Lecture 3: Databases and Description Logics

2-AIN-144/2-IKV-131 Knowledge Reperesentation & Reasoning

Martin Baláž, Martin Homola

Department of Applied Informatics
Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava

28 Feb 2013

Entity-Relationship Schemas

ER Schemas

Definition (ER Schema)

An ER schema $\mathcal{S}$ consists of pairwise disjoint sets of entity symbols $\mathcal{E}_\mathcal{S}$, relationship symbols $\mathcal{R}_\mathcal{S}$, attribute symbols $\mathcal{A}_\mathcal{S}$, and domain symbols $\mathcal{D}_\mathcal{S}$, s.t.:

- each entity $E \in \mathcal{E}_\mathcal{S}$ is assoc. with a set of attributes from $\mathcal{A}_\mathcal{S}$
- each attribute $A \in \mathcal{A}_\mathcal{S}$ is assoc. with its base domain $D^A \in \mathcal{D}_\mathcal{S}$
- each relationship $R \in \mathcal{R}_\mathcal{S}$ is associated with arity $n > 0$, a numbered set of n entities $R{:}1, \ldots, R{:}n$ from $\mathcal{E}_\mathcal{S}$ and cardinality constraints $cmin_\mathcal{S}(R{:}i) \in \{0, 1, \ldots\}$ and $cmax_\mathcal{S}(R{:}i) \in \{1, 2, \ldots, \infty\}$ for $0 \leq i \leq n$
- Two IS-A relations between entities and relationships (both) denoted $\preceq_\mathcal{S}$ (i.e., $\preceq_\mathcal{S} \subseteq (\mathcal{E}_\mathcal{S} \times \mathcal{E}_\mathcal{S}) \cup (\mathcal{R}_\mathcal{S} \times \mathcal{R}_\mathcal{S})$)

ER Schemas (cont.)

Definition (Database state)

Given an ER schema $\mathcal{S}$, a database state $\mathcal{B} = (\Delta^{\mathcal{B}}, \cdot^{\mathcal{B}})$ consists of non-empty finite set $\Delta^{\mathcal{B}}$ disjoint from all domains in $\mathcal{D}_{\mathcal{S}}$, and a function $\cdot^{\mathcal{B}}$ that maps

- every entity $E \in \mathcal{E}_{\mathcal{S}}$ to $E^{\mathcal{B}} \subseteq \Delta^{\mathcal{B}}$
- every attribute $A \in \mathcal{A}_{\mathcal{S}}$ to $A^{\mathcal{B}} \subseteq \Delta^{\mathcal{B}} \times \bigcup_B D^B$
- every relationship $R \in \mathcal{R}_{\mathcal{S}}$ of arity n to $R^{\mathcal{B}} \subseteq (\Delta^{\mathcal{B}})^n$

ER Schemas (cont.)

Definition (Legal database state)

A database state $\mathcal{B}$ is legal w.r.t. an ER schema $\mathcal{S}$ iff

- $E_1^{\mathcal{B}} \subseteq E_2^{\mathcal{B}}$ for every two entities $E_1 \preceq_{\mathcal{S}} E_2$
- $R_1^{\mathcal{B}} \subseteq R_2^{\mathcal{B}}$ for every two relationships $R_1 \preceq_{\mathcal{S}} R_2$
- for every entity E with attribute A, and for every $e \in E^{\mathcal{B}}$ there is exactly one element in $\langle e, d \rangle \in A^{\mathcal{B}}$, and in addition $d \in D^A$
- for each relationship R with arity n we have $R^{\mathcal{B}} \in R{:}1^{\mathcal{B}} \times \cdots \times R{:}n^{\mathcal{B}}$
- for each relationship R of arity n, for every $1 \leq i \leq n$ and for every $(e_1, \ldots, e_{i-1}, e_{i+1}, \ldots, e_n) \in (\Delta^{\mathcal{B}})^{n-1}$ we have $cmin_{\mathcal{S}}(R{:}i) \leq \#\{x \mid (e_1, \ldots, e_{i-1}, x, e_{i+1}, \ldots, e_n) \in R^{\mathcal{B}}\} \leq cmax_{\mathcal{S}}(R{:}i)$

ER Schemas (cont.)

Definition (ER schema consitence)

An ER schema $\mathcal{S}$ is consistent if there at least one database state $\mathcal{B}$ that is legal w.r.t. $\mathcal{S}$.

Translating ER Schemas into DL

Translating ER Schemas into DL

ER Schema $\mathcal{S}$ is called binary if all relationships in $\mathcal{R}_{\mathcal{S}}$ are of arity 2.

Translating ER Schemas into DL

ER Schema $\mathcal{S}$ is called binary if all relationships in $\mathcal{R}_\mathcal{S}$ are of arity 2.

In the following we will learn how to translate any binary ER schema $\mathcal{S}$ in an $\mathcal{ALCQHI}$ TBox $\mathcal{T}_\mathcal{S}$.

ER Schema $\mathcal{S}$ is called binary if all relationships in $\mathcal{R}_{\mathcal{S}}$ are of arity 2. In the following we will learn how to translate any binary ER schema $\mathcal{S}$ in an $\mathcal{ALCQHI}$ TBox $\mathcal{T}_{\mathcal{S}}$.

Note: for ER schemas with higher arity one requires DL with n-ary relations, such as $\mathcal{DLR}$ (see DL Handbook).

Translating ER Schemas into DL (cont.)

Definition (Translating ER into DL)

Given a binary ER schema $\mathcal{S}$, we define $\mathcal{T}_\mathcal{S}$ as an $\mathcal{ALCQHI}$ TBox over vocabulary $N_\mathsf{I} = \emptyset$, $N_\mathsf{C} = \mathcal{E}_\mathcal{S} \cup \mathcal{D}_\mathcal{S}$, $N_\mathsf{R} = \mathcal{R}_\mathcal{S} \cup \mathcal{A}_\mathcal{S}$ and the following axioms:

- $E_1 \sqsubseteq E_2$ for each two entities s.t. $E_1 \preceq_\mathcal{S} E_2$
- $R_1 \sqsubseteq R_2$ for each two relationships s.t. $R_1 \preceq_\mathcal{S} R_2$
- $E \sqsubseteq (\forall A.D^A) \sqcap ({=}1\, A.\top)$ for every entity E and each attribute A associated with E
- and for each relationship R add:
  - $\exists R.\top \sqsubseteq R{:}1$ and $\top \sqsubseteq \forall R.R{:}2$
  - $R{:}1 \sqsubseteq {\geqslant} cmin_\mathcal{S}(R{:}2)\, R.R2$ if $cmin_\mathcal{S}(R{:}2) \neq 0$ and $R{:}1 \sqsubseteq {\leqslant} cmax_\mathcal{S}(R{:}2)\, R.R2$ if $cmax_\mathcal{S}(R{:}2) \neq \infty$
  - $R{:}2 \sqsubseteq {\geqslant} cmin_\mathcal{S}(R{:}1)\, R^-.R1$ if $cmin_\mathcal{S}(R{:}1) \neq 0$ and $R{:}2 \sqsubseteq {\leqslant} cmax_\mathcal{S}(R{:}1)\, R^-.R1$ if $cmax_\mathcal{S}(R{:}1) \neq \infty$

Translating ER Schemas into DL (cont.)

Theorem (Calvanese et al., 1999)

Let $\mathcal{S}$ be an ER schema and let $\mathcal{T}_{\mathcal{S}}$ be the respective $\mathcal{ALCQHI}$ TBox. There exists a legal database state of $\mathcal{S}$ iff there exists a finite model of $\mathcal{T}_{\mathcal{S}}$.

Note: Hence the problem of checking ER schema consistence reduces into finite-model satisfiability of DL TBoxes.

Translating ER Schemas into DL (cont.)

Using DL reasoners we can now automatically check:

- Schema consistence: is there at least one legal DB state for $\mathcal{S}$?
- Entity/relationship satisfiability: is there a legal DB state with $E^{\mathcal{B}}$ ($R^{\mathcal{B}}$) non-empty?
- Redundancy: are there two entities s.t. $E \prec_{\mathcal{S}} F$ and $F \prec_{\mathcal{S}} E$?

DL also extends capabilities of ER schemas:

- Refinement of properties along I-SA hierarchy
- Introducing sufficient conditions
- Definition of classes (i.e., entities) by means of complex properties