Relational translation - 1

Knowledge objectives

1. Name the two meanings of the NULL value

Understanding Objectives

1. Explain the storage space problem generated by NULL values, and how DBMSs solve it
2. Explain the access time problem generated by NULL values, and how we should take it into account in the DB design
3. Explain the general way to translate classes and associations into relational model

Application Objectives

1. Translate a DB query from natural language into SQL, given a DB schema, taking into account the presence of NULL values

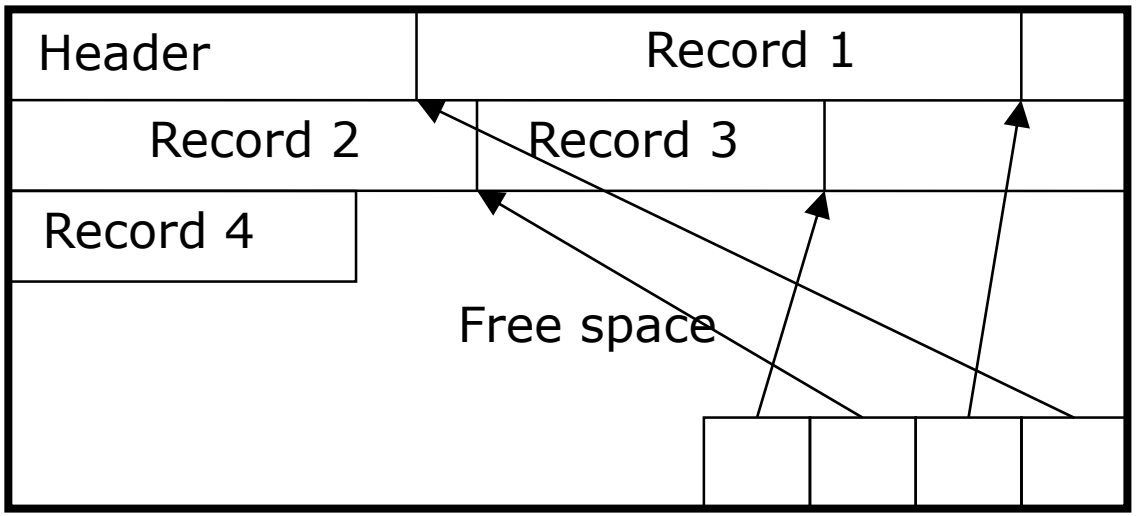
Which rows do we get?

```
SELECT *
FROM R
WHERE A=10 OR A<>10;
```

