2.3.- Model Relacional de Dades (Aproximació Lògica)

Existeixen dos llenguatges lògics de manipulació per al model relacional:

- El Càlcul Relacional de Tuples.
- El Càlcul Relacional de Dominis.

La perspectiva lògica del model relacional de dades permet de realitzar consultes i establir restriccions de forma molt més simple que des de la perspectiva algebraica.

El **Càlcul Relacional de Tuples** és en el que es basa el llenguatge de manipulació SQL.

2.3.1.- Lògica de primer ordre

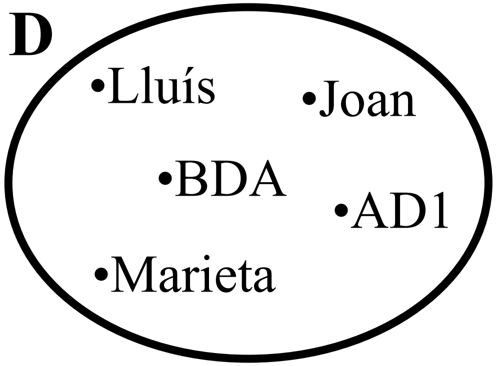
Lògica de 1er ordre

- "Sistema formal que permet de raonar sobre un univers de discurs"
- La lògica de 1er ordre és un llenguatge formal. Per tant, posseeix dos elements:
  - Un **llenguatge** en el qual es poden expressar assercions sobre l'univers d'interés. (SINTAXI)
  - unes **regles** amb les quals es pot determinar el valor de veritat de les assercions o afirmacions realitzades. (SEMÀNTICA)

Exemple: "Mortal(Sòcrates)" i "Planeta(Sòcrates)" són sintàcticament correctes, però només la primera és certa semànticament en el nostre món.

2.3.1.- Lògica de primer ordre

D



P:

- •ser un alumne
- •ser una assignatura
- •estar matriculat un alumne en una assignatura

ASSERCIÓ: "Tots els alumnes estan matriculats d'alguna assignatura"

¿És certa aquesta asserció?

Es necessita tindre més informació sobre les propietats de ***P*** en el domini ***D***

Si aquesta informació és:

- es-alumne = {Joan, Lluís, Marieta}
- es-assignatura = {AD1, BDA}
- està-matriculat = {(Lluís,AD1), (Joan,BDA)}

L'asserció és <u>falsa</u> per al coneixement que es té de les propietats ***P*** en el domini ***D***

2.3.1.- Lògica de primer ordre

FORMALITZACIÓ (sintaxi):

- S'ha de definir un llenguatge de 1er ordre ***L*** per a poder referir-se als individus i a les propietats de l'univers de discurs:

  L:

  Constants = {Lluís, Marieta, Joan, AD1, BDA}

  Predicats = {Alumne(.), Assignatura(.), Matrícula(.,.)}

  Variables = {x, y}

  Connectives = $\{\rightarrow, \neg, \wedge, \vee\}$

  Quantificadors = $\{\forall, \exists\}$

- Exemple de fórmula sintàcticament correcta:

  F: $\forall x\ (\text{Alumne}(x) \rightarrow \exists y\ \text{Matrícula}(x, y))$

- Exemple de fórmula sintàcticament incorrecta:

  F': $\forall\text{Alumne}\ \exists\text{Matrícula}(x, y)$

2.3.1.- Lògica de primer ordre

FORMALITZACIÓ (SEMÀNTICA):

- La interpretació ***I*** del llenguatge de 1er ordre en el domini ***D,*** *corresponent a l'exemple anterior*:

D = {Lluís, Marieta, Joan, AD1, BDA}

Alumne = {Joan, Lluís, Marieta}

Assignatura = {AD1, BDA}

Matrícula = {(Lluís, AD1), (Joan, BDA)}

L'avaluació de **F:** $\forall x$ (Alumne(x) $\rightarrow$ $\exists y$ Matrícula(x, y)) en ***I*** es realitza seguint unes regles fixes, que veiem a continuació.

2.3.1.- Lògica de primer ordre

AVALUACIÓ D'UNA FÓRMULA EN LÒGICA DE PRIMER ORDRE (SEMÀNTICA)

Donades:

- Una fórmula F.
- Una interpretació I.
- Un domini D.
- Una assignació de valores a les variables en F.

Les regles per a l'avaluació de F són com segueix:

2.3.1.- Lògica de primer ordre

- 1) Si F és una comparació:
  - si F és de la forma $X \alpha Y$ on X i Y són constants o variables, aleshores F s'avalua al valor de veritat de la comparació.

- 2) Si F és un predicat n-ari de la forma $R(x1, ..., xn)$, aleshores F s'avalua a cert si $(x1, ..., xn)$ pertany a la interpretació de R en I (o siga, a la base de dades); si no, F s'avalua a fals.

- 3) Si F és de la forma (G), F s'avalua al valor de veritat de G.

2.3.1.- Lògica de primer ordre

4) Si F és d'una de les següents formes $\neg G$, $G \wedge H$, $G \vee H$ o $G \rightarrow H$ a on G i H són fórmules ben formades, llavors F s'avalua d'acord a les següents taules de veritat:

G	H	F = G ∧ H	F = G ∨ H	F = G → H
fals	fals	fals	fals	cert
indefinit	fals	fals	indefinit	indefinit
cert	fals	fals	cert	fals
fals	indefinit	fals	indefinit	cert
indefinit	indefinit	indefinit	indefinit	indefinit
cert	indefinit	indefinit	cert	indefinit
fals	cert	fals	cert	cert
indefinit	cert	indefinit	cert	cert
cert	cert	cert	cert	cert

G	F = ¬ G
fals	cert
indefinit	indefinit
cert	fals

2.3.1.- Lògica de primer ordre

- 5) Si F és de la forma $\exists X\ G$, llavors F és certa si existeix una assignació per a la variable X que fa certa la fórmula G.

- 6) Si F és de la forma $\forall X\ G$, llavors F és certa si per a tota assignació de la variable X, la fórmula G és certa.

2.3.1.- Lògica de primer ordre

FORMALITZACIÓ (SEMÀNTICA):

AVALUACIÓ DE FÓRMULES OBERTES I TANCADES

- Les fórmules tancades s'utilitzen per a expressar afirmacions (restriccions)
- Les fórmules obertes s'utilitzen per a expressar consultes.

Donada la interpretació *I* de l'exemple anterior:

D = {Lluís, Marieta, Joan, AD1, BDA}

Alumne = {Joan, Lluís, Marieta}

Assignatura = {AD1, BDA}

Matrícula = {(Lluís, AD1), (Joan, BDA)}

Una fórmula tancada**:**

$\forall x$ (Alumne(x) $\rightarrow \exists y$ Matrícula(x, y))

Una fórmula oberta**:**

Alumne(x) ^ Matrícula(x, 'AD1')

2.3.1.- Lògica de primer ordre

FORMALITZACIÓ (SEMÀNTICA):

AVALUACIÓ DE FÓRMULES TANCADES

L'avaluació de **F:** $\forall x\ (\text{Alumne}(x) \rightarrow \exists y\ \text{Matrícula}(x, y))$

Donada la interpretació *I* de l'exemple anterior:

D = {Lluís, Marieta, Joan, AD1, BDA}

Alumne = {Joan, Lluís, Marieta}

Assignatura = {AD1, BDA}

Matrícula = {(Lluís, AD1), (Joan, BDA)}

Seguint aquestes regles es podrà afirmar que: ***"F*** és falsa en ***I***"

2.3.1.- Lògica de primer ordre

FORMALITZACIÓ (SEMÀNTICA):

AVALUACIÓ DE FÓRMULES OBERTES

L'avaluació de **F:** Alumne(x) ^ Matrícula(x, 'AD1')

Donada la interpretació *I* de l'exemple anterior:

D = {Lluís, Marieta, Joan, AD1, BDA}

Alumne = {Joan, Lluís, Marieta}

Assignatura = {AD1, BDA}

Matrícula = {(Lluís, AD1), (Joan, BDA)}

AVALUACIÓ:

- Busca els valors del domini tal que, assignats a les variables lliures (*x* en aquest cas), fan la fórmula certa..

2.3.2.- Interpretació lògica d'una base de dades relacional

Una **interpretació** consisteix en associar a cada predicat n-ari del llenguatge una relació n-ària definida sobre el domini ***D***:

Alumne

Joan
Lluís
Marieta

Assignatura

AD1
BDA

Matrícula

Lluís	AD1
Joan	BDA

Per tant, una interpretació d'un llenguatge de 1er ordre pot veure's com una base de dades relacional en la que:

- els noms de relació coincideixen amb els predicats.
- els dominis dels atributs coincideixen amb les constants.

2.3.2.- Interpretació lògica d'una base de dades relacional

Província

p_id	nomprov
44	Teruel
46	Valencia
16	Cuenca
12	Castellón

Riu

r_id	nom
r1	Sénia
r2	Túria
r3	Xúquer

Passa_Per

p_id	r_id
44	r1
46	r2
30	r2
20	r1
44	r3
12	r1

Predicats: {Província(.,.) Riu(.,.) Passa_per(.,.)}

Interpretació: Les extensions de les relacions en la base de dades

F1: Riu(x,y) ^ Passa_Per(44,x) ⇒

F2: Riu(x,y) ^ ¬∃z Passa_Per(z,x) ➡

2.3.2.- Interpretació lògica d'una base de dades relacional

Variable tupla:

- Són variables que es declaren sobre les relacions de la base de dades *variable_tupla: nom_relació.*
- Els seus valors possibles es restringeixen a les tuples de la relació sobre la qual s'ha definit.
- Els seus components es poden referir com: *Variable_tupla.atribut_de_relació.*

EXEMPLE:

Riu(r_id:cadena(6), name:cadena(20))

RX : Riu

Valors possible per a ***RX***:

{(r_id: "r1"), (nom: "Sénia")}
{(r_id: "r3"), (nom: "Xúquer")}
~~{(r_id: "xx"), (nom: "xftrfsdh")}~~
~~{(r_id: "r2"), (nom: "Tajo")}~~
~~{(r_id: "r3"), (nom: "Túria")}~~

id_r	nom
r1	Sénia
r2	Túria
r3	Xúquer

2.3.2.- Interpretació lògica d'una base de dades relacional

Consultes amb variables tuples:

- Les consultes amb variables tuples tenen la següent forma:

  {*declaració_de_variables_lliures*|*fórmules_ben_formades*}

- Els exemples escrits en lògica de primer ordre poden reescriure's amb variables tuples de la següent manera:

Lògica de primer ordre:

F1: Riu(x,y) ^ Passa_Per(44,x)

F2: Riu(x,y) ^ ¬∃z Passa_Per(z,x)

Lògica de primer ordre amb variables tupla:

F1: RX:Riu | ∃PPX:Passa_Per (PPX.r_id = RX.r_id ^ PPX.p_id = 44)

F2: RX:Riu | ¬∃PPX:Passa_Per (RX.r_id = PPX.r_id)

2.3.2.- Interpretació lògica d'una base de dades relacional

Consultes amb variables tupla:

EXEMPLES:

Riu(r_id:r_id_dom, nom:nom_dom, talla:talla_dom)

Província(p_id:p_id_dom, nom:nom_dom)

Passa_Per(p_id:p_id_dom, r_id:r_id_dom)

Quins rius travessen dues províncies almenys?

Quins rius travessen totes les províncies?