

Lógica para Ciencias de la Computación

Clase 15 - Algebra Relacional - Cálculo Relacional de Tuplas - Cálculo Relacional de Dominio - Datalog - Description Logic

Laura Cecchi

`lcecchi@fi.uncoma.edu.ar`

Depart. de Teoría de la Computación - Facultad de Informática
Universidad Nacional del Comahue

Primer Cuatrimestre de 2016

Contenido

Bases de Datos

Sistemas basados en Ontologías

Bibliografía

What is a Database?

A **Database** is a collection of data with a correlated meaning which are *stored* and *manipulated*.

A database represents some aspect of the real world called the mini-world or **Universe of Discourse (UoD)**.

Álgebra Relacional

- ▶ The relational algebra is **The Query Language** for the relational model. **SQL** can be intended as an implementation of the *relational algebra*.
- ▶ The *Relational Algebra* provides a collection of operations that can be used to construct new relations from old ones.

Las operaciones del álgebra relacional están divididas en dos grupos:

Conjuntos

Nombre	Símbolo
Unión	\cup
Diferencia	$-$
Intersección	\cap
Producto Cartesiano	\times

Relacionales

Nombre	Símbolo
Proyección	$\pi_A(R)$
Selección	$\sigma_C(R)$
Join Natural	\bowtie
θ -Join	\bowtie_{θ}

Cálculo Relacional

- ▶ TUPLAS
- ▶ DOMINIO

Poder Expresivo Equivalente entre los Lenguajes

- ▶ **ALGEBRA RELACIONAL**
- ▶ **CALCULO RELACIONAL de TUPLAS**
(restringido a expresiones seguras).
- ▶ **CALCULO RELACIONAL de DOMINIOS**
(restringido a expresiones seguras).

Recordemos que una EXPRESIÓN SEGURA en el Cálculo Relacional es aquella que siempre produce un número finito de tuplas como resultado.

Limitante: ¿Es posible expresar la clausura transitiva de una relación?

Uno scherzo...



Datalog: Base de Conocimiento

Una distinción entre el modelo relacional y los modelos Datalog es que en éste último hay 2 tipos de conocimiento:

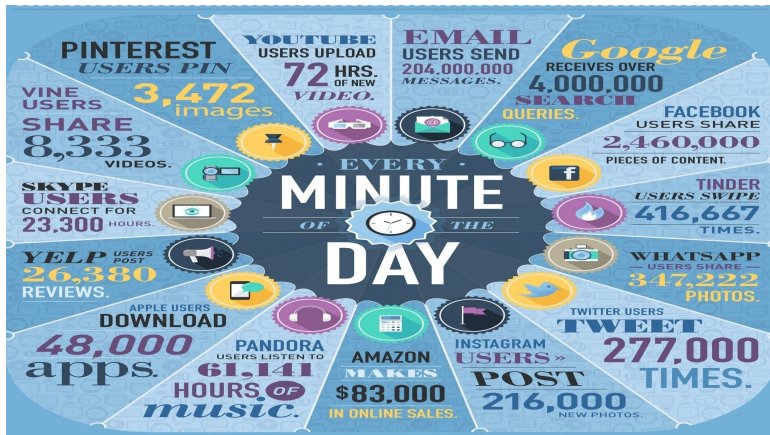
- ▶ **Extensional** (EDB): Un predicado cuya relación es almacenada en la base de datos. En un modelo relacional todas las relaciones son extensionales. Ejemplo, $\text{padre}(X,Y)$, $\text{camino_directo}(X,Y)$
- ▶ **Intensional** (IDB): Un predicado definido por reglas lógicas. Ejemplo, $\text{ancestro}(X,Y)$ o $\text{hermano}(X,Y)$ o $\text{camino}(X,Y)$.

Cada símbolo de predicado representa una relación EDB o IDB, pero no ambas

Datalog: Base de Conocimiento

Datalog sin recursión tiene el mismo poder expresivo que el álgebra relacional.

Data never sleeps



New challenges in information management

Information management is a key challenge in complex systems today:

- ▶ The volume of information to manage is enormous.
- ▶ Data increases with incredible velocity.
- ▶ The variety of information has increased: data is distributed and heterogeneous
- ▶ The meaning of data is variable, and depends on the context.
- ▶ The veracity of the data needs to be questioned and assessed –incompleteness, inconsistency, lack of precision.
- ▶ To understand complex data it needs to be visualized.
- ▶ Data increasingly represents an important value for an organization.

New challenges in information management

There is an increased need to access data in a uniform and integrated way, extract information, and perform various forms of analysis on it.

Traditional data management systems are not sufficient anymore to fulfill today's information management requirements.

The role of Knowledge Representation

- ▶ Provee una caracterización precisa de una base de conocimiento, lo que involucra caracterizar el **tipo de conocimiento** a ser especificado por el sistema, como también definir claramente los **servicios de razonamiento** que el sistema necesita proveer (clases de consultas que el sistema debería responder).
- ▶ Énfasis en la **SEMÁNTICA** de los datos: fundamental para compartir, entender y razonar sobre los datos.

Ontologías

Ontología en Ciencias de la Computación

Una **ontología** es un esquema de representación que describe a una **conceptualización formal** de un dominio de interés.

Ontologías

La especificación de una ontología comprende dos niveles diferentes:

- ▶ **Nivel Intensional**: especifica el conjunto de elementos conceptuales y las restricciones/axiomas que describen la estructura conceptual del dominio del problema.
TBox(taxonomía) contruido a través de declaraciones que describen propiedades generales de los conceptos.
- ▶ **Nivel Extensional**: especifica un conjunto de instancias de los elementos conceptuales descriptos en el nivel intensional
ABox (aserciones) depende de un conjunto de circunstancias, variando en el tiempo.

Clasificación de los lenguajes de ontologías

- ▶ **Basados en Grafos**
 - ▶ UML Class Diagrams
 - ▶ Entity-Relationship Diagrams
 - ▶ Semantic networks
 - ▶ Conceptual graphs
- ▶ **Basados en Frame:** Frame Systems
- ▶ **Basados en Lógica:**
 - ▶ Description Logics (e.g., SHOIQ, DLR, DL-Lite , OWL, . . .)
 - ▶ Rules (e.g., RuleML, LP/Prolog)
 - ▶ First Order Logic (e.g., KIF)
 - ▶ Non-classical logics (e.g., non-monotonic, probabilistic)

Description Logic

Los fundamentos formales de los lenguajes de ontología se encuentran en la lógica y específicamente en las **Lógicas de Descripción o Description Logic**

Description Logics son fragmentos de la lógica de primer orden que se ajustan específicamente para la representación de la estructura del conocimiento:

- ▶ Basándonos en los formalismos lógicos, la información es provista con una semántica formal.
- ▶ La formalización basada en lógica permite proveer soporte automático para las tareas relacionadas al Data Management, a través de la inferencia basada en lógica
- ▶ Los aspectos computacionales son de interés, luego las herramientas deben proveer un soporte efectivo al razonamiento automático.

Arquitectura de un Sistema basado en Description Logics

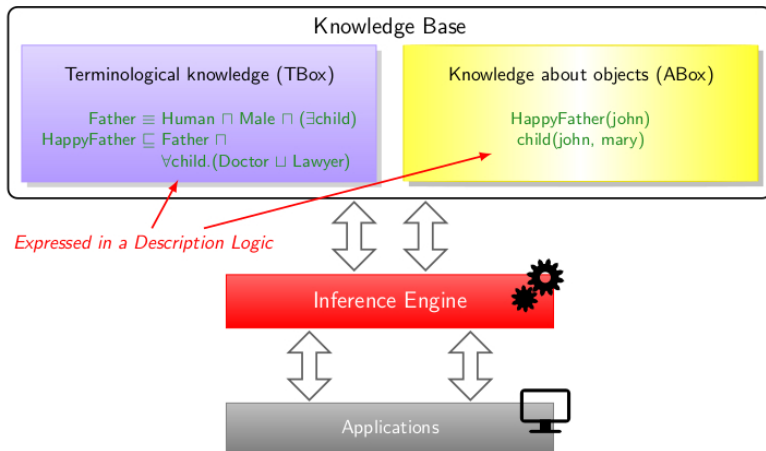


Imagen extraída del Curso de Alessandro Artale.

Description Language

A description language provides the means for defining:

- ▶ **concepts**, corresponding to classes: interpreted as sets of objects;
- ▶ **roles**, corresponding to relationships: interpreted as binary relations on objects.

To define concepts and roles:

- ▶ We start from a (countably infinite) alphabet of concept names and role names, forming so called **atomic concepts and roles**.
- ▶ Then, by applying specific constructors, we can build **complex concepts and roles**, starting from the atomic ones.

Semántica Formal

The **formal semantics** of DLs is given in terms of interpretations.

Def.: An **interpretation** $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of:

- a nonempty set $\Delta^{\mathcal{I}}$, called the **interpretation domain** (of \mathcal{I})
- an **interpretation function** $\cdot^{\mathcal{I}}$, which maps
 - each atomic concept A to a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$
 - each atomic role P to a subset $P^{\mathcal{I}}$ of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Imagen extraída del Curso de Alessandro Artale.

El Lenguaje Básico \mathcal{AL} (Attributive Language)

Construct	Syntax	Example	Semantics
atomic concept	A	Doctor	$A^I \subseteq \Delta^I$
atomic role	P	hasChild	$P^I \subseteq \Delta^I \times \Delta^I$
atomic negation	$\neg A$	\neg Doctor	$\Delta^I \setminus A^I$
conjunction	$C \sqcap D$	Hum \sqcap Male	$C^I \cap D^I$
(unqual.) exist. res.	$\exists R$	\exists hasChild	$\{ o \mid \exists o'. (o, o') \in R^I \}$
value restriction	$\forall R.C$	\forall hasChild.Male	$\{ o \mid \forall o'. (o, o') \in R^I \rightarrow o' \in C^I \}$
bottom	\perp		\emptyset

(C , D denote arbitrary concepts and R an arbitrary role)

Imagen extraída del Curso de Alessandro Artale.

Axiomas Terminológicos

Definición

Tienen la forma

$$C \sqsubseteq D \quad (R \sqsubseteq S)$$

llamados **inclusión** o

$$C \equiv D \quad (R \equiv S)$$

llamados **igualdades**.

Siendo C y D conceptos y R y S roles.

Axiomas Terminológicos

Definición

Tienen la forma $C \sqsubseteq D$ ($R \sqsubseteq S$) llamados **inclusión** o $C \equiv D$ ($R \equiv S$) llamados **igualdades**.

Ejemplos

- ▶ Todo administrador es un empleado

$$\textit{Administrador} \sqsubseteq \textit{Empleado}$$

- ▶ Una mujer es una persona de sexo femenino

$$\textit{Woman} \equiv \textit{Person} \sqcap \textit{Female}$$

- ▶ Un hombre es una persona que no es una mujer.

$$\textit{Man} \equiv \textit{Person} \sqcap \neg \textit{Woman}$$

Axiomas Terminológicos

Definición

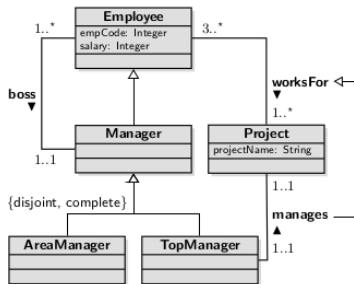
Tienen la forma $C \sqsubseteq D$ ($R \sqsubseteq S$) llamados **inclusión** o $C \equiv D$ ($R \equiv S$) llamados **igualdades**.