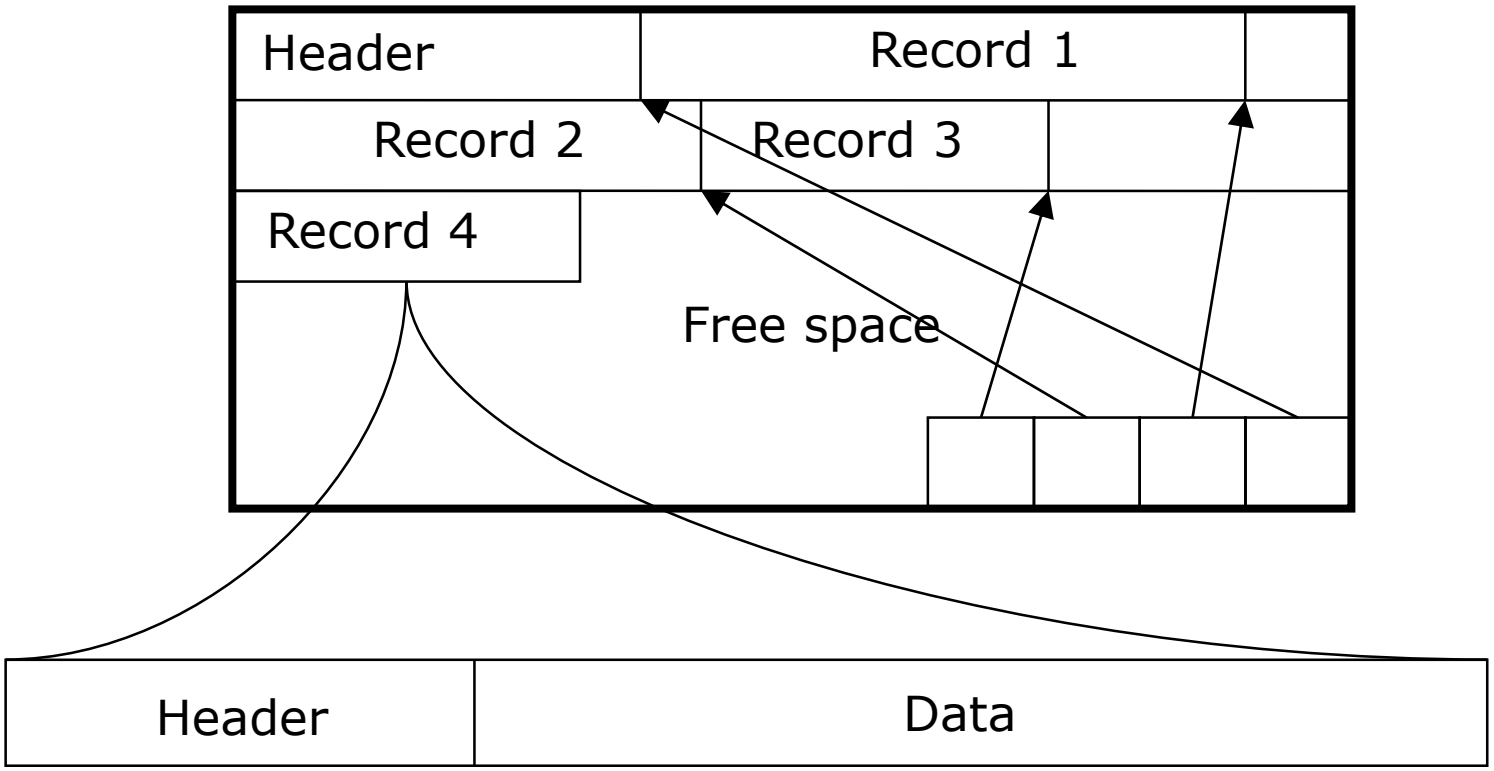
Null values

- Two meanings
- Reasons to use them:
  - Inserting a tuple with an unknown value
  - Adding a new attribute to a non-empty relation
  - Special aggregation cases
  - Avoiding exceptions in aggregations with unknown values
- Representation:
  - Different from any non-null value

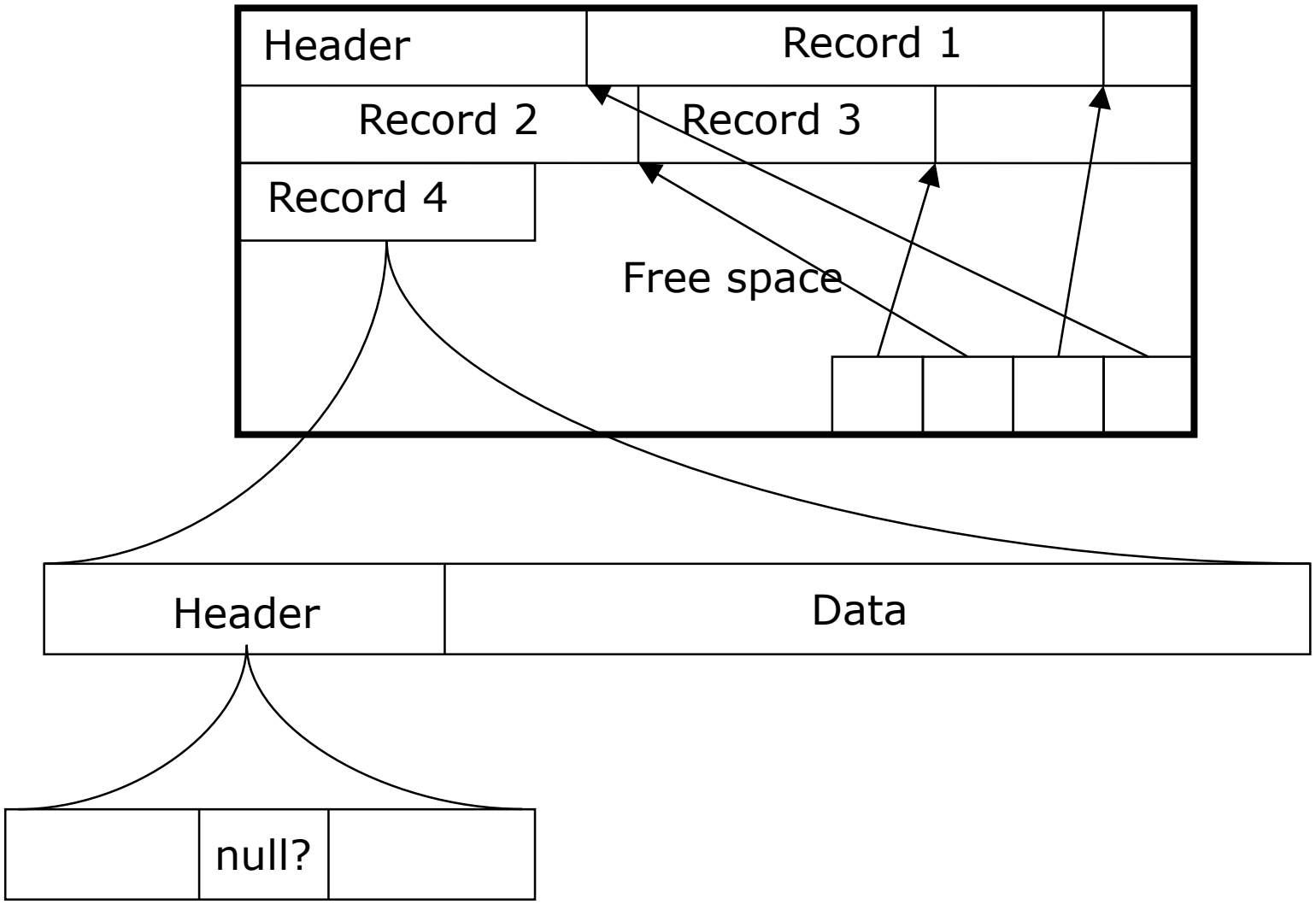
Null values storage



Null values storage



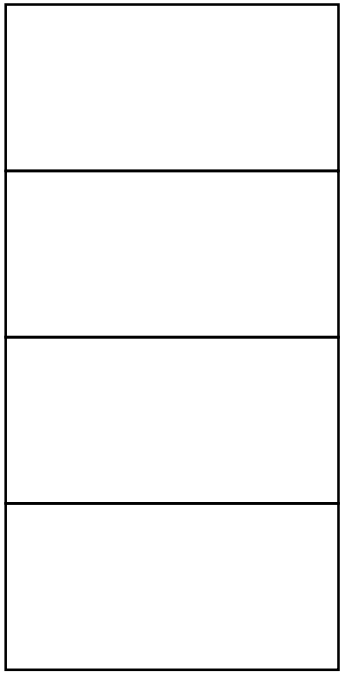
Null values storage



Null values query

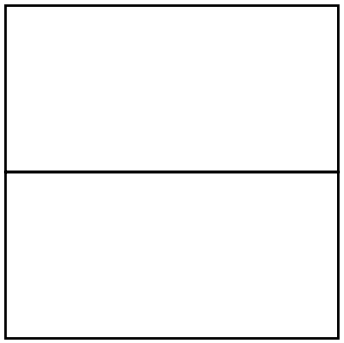
a) SELECT name
FROM People
WHERE #labour>0;

People

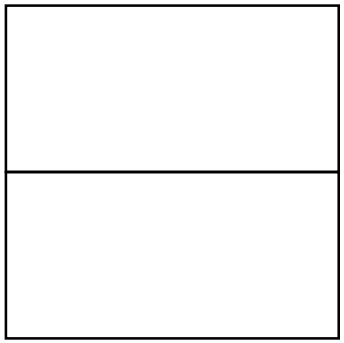


b) SELECT name
FROM Women
WHERE #labour>0;

Men



Women



Ternary logic for null values

NULL θ X -> UNKNOWN
NULL=NULL -> UNKNOWN

Ternary logic for null values

NULL θ X -> UNKNOWN
NULL=NULL -> UNKNOWN

NOT	
T	F
U	U
F	T

Ternary logic for null values

NULL θ X -> UNKNOWN
NULL=NULL -> UNKNOWN

NOT	
T	F
U	U
F	T

AND	T	U	F
T	T	U	F
U	U	U	F
F	F	F	F

Ternary logic for null values

NULL θ X -> UNKNOWN
NULL=NULL -> UNKNOWN

NOT	
T	F
U	U
F	T

AND	T	U	F
T	T	U	F
U	U	U	F
F	F	F	F

OR	T	U	F
T	T	T	T
U	T	U	U
F	T	U	F

Consequences of ternary logic

- Queries
  - Return rows when the predicate is true
  - Do not return those evaluating unknown
- Constraints
  - Raise an exception when the predicate is false
  - Do not raise anything when evaluates unknown

Effect of null values in aggregates

- COUNT
  - With "*", counts all tuples
  - With "a", counts those with non-null value for the attribute
- SUM
  - Adds only non-null values
  - Returns null if there are not non-null values
- MIN/MAX
  - Returns null if there are not non-null values
- AVG
  - Its result always coincide with SUM(a)/COUNT(a)

Content	SUM(a)	COUNT(*)	COUNT(a)	AVG(a)	SUM(a) / COUNT(a)	SUM(a) / COUNT(*)	MIN(a)
empty	null	0	0	null	null	null	null
null	null	1	0	null	null	null	null
null 0 1	1	3	2	0.5	0.5	0.33333	0

Effect of null values in usual other cases

- V IN (X_1, X_2, ...) is the same as V = X_1 OR V = X_2 OR...
  - NULL IN (X_1, X_2, ...) -> UNKNOWN
- GROUP BY
  - Null values are put onto a single group
- UNIQUE
  - Unique constraints treat null as different to everything (different to null and different to other values)
  - This makes every null a different null so that unique constraints accept multiple null values

Specific comparison for nulls

```
SELECT id FROM T WHERE a IS NULL;
```

vs

```
SELECT id FROM T WHERE a=NULL;
```

Specific comparison for nulls

```
SELECT id FROM T WHERE a IS NULL;
```

vs

~~SELECT id FROM T WHERE a=NULL;~~ 

Query examples with nulls

teachers living in a city where no student lives

```
SELECT id
FROM teachers
WHERE city NOT IN (SELECT city FROM students);
```

vs

```
SELECT id
FROM teachers
WHERE NOT EXISTS (    SELECT *
                      FROM students
                      WHERE teachers.city =
                            students.city);
```

Algebraic operations with nulls

R(	A,	B)
	?	?
	a	?
	a	1
	?	1

S(	A,	B)
	?	?
	a	?
	a	1

T(	A,	B,	C)
	?	?	z
	a	?	y
	?	1	x
	a	?	w
	?	1	v

R ∪ S	(A,	B)
	?	?
	a	?
	a	1
	?	1

R ∩ S	(A,	B)
	?	?
	a	?
	a	1

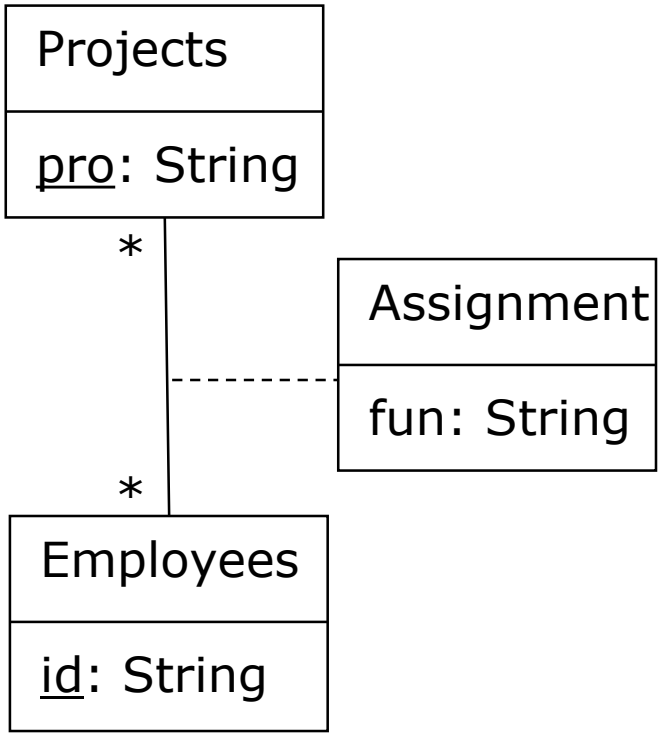
R - S	(A,	B)
	?	1

T[A,B]	(A,	B)
	?	?
	a	?
	?	1

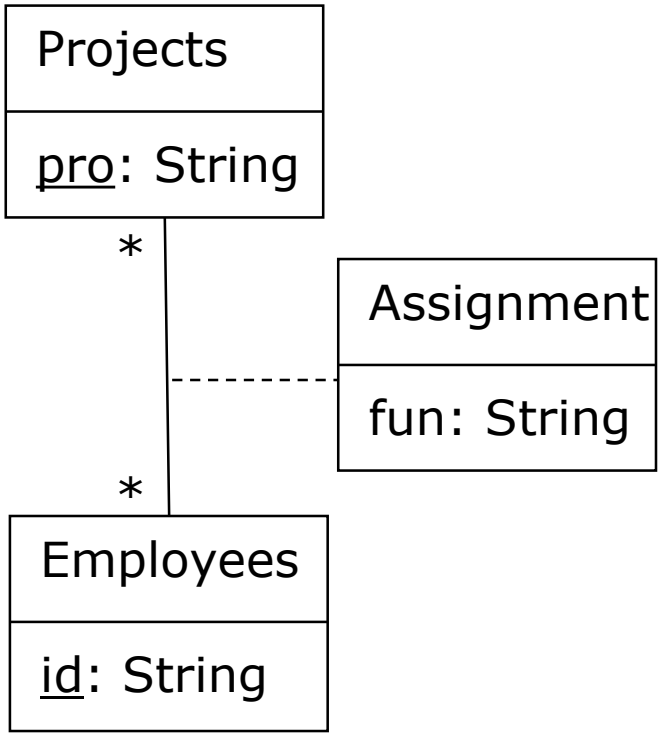
T[A]	(A)
	?
	a

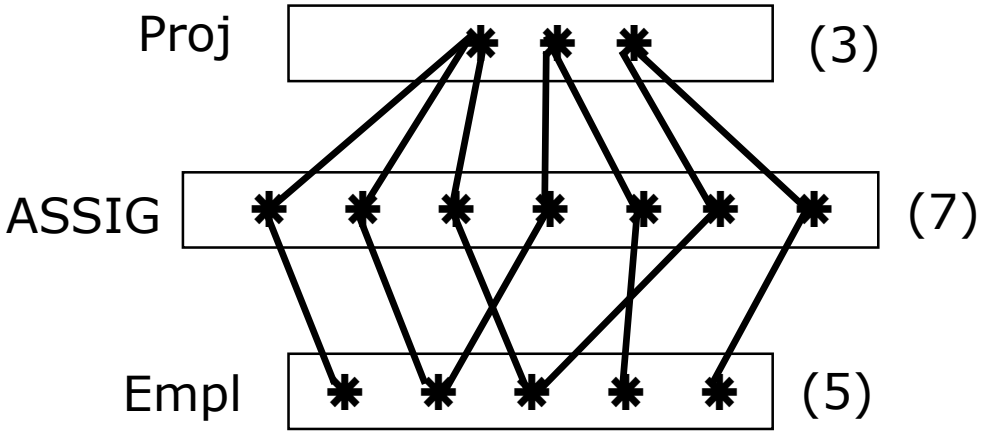
R(B = B)T	(A,	B,	A',	B',	C)
	a	1	?	1	x
	?	1	?	1	x
	a	1	?	1	v
	?	1	?	1	v

UML model and instantiation



UML model and instantiation





Relational model representation

- Long

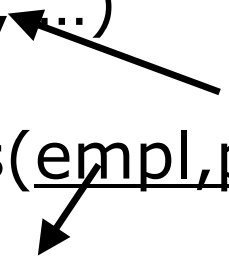
```
CREATE TABLE Projects (pro CHAR(25), …);
CREATE TABLE Employees (id CHAR(9), …);
CREATE TABLE Assignments (…);
```

- Short

Projects(pro,…)

Assignments(empl,pro,function)

Employees(id, …)



ONLY CORRECT INSTANTIATIONS OF UML DIAGRAM MUST BE POSSIBLE TO REPRESENTIN THE RELATIONAL SCHEMA

WITHOUT OID!!!

Summary

- Meaning of NULL values
- Consequences of NULL values
  - Space
  - Time
  - Writing queries
- General way to translate classes and associations

Bibliography

- Jaume Sistac et al. *Disseny de bases de dades*. Editorial UOC, 2002. Col·lecció Manuals, number 43
- J. Melton and A. Simon. *SQL 1999*. Morgan Kaufmann, 2002
- P. Gulutzan and T. Pelzer. *SQL-99 Complete, really*. R&D Books, 1999