



UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

Dipartimento di Scienze Fisiche,
Informatiche e Matematiche

Basi di Dati

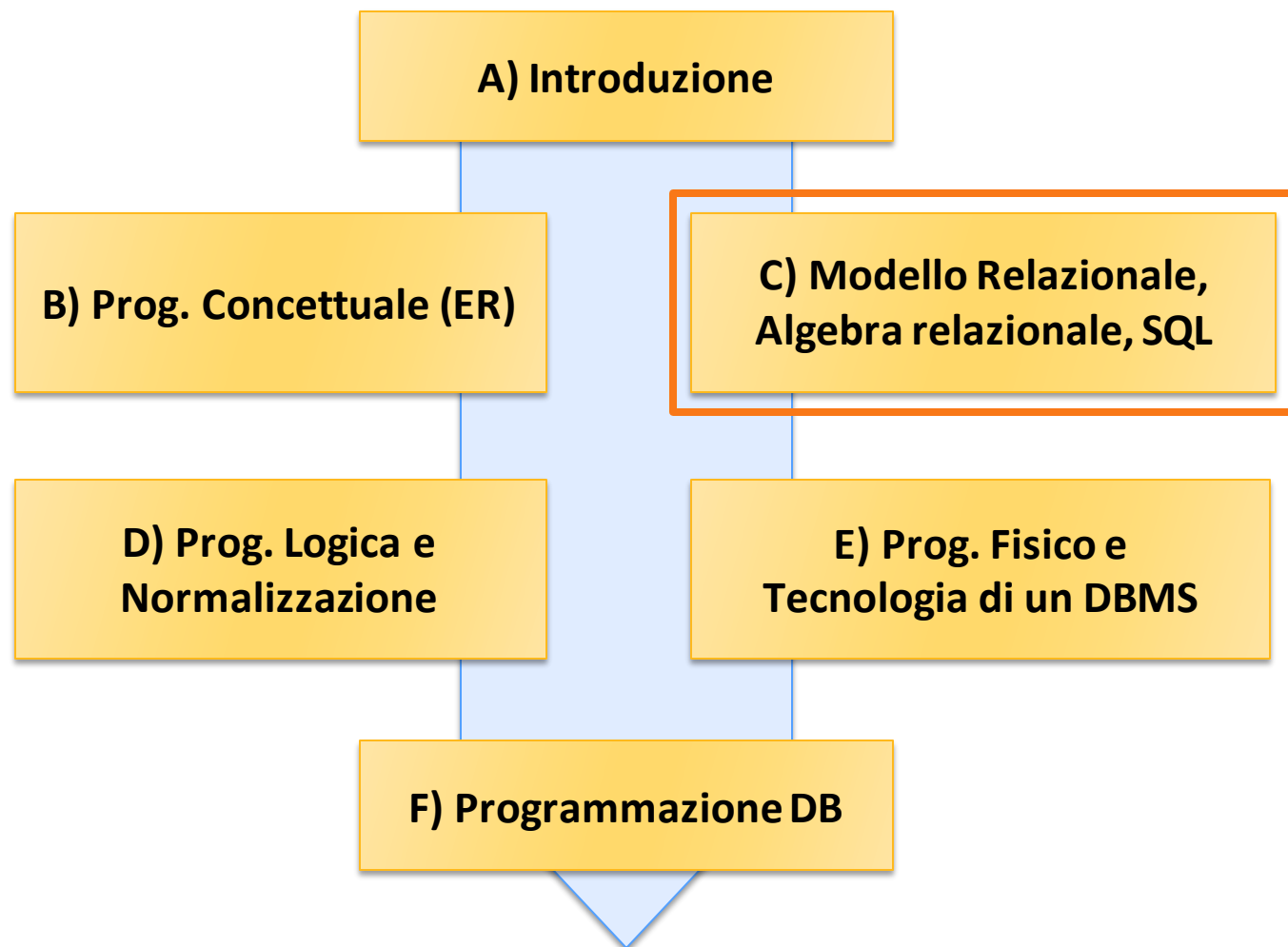
Corso di Laurea in Informatica

E' vietata la copia e la riproduzione dei contenuti e immagini in qualsiasi forma. E' inoltre vietata la redistribuzione e la pubblicazione dei contenuti e immagini non autorizzata espressamente dall'autore o dall'Università di Modena e Reggio Emilia.