Semántica de los axiomas

Una interpretación \mathcal{I} satisface:

- ▶ una inclusión $C \sqsubseteq D$ si $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
- ▶ una igualdad $C \equiv D$ si $C^{\mathcal{I}} = D^{\mathcal{I}}$.

Ejemplo: UML a DL



$\text{Manager} \sqsubseteq \text{Employee}$
 $\text{AreaManager} \sqsubseteq \text{Manager}$
 $\text{TopManager} \sqsubseteq \text{Manager}$
 $\text{AreaManager} \sqsubseteq \neg \text{TopManager}$
 $\text{Manager} \sqsubseteq \text{AreaManager} \sqcup \text{TopManager}$
 $\exists \text{salary}_{\text{Emp}} \sqsubseteq \text{Integer}$
 $\exists \text{salary}_{\text{Emp}} \sqsubseteq \text{Employee}$
 $\text{Employee} \sqsubseteq \exists \text{salary}_{\text{Emp}} \sqcap (\leq 1 \text{ salary}_{\text{Emp}})$
 $\exists \text{worksFor} \sqsubseteq \text{Employee}$
 $\exists \text{worksFor} \sqsubseteq \text{Project}$
 $\text{Employee} \sqsubseteq \exists \text{worksFor}$
 $\text{Project} \sqsubseteq (\geq 3 \text{ worksFor})$
 $\text{manages} \sqsubseteq \text{worksFor}$
 ...

Imagen extraída del Curso de Alessandro Artale.

Más allá de \mathcal{AL}

Construct	\mathcal{AL}	Syntax	Semantics
disjunction	\mathcal{U}	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
top		\top	$\Delta^{\mathcal{I}}$
qual. exist. res.	\mathcal{E}	$\exists R.C$	$\{ o \mid \exists o'. (o, o') \in R^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}} \}$
(full) negation	\mathcal{C}	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
number restrictions	\mathcal{N}	$(\geq k R)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}}\} \geq k \}$
		$(\leq k R)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}}\} \leq k \}$
qual. number restrictions	\mathcal{Q}	$(\geq k R.C)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}}\} \geq k \}$
		$(\leq k R.C)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}}\} \leq k \}$
inverse role	\mathcal{I}	R^{-}	$\{ (o, o') \mid (o', o) \in R^{\mathcal{I}} \}$
role closure	$_{reg}$	R^{*}	$(R^{\mathcal{I}})^{*}$

Imagen extraída del Curso de Alessandro Artale.

Relación entre Clases de Complejidad

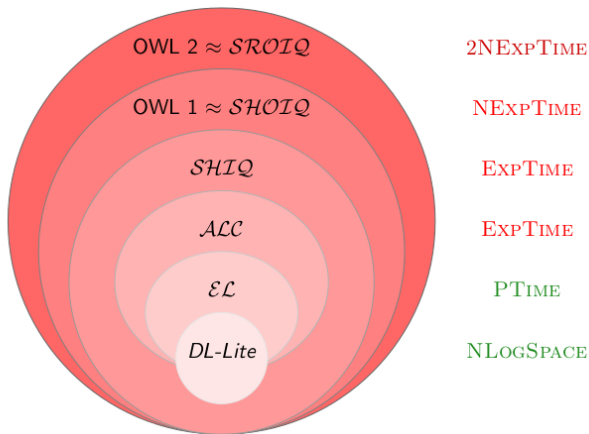
Las siguientes relaciones son conocidas:

$$\begin{aligned} AC^0 \subsetneq LogSpace \subseteq NLogSpace \subseteq PTime \subseteq \\ \subseteq NP \subseteq PSpace \subseteq \\ \subseteq ExpTime \subseteq NExpTime \end{aligned}$$

Además se conoce que

$$PTime \subsetneq ExpTime$$

Diferentes Description Logics



Nota: $ALC = \mathcal{S}$

Imagen extraída del Curso de Alessandro Artale.

Bibliografía

[Baader et al., 2003] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors.

The Description Logic Handbook: Theory, Implementation and Applications.

Cambridge University Press, 2003.