

	TCDB (SQL)	Symbolic Reasoning	Formale Systeme Metalevel Beweise über FOL
Variablen x, y, ...	(-)	✓	
Anwendungssprache Präd. Symbole P(...), q(...)	!	!	!
Built-in predicates überbuilt in Datenbase <=, >, Datentypen	!	✓ über Datentypen > < ...	(-)
Literalkonstanten 3, 4, 3.14, "a" ...	!	✓	(-)
Konstanten (nullstellige Fkt. Symbole)	SQL- CO-DB ✓ RDF-BSV	✓	I: john \mapsto obj. FOL I(john) \in I(reason)
Spezialisierte Fkt. Symbole			(-)
• built-in Fkt +, -, *, /	built-in überbuilt-in DT	✓	
• Anwendungssprache Fkt. Beziehung	coding in Relationen \rightarrow (v)	✓	I: I(capital)(germany) = berlin \hookrightarrow capital(germany) \hookrightarrow Gleichheit = berlin

Okt 26-10:23

yl.FS

person(john)
 person(alice)
 hasChild(john, alice)
 !
 name(john, "John")
 age(john, 32)

Metalevel

FS: nullstellige
 Konstantensymbole
 $\hat{=}$ object identifier
 J: \mapsto Objekte in Domain

Symbolic Reasoning
 $\hat{=}$ Direkte, indirekte
 nur 100er
 $\hat{=}$ Objektorientierung!

Okt 26-10:31

Mehrstellige Fkt Symbole

- plus(3,4) = 7
- $I(\text{plus})(3,4) = 7$
- Capital/n : Capital (germany) = Berlin
- FS: $I(\text{capital}) : \{ \text{germany} \mapsto \text{Berlin}, \text{france} \mapsto \text{Paris}, \dots \}$

args.	capital	city
	germany	Berlin
	france	Berlin

\Rightarrow n-stellige Fkt-Symbole \rightarrow n+1-stellige Relation abbild

Konstantensymbole

Okt 26-10:38

Formeln

atomare F: ^{(rel.) DB} erledigt ^{syml. Reasoning} $p(\dots)$ ^{Metakernel} \sum, \leq Gmlt-12.5

query (select from) \rightarrow \parallel

Negation: $\neg F$ "closed world" Safety **DD** **closedw.** **Sicherheit** **open world** **downward-dependence**

Konjunktio: $F \wedge G$ a. f. g. p. i. v Speiden: v

Disjunktio: $F \vee G$ a. f. g. p. i. v Speiden: -

Implication $A \Rightarrow B \equiv \neg A \vee B$ **De Morgan**

SQL: \checkmark **at. pos. Prädikat** \checkmark **Konj.** \checkmark

$\text{Pos. Detlog} \checkmark$ **Res. Gfc** $(\dots \vee)$ \checkmark

$M \neq T \wedge G$
 $\Leftrightarrow M \neq F$
 $\text{und } M = G$ \checkmark

Okt 26-10:49

Disjunction:

select to how colors
 where Pop > 10.000.000 or
 area > 1.000.000 → OR, U

stadt	lehre	unfall	m	w
A, B.	DD	⋮	true	false
C, D.	DB	⋮	false	true
X, Z.	XP	⋮	null	null

in complete knowledge

alle w. personen U alle natürl. Personen = {A, B, C, D}
 alle Person, die m u w sind = ! = {A, B, C, D, X, Z}

Ans: Person ≡ M ∪ W
 Formel → knowledge Base

Okt 26-11:07

Fish Puzzle:

color (HAUS, color)
 smokes (HAUS, CIG)

1) color (X, yellow) ↔ smokes (X, Dunhill)
 color (Z, yellow)

¬ color (X, yellow) ∨ smokes (X, Dunhill)

A → B
 ¬ A ∨ B

color (Z, yellow)

smokes (X, Dunhill)

¬ A ∨ B A → B
 A A
 B B

2) color (X, yellow) ↔ smokes (X, Dunhill)
 color (Z, yellow) ∨ pet (Z, fish)

¬ color (X, yellow) ∨ smokes (X, Dunhill)

falls gelb: color (Z, yellow) ∨ pet (Z, fish)
 falls grün

smokes (X, Dunhill) ∨ pet (Z, fish)

Resolutionskizze

Okt 26-11:14

SQL views:

$$\text{symborder}(x,y) \leftarrow \text{border}(x,y) \cup \text{border}(y,x)$$

$$\begin{cases} \text{symborder}(x,y) \leftarrow \text{border}(x,y). \\ \text{symborder}(x,y) \leftarrow \text{border}(y,x). \end{cases}$$

Create view symborder as
 (select c1,c2) from border
 union (select c2,c1) from border

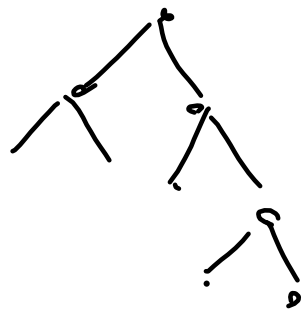


\Rightarrow SQL \supseteq pos. Relch
 Union of Relch

Okt 26-11:20

Transitive Closure
hasChild

john	alice
jack	john
⋮	⋮



\neg in SQL!

$\forall x,y$

$$\text{ancestor}(x,y) \leftarrow \text{hasChild}(x,y).$$

$\forall x,y$

$$\text{ancestor}(x,y) \leftarrow \exists z: \text{ancestor}(x,z) \wedge \text{child}(z,y)$$

rekursives Datentyp

Okt 26-11:34

	SQL	Symb. reasoning	FS Metalevel
Existenz:	Ansatz: ✓	(incomplete knowledge)	✓
speichern -		✓ über Schema-fkt. (DD-) (Sem Web)	✓

Okt 26-11:38

Existenz:

$\exists x: \text{river}(x) \wedge \text{locatedIn}(x, \text{andorra})$

$\text{riverInCountry}: \text{Land} \rightsquigarrow \text{irgendwelcher Fluss in dem Land}$
 "Zeugen" für die Existenz

$\text{river}(\text{f}_{\text{ric}}(\text{andorra})) \wedge \text{locatedIn}(\text{f}_{\text{ric}}(\text{andorra}), \text{andorra})$

$\forall x: \text{person}(x) \rightarrow \exists y: \text{hasSport}(x, y)$

$\text{f}_{\text{person}}(x) \rightsquigarrow x\text{'s sport}$

$\exists x: \text{hasSport}(\text{alice}, x)$

$\text{hasSport}(\text{alice}, \text{f}_{\text{person}}(\text{alice}))$

Term

(zblen-kalkül)

Okt 26-11:41