Basi di Dati

Dichiarazione degli schemi e interrogazioni in SQL (I Parte)

Basi di Dati – Dove ci troviamo?



Storia di SQL

- ❑ Definito nell'ambito del progetto
 - ❑ **SYSTEM R** (IBM S. JOSE) nel **1976**
- ❑ Nome originario: **SEQUEL**
- ❑ Adottato progressivamente da tutti i sistemi commerciali
- ❑ Standardizzato da **ANSI** e **ISO**

Composizione di SQL

❓ Data Definition Language (DDL)

- ❓ definizione di domini, tabelle, indici, viste, vincoli, procedure, trigger

❓ Data Manipulation Language (DML)

- ❓ linguaggio di query, modifica, comandi transazionali

❓ Data Control Language (DCL)

- ❓ linguaggio per la gestione degli utenti e dei privilegi di accesso

Standardizzazione di SQL

❑ Progressione dello standard per estensioni quasi-compatibili

❑ prima versione nel 1986 (SQL-1)

❑ modifica alla prima versione nel 1989 (SQL-89)

❑ seconda versione nel 1992 (SQL-2, SQL-92)

❑ terza versione nel 1999 (SQL-3, SQL-99)

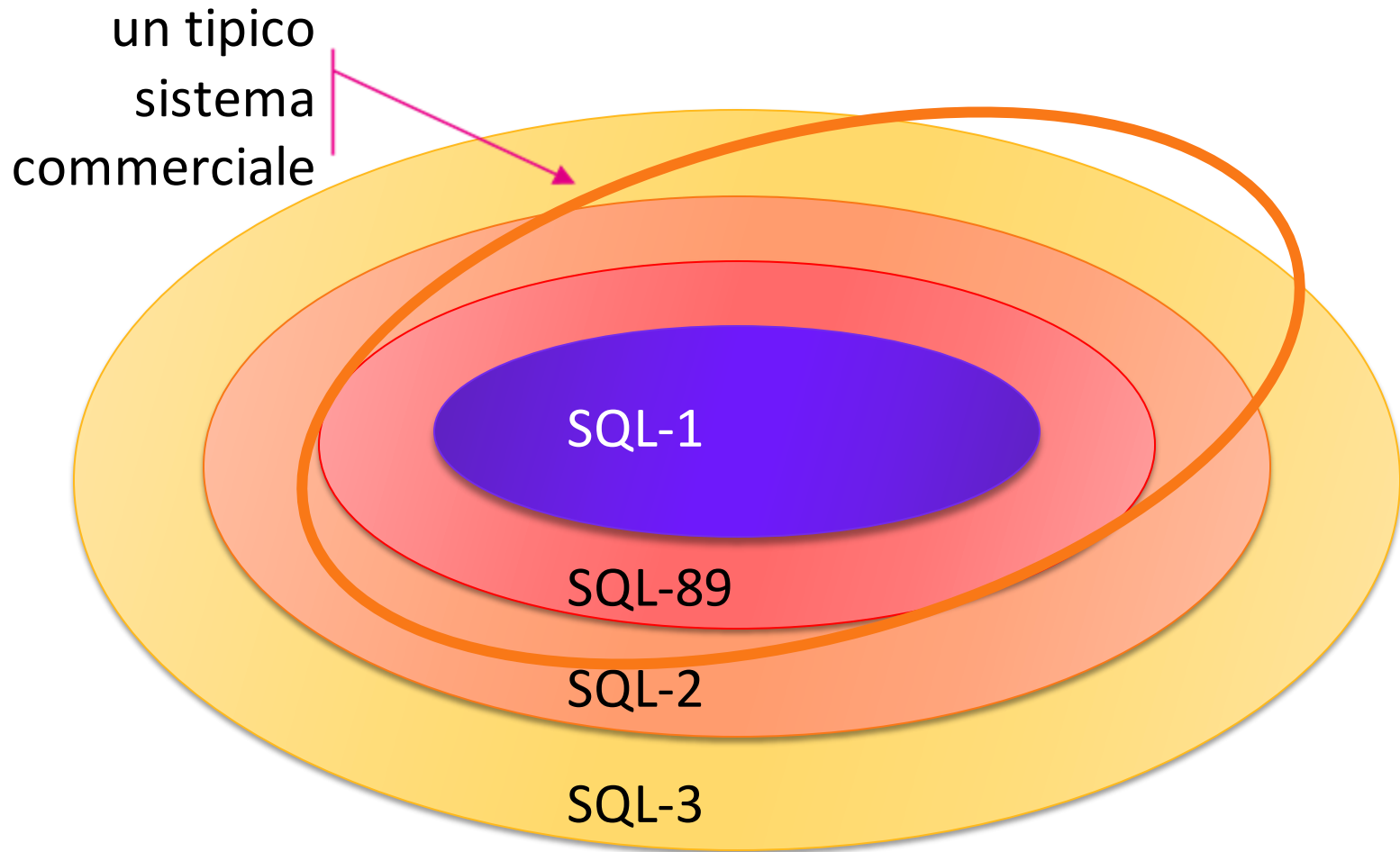
❑ In SQL-2:

❑ entry SQL (più o meno equivalente a SQL-89)

❑ intermediate SQL

❑ full SQL

Potere espressivo di standard e sistemi commerciali



Tipi di dati in SQL-2

stringhe

- ❓ CHAR (N)
- ❓ VARCHAR (N)

stringhe di bit (0/1)

- ❓ BIT (N)
- ❓ VARBIT (N)

numerici esatti

- ❓ NUMERIC (Prec, Scale)
(o anche DECIMAL)
- ❓ INTEGER
- ❓ SMALLINT

Precision e Scale

Esempio

23.5141

- ❓ Precision: 6
(numero di cifre significative in tutto il numero)
- ❓ Scale: 4
(numero di cifre decimali)

Tipi di dati in SQL-2 (cont.)

Numerici approssimati

❑ REAL

❑ DOUBLE PRECISION

Domini speciali

❑ DATE

❑ YYYY-MM-DD

❑ TIME(N)

❑ HH:MM:SS.NNNN

❑ TIMESTAMP

❑ YYYY-MM-DD HH:MM:SS

❑ INTERVAL

❑ INTERVAL YEAR(2) TO MONTH

❑ (tra 0 anni – 0 mesi e 99 anni – 11 mesi)

Il valore "null"

- ❑ **null** è un valore polimorfo (che appartiene a tutti i domini) col significato di valore non noto
- ❑ il valore esiste in realtà ma è **ignoto** al database
 - ❑ Es: data di nascita
- ❑ il valore è **inapplicabile**
 - ❑ Es: numero patente per minorenni

Definizione delle tabelle

- ❓ Una tabella è costituita da:
 - ❓ una lista di uno o più attributi (colonne)
 - ❓ un insieme di zero o più vincoli

```
CREATE TABLE <nome-tabella>  
(<nome-col> <dominio> [<vincoli-col>] ,  
...  
<nome-col> <dominio> [<vincoli-col>] ,  
  [<vincoli-tab>]  
)
```

Vincoli di colonna

? NOT NULL

- ? l'attributo non può assumere il valore null

? UNIQUE

- ? unicità dell'attributo

? PRIMARY KEY

- ? l'attributo è la chiave primaria

? CHECK

- ? esprime un generico vincolo sulla colonna tramite una espressione logico-relazionale

? REFERENCES

- ? esprime il vincolo della Foreign Key

Vincoli di tabella

❓ UNIQUE (<lista-colonne>)

❓ la combinazione dei valori delle colonne deve essere unica per tutte le tuple della tabella

❓ PRIMARY KEY (<lista-colonne>)

❓ chiave primaria della tabella (implica NOT NULL)

❓ FOREIGN KEY (<lista-colonne>) REFERENCES <tab> [(<lista-colonne>)]

❓ foreign key della tabella

❓ CHECK (<condizione>)

❓ predicato che deve essere soddisfatto per tutte le tuple della tabella

Esempi

CREATE TABLE STUDENTE

```
(  MATR CHAR(6) PRIMARY KEY,  
  NOME VARCHAR (30) NOT NULL,  
  CITTÀ VARCHAR (20),  
  C-DIP CHAR (3) )
```

CREATE TABLE CORSO

```
(  COD-CORSO CHAR(6) PRIMARY KEY,  
  TITOLO VARCHAR(30) NOT NULL,  
  DOCENTE VARCHAR(20) )
```

Esempi (cont.)

CREATE TABLE ESAME

```
(  MATR CHAR(6),  
    COD-CORSO CHAR(6),  
    DATA DATE NOT NULL,  
    VOTO SMALLINT NOT NULL,  
    PRIMARY KEY(MATR,COD-CORSO),  
    FOREIGN KEY (MATR) REFERENCES STUDENