Lecture 4: Relations between Database Schemas

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

Relations between Databases

Multiple Database Setting

In realistic applications more then one inter-related databases may be employed.

- Data in the databases may partially overlap
- Constraints may be found inside each database but as well involve multiple databases

Formally, we model these databases by ER schemas $\mathcal{S}_1, \ldots, \mathcal{S}_n$. Constraints between elements inside each schema are naturally expressed by IS-A relations, and other ER constraints (intra-schema relations). However there can also be constraints between elements from different schemas (inter-schema relations).

Inter-Schema Relations

Inter-schema relations are expressed on both intensional and extensional levels. (Recall: intension of an entity refers to its conceptual definition; its extension refers directly to the objects belonging to the entity.)

Inter-Schema Relations

Inter-schema relations are expressed on both intensional and extensional levels. (Recall: intension of an entity refers to its conceptual definition; its extension refers directly to the objects belonging to the entity.)

Given two entities E_1 (from $\mathcal{S}_i$) and E_2 (from $\mathcal{S}_j$) inter-schema relations are defined as follows:

- $E_1 =_{\mathrm{int}} E_2$: E_1 and E_2 are intensionally equivalent – if $\mathcal{S}_i$ and $\mathcal{S}_j$ referred to the same set of objects, extensions of E_1 and E_2 would be equivalent

Inter-Schema Relations

Inter-schema relations are expressed on both intensional and extensional levels. (Recall: intension of an entity refers to its conceptual definition; its extension refers directly to the objects belonging to the entity.)

Given two entities E_1 (from $\mathcal{S}_i$) and E_2 (from $\mathcal{S}_j$) inter-schema relations are defined as follows:

- $E_1 =_{\text{int}} E_2$: E_1 and E_2 are intensionally equivalent
- $E_1 \leq_{\text{int}} E_2$: E_1 is intentionally more specific then E_2 – if $\mathcal{S}_i$ and $\mathcal{S}_j$ referred to the same set of objects, extension of E_1 would be contained within the extension of E_2

Inter-Schema Relations

Inter-schema relations are expressed on both intensional and extensional levels. (Recall: intension of an entity refers to its conceptual definition; its extension refers directly to the objects belonging to the entity.)

Given two entities E_1 (from $\mathcal{S}_i$) and E_2 (from $\mathcal{S}_j$) inter-schema relations are defined as follows:

- $E_1 =_{\text{int}} E_2$: E_1 and E_2 are intensionally equivalent
- $E_1 \leq_{\text{int}} E_2$: E_1 is intentionally more specific then E_2
- $E_1 =_{\text{ext}} E_2$: E_1 and E_2 are extensionally equivalent – extensions of E_1 and E_2 are equivalent

Inter-Schema Relations

Inter-schema relations are expressed on both intensional and extensional levels. (Recall: intension of an entity refers to its conceptual definition; its extension refers directly to the objects belonging to the entity.)

Given two entities E_1 (from $\mathcal{S}_i$) and E_2 (from $\mathcal{S}_j$) inter-schema relations are defined as follows:

- $E_1 =_{\mathrm{int}} E_2$: E_1 and E_2 are intensionally equivalent
- $E_1 \leq_{\mathrm{int}} E_2$: E_1 is intentionally more specific then E_2
- $E_1 =_{\mathrm{ext}} E_2$: E_1 and E_2 are extensionally equivalent
- $E_1 \leq_{\mathrm{ext}} E_2$: E_1 is extensionally more specific then E_2 – extension of E_1 is contained within the extension of E_2

Given ER schemas $\mathcal{S}_1, \ldots, \mathcal{S}_n$ and their corresponding DB states $\mathcal{B}_1, \ldots, \mathcal{B}_n$, the semantics of inter-schema relations is as follows:

- $E_1 =_{\mathrm{int}} E_2$ requires $E_1^{\mathcal{B}_i} \cap \Delta^{\mathcal{B}_j} = E_2^{\mathcal{B}_j} \cap \Delta^{\mathcal{B}_i}$
- $E_1 \leq_{\mathrm{int}} E_2$ requires $E_1^{\mathcal{B}_i} \cap \Delta^{\mathcal{B}_j} \subseteq E_2^{\mathcal{B}_j}$
- $E_1 =_{\mathrm{ext}} E_2$ requires $E_1^{\mathcal{B}_i} = E_2^{\mathcal{B}_j}$
- $E_1 \leq_{\mathrm{ext}} E_2$ requires $E_1^{\mathcal{B}_i} \subseteq E_2^{\mathcal{B}_j}$

for any entities E_1 of $\mathcal{S}_i$ and E_2 of S_j where $1 \leq i, j \leq n$ and $i \neq j$.

Example

Assume ER schemas S_1 (study results information system: students, professors, courses, ...), S_2 (students' body of academic senate: all senators who are students), S_3 (correspondence seminar organization: allows student and non-student members). We have e.g. the following inter-schema relations:

$$\text{Student}_3 \leq_{\text{ext}} \text{Student}_1$$
$$\text{Senator}_2 \leq_{\text{ext}} \text{Student}_1$$
$$\text{Senator}_2 \leq_{\text{int}} \text{Student}_3$$

Note: in the examples we will always assume that E_j belongs to schema S_j.

Moving to DL

Straightforward transformation into DL:

- $S_i \quad \rightsquigarrow \quad \mathcal{T}_{\mathcal{S}_i}$
- $E_1 =_{\mathrm{int}} E_2 \quad \rightsquigarrow \quad E_1 \equiv_{\mathrm{int}} E_2$
- $E_1 \leq_{\mathrm{int}} E_2 \quad \rightsquigarrow \quad E_1 \sqsubseteq_{\mathrm{int}} E_2$
- $E_1 =_{\mathrm{ext}} E_2 \quad \rightsquigarrow \quad E_1 \equiv_{\mathrm{ext}} E_2$
- $E_1 \leq_{\mathrm{ext}} E_2 \quad \rightsquigarrow \quad E_1 \sqsubseteq_{\mathrm{ext}} E_2$

Moving to DL (cont.)

Semantics of inter-schema axioms is analogous: given DL Tboxes $\mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}$ and the respective DL interpretations $\mathcal{I}_1, \dots, \mathcal{I}_n$, and some concepts E_1 of $\mathcal{T}_{\mathcal{S}_i}$ and E_2 of $\mathcal{T}_{\mathcal{S}_j}$:

- $E_1 \equiv_{\mathrm{int}} E_2$ requires $E_1^{\mathcal{I}_i} \cap \Delta^{\mathcal{I}_j} = E_2^{\mathcal{I}_j} \cap \Delta^{\mathcal{I}_i}$
- $E_1 \sqsubseteq_{\mathrm{int}} E_2$ requires $E_1^{\mathcal{I}_i} \cap \Delta^{\mathcal{I}_j} \subseteq E_2^{\mathcal{I}_j}$
- $E_1 \equiv_{\mathrm{ext}} E_2$ requires $E_1^{\mathcal{I}_i} = E_2^{\mathcal{I}_j}$
- $E_1 \sqsubseteq_{\mathrm{ext}} E_2$ requires $E_1^{\mathcal{I}_i} \subseteq E_2^{\mathcal{I}_j}$

In fact only $\sqsubseteq_{\mathrm{ext}}$ is necessary and other inter-schema relations can be reduced:

- $E_1 \equiv_{\mathrm{ext}} E_2 \quad \rightsquigarrow \quad E_1 \sqsubseteq_{\mathrm{ext}} E_2$ and $E_2 \sqsubseteq_{\mathrm{ext}} E_1$
- $E_1 \sqsubseteq_{\mathrm{int}} E_2 \quad \rightsquigarrow \quad E_1 \sqcap \top_j \sqsubseteq_{\mathrm{ext}} E_2$
- $E_1 \equiv_{\mathrm{int}} E_2 \quad \rightsquigarrow \quad E_1 \sqsubseteq_{\mathrm{int}} E_2$ and $E_2 \sqsubseteq_{\mathrm{int}} E_1$

for any concepts E_1 of $\mathcal{T}_{\mathcal{S}_i}$, E_2 of $\mathcal{T}_{\mathcal{S}_j}$, and for $\top_j$ the top concept of $\mathcal{T}_{\mathcal{S}_j}$, where $1 \leq i, j \leq n$ and $i \neq j$.

Example (cont.)

Let $\mathcal{S}_1, \ldots, \mathcal{S}_3$ be as before and let us have in addition $\mathcal{S}_4$ (faculty body of academic senate) and $\mathcal{S}_5$ (financial information system). Thanks to DL can express even more complex inter-schema relations:

$$\mathsf{Senator}_2 \sqsubseteq_{\mathrm{ext}} \neg\mathsf{Senator}_4$$
$$\mathsf{Student}_3 \sqcap \mathsf{Member}_3 \sqsubseteq_{\mathrm{ext}} \exists\mathsf{receives}_5.\mathsf{Scholarship}_5$$
$$\mathsf{Student}_3 \sqcap \mathsf{Member}_3 \sqsubseteq_{\mathrm{ext}} \forall\mathsf{hasGrading}_1.\mathsf{GradeA}_1$$

Cooperative Database Systems

Cooperative Database Systems

In cooperative database systems (CDS) we deal with n ER schemas $\mathcal{S}_1, \ldots, \mathcal{S}_n$ whose association in modeled with inter-schema relations.

Cooperative Database Systems

In cooperative database systems (CDS) we deal with n ER schemas $\mathcal{S}_1, \dots, \mathcal{S}_n$ whose association in modeled with inter-schema relations.

Formally a CDS is modeled as $(\mathcal{T}_{\mathcal{S}_0}, \mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}, \Sigma)$ where:

- $\mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}$ represent $\mathcal{S}_1, \dots, \mathcal{S}_n$
- $\mathcal{T}_{\mathcal{S}_0}$ represents common knowledge, which can also be empty
- Σ contains inter-schema axioms, typically between any pair of $\mathcal{T}_{\mathcal{S}_0}, \dots, \mathcal{T}_{\mathcal{S}_n}$

A model of a CDS consists of $n + 1$ DL interpretations $(\mathcal{I}_0, \dots, \mathcal{I}_n)$ s.t. $I_i \models \mathcal{T}_{\mathcal{S}_i}$ for $i \geq 0$ and $\Delta^{\mathcal{I}_i} \subseteq \Delta^{\mathcal{I}_0}$ for $i > 0$.

In cooperative database systems (CDS) we deal with n ER schemas $\mathcal{S}_1, \dots, \mathcal{S}_n$ whose association in modeled with inter-schema relations.

Formally a CDS is modeled as $(\mathcal{T}_{\mathcal{S}_0}, \mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}, \Sigma)$ where:

- $\mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}$ represent $\mathcal{S}_1, \dots, \mathcal{S}_n$
- $\mathcal{T}_{\mathcal{S}_0}$ represents common knowledge, which can also be empty
- Σ contains inter-schema axioms, typically between any pair of $\mathcal{T}_{\mathcal{S}_0}, \dots, \mathcal{T}_{\mathcal{S}_n}$

A model of a CDS consists of $n+1$ DL interpretations $(\mathcal{I}_0, \dots, \mathcal{I}_n)$ s.t. $I_i \models \mathcal{T}_{\mathcal{S}_i}$ for $i \geq 0$ and $\Delta^{\mathcal{I}_i} \subseteq \Delta^{\mathcal{I}_0}$ for $i > 0$.

CDS allow e.g. to check and maintain global consistency of databases used inside an organization/application.

Database Schema Integration

Integrated Database Schemas

In integrated database schemas (IDS) we deal with n ER schemas $\mathcal{S}_1, \ldots, \mathcal{S}_n$ which are integrated into a global (enterprise) schema $\mathcal{S}_0$ using inter-schema relations.

Integrated Database Schemas

In integrated database schemas (IDS) we deal with n ER schemas $\mathcal{S}_1, \ldots, \mathcal{S}_n$ which are integrated into a global (enterprise) schema $\mathcal{S}_0$ using inter-schema relations.

Formally an IDS is modeled as $(\mathcal{T}_{\mathcal{S}_0}, \mathcal{T}_{\mathcal{S}_1}, \ldots, \mathcal{T}_{\mathcal{S}_n}, \Sigma)$ where:

- $\mathcal{T}_{\mathcal{S}_1}, \ldots, \mathcal{T}_{\mathcal{S}_n}$ represent $\mathcal{S}_1, \ldots, \mathcal{S}_n$
- $\mathcal{T}_{\mathcal{S}_0}$ represents the global schema, which is always non-empty
- Σ contains inter-schema axioms, typically all between $\mathcal{T}_{\mathcal{S}_0}$ and some other $\mathcal{T}_{\mathcal{S}_i}$, $i > 0$.

A model of an IDS consists of $n+1$ DL interpretations $(\mathcal{I}_0, \ldots, \mathcal{I}_n)$ s.t. $I_i \models \mathcal{T}_{\mathcal{S}_i}$ for $i \geq 0$ and $\Delta^{\mathcal{I}_i} \subseteq \Delta^{\mathcal{I}_0}$ for $i > 0$.

Integrated Database Schemas

In integrated database schemas (IDS) we deal with n ER schemas $\mathcal{S}_1, \dots, \mathcal{S}_n$ which are integrated into a global (enterprise) schema $\mathcal{S}_0$ using inter-schema relations.

Formally an IDS is modeled as $(\mathcal{T}_{\mathcal{S}_0}, \mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}, \Sigma)$ where:

- $\mathcal{T}_{\mathcal{S}_1}, \dots, \mathcal{T}_{\mathcal{S}_n}$ represent $\mathcal{S}_1, \dots, \mathcal{S}_n$
- $\mathcal{T}_{\mathcal{S}_0}$ represents the global schema, which is always non-empty
- Σ contains inter-schema axioms, typically all between $\mathcal{T}_{\mathcal{S}_0}$ and some other $\mathcal{T}_{\mathcal{S}_i}$, $i > 0$.

A model of an IDS consists of $n+1$ DL interpretations $(\mathcal{I}_0, \dots, \mathcal{I}_n)$ s.t. $I_i \models \mathcal{T}_{\mathcal{S}_i}$ for $i \geq 0$ and $\Delta^{\mathcal{I}_i} \subseteq \Delta^{\mathcal{I}_0}$ for $i > 0$.

IDS allow esp. to express a unified view and unified query interface over multiple databases.

Lietrature

1. Borgida, A., Manzerini, M., Rpsoati, R.: Description logics and databases. In: The Description Logics Handbook, 2nd ed., Oxford University Press 2003. – start here
2. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., Rosati, R.: Description logic framework for information integration. In: KR 1998, Morgan Kaufmann 1998.
3. Calvanese, D., Lenzerini, M., Nardi, D.: Unifying class-based representation formalisms. J. Artif. Intell. Res. (JAIR) 11, 199-240, 1999
4. Catarci, T., Lenzerini, M,: Representing and using interschema knowledge in cooperative information systems. Int. J. Cooperative Inf. Syst. 2(4), 375-398, 1993.

Lietrature

1. Borgida, A., Manzerini, M., Rpsoati, R.: Description logics and databases. In: The Description Logics Handbook, 2nd ed., Oxford University Press 2003.
2. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., Rosati, R.: Description logic framework for information integration. In: KR 1998, Morgan Kaufmann 1998. – more about representing ER schemas in $\mathcal{DLR}$ DL
3. Calvanese, D., Lenzerini, M., Nardi, D.: Unifying class-based representation formalisms. J. Artif. Intell. Res. (JAIR) 11, 199-240, 1999
4. Catarci, T., Lenzerini, M,: Representing and using interschema knowledge in cooperative information systems. Int. J. Cooperative Inf. Syst. 2(4), 375-398, 1993.

Lietrature

1. Borgida, A., Manzerini, M., Rpsoati, R.: Description logics and databases. In: The Description Logics Handbook, 2nd ed., Oxford University Press 2003.
2. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., Rosati, R.: Description logic framework for information integration. In: KR 1998, Morgan Kaufmann 1998.
3. Calvanese, D., Lenzerini, M., Nardi, D.: Unifying class-based representation formalisms. J. Artif. Intell. Res. (JAIR) 11, 199-240, 1999 – more about representing ER in $\mathcal{ALUNI}$ DL
4. Catarci, T., Lenzerini, M,: Representing and using interschema knowledge in cooperative information systems. Int. J. Cooperative Inf. Syst. 2(4), 375-398, 1993.

Lietrature

1. Borgida, A., Manzerini, M., Rpsoati, R.: Description logics and databases. In: The Description Logics Handbook, 2nd ed., Oxford University Press 2003.
2. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., Rosati, R.: Description logic framework for information integration. In: KR 1998, Morgan Kaufmann 1998.
3. Calvanese, D., Lenzerini, M., Nardi, D.: Unifying class-based representation formalisms. J. Artif. Intell. Res. (JAIR) 11, 199-240, 1999
4. Catarci, T., Lenzerini, M,: Representing and using interschema knowledge in cooperative information systems. Int. J. Cooperative Inf. Syst. 2(4), 375-398, 1993. – more about DL in cooperative database systems