{\#})^{\mathcal{I}}$;
- $\mathcal{I} \models \mathcal{T}$ gdw. $\mathcal{I} \models \mathcal{T}^{\#}$.

Wir übersetzen so, dass wir mit insgesamt zwei Variablen auskommen.

Zusammenhang zu FO

Definiere Abbildungen

π_x, π_y von $\mathcal{ALC}$-Konzepten auf FO-Formeln mit freier Variable x, y:

$$\text{Für } z \in \{x, y\}, \text{ sei } \widehat{z} := \begin{cases} x & \text{if } z = y \\ y & \text{if } z = x \end{cases}$$

$$\begin{aligned}
\pi_z(A) &:= A(z) \\
\pi_z(\neg C) &:= \neg\pi_z(C) \\
\pi_z(C \sqcap D) &:= \pi_z(C) \wedge \pi_z(D) \\
\pi_z(C \sqcup D) &:= \pi_z(C) \vee \pi_z(D) \\
\pi_z(\exists r.C) &:= \exists\widehat{z}.(r(z, \widehat{z}) \wedge \pi_{\widehat{z}}(C)) \\
\pi_z(\forall r.C) &:= \forall\widehat{z}.(r(z, \widehat{z}) \rightarrow \pi_{\widehat{z}}(C))
\end{aligned}$$

T2.19

Zusammenhang zu FO

Setze $C^{\#} := \pi_x(C)$

$$\mathcal{T}^{\#} := \bigwedge_{C \sqsubseteq D \in \mathcal{T}} \forall x.(C^{\#}(x) \rightarrow D^{\#}(x))$$

Lemma 2.18.
Für alle Konzepte C, TBoxen $\mathcal{T}$ und Interpretationen $\mathcal{I}$:

1. $C^{\mathcal{I}} = (C^{\#})^{\mathcal{I}}$;
2. $\mathcal{I} \models \mathcal{T}$ gdw. $\mathcal{I} \models \mathcal{T}^{\#}$;

T2.20

Zusammenhang zu FO

Lemma 2.19.
Für alle Konzepte C und TBoxen $\mathcal{T}$:

1. C erfüllbar bzgl. $\mathcal{T}$ gdw. $\mathcal{T}^{\#} \wedge \exists x.C^{\#}(x)$ erfüllbar;
2. $\mathcal{T} \models C \sqsubseteq D$ gdw. $\mathcal{T}^{\#} \rightarrow \forall x.(C^{\#}(x) \rightarrow D^{\#}(x))$ gültig;
3. $\mathcal{T} \models C \equiv D$ gdw. $\mathcal{T}^{\#} \rightarrow \forall x.(C^{\#}(x) \leftrightarrow D^{\#}(x))$ gültig.

Dies sind Reduktionen von Schließen in $\mathcal{ALC}$ auf Schließen in FO, sogar:

- in das zwei-Variablen Fragment FO^2;
- in das Guarded Fragment GF, in dem quantifizierte Variablen von syntaktischen "Guards" relativiert werden müssen, z.B.

$$\forall y.(\underbrace{r(x,y)}_{\text{Guard}} \rightarrow A(y))$$

Zusammenhang zu FO

Erweiterungen können auch übersetzt werden, z.B. inverse Rollen:

$$\begin{aligned}
\pi_z(\exists r^-.C) &:= \exists \widehat{z}.(r(\widehat{z}, z) \wedge \pi_z(C)(\widehat{z})) \\
\pi_z(\forall r^-.C) &:= \forall \widehat{z}.(r(\widehat{z}, z) \rightarrow \pi_z(C)(\widehat{z}))
\end{aligned}$$

Immernoch guarded und nur zwei Variablen (gilt nicht für $\mathcal{ALCQ}$).

Zusammenhang zu Modallogik

Ein enger Zusammenhang besteht auch zur Modallogik.

$\mathcal{ALC}$ ist multi-modales K mit anderer Syntax:

- BL Interpretation ist Kripke Struktur;
- Konzeptname ist propositionale Variable;
- Rollenname ist Erreichbarkeitsrelation;
- $\neg$, $\sqcap$, $\sqcup$ entsprechen $\neg$, $\wedge$, $\vee$;
- $\exists \boldsymbol{r}.\boldsymbol{C}$ entspricht $\Diamond_r \boldsymbol{C}$;
- $\forall \boldsymbol{r}.\boldsymbol{C}$ entspricht $\Box_r \boldsymbol{C}$;
- $\mathcal{ALC}$-Konzept entspricht K-Formel.

z.B. $\boldsymbol{A} \sqcap \exists \boldsymbol{r}.(\boldsymbol{B} \sqcap \forall \boldsymbol{r}.(\boldsymbol{A} \sqcup \boldsymbol{B}))$

$$\approx \boldsymbol{a} \wedge \Diamond_r (\boldsymbol{b} \wedge \Box_r (\boldsymbol{a} \vee \boldsymbol{b}))$$

TBoxen haben nur ungefähre Entsprechungen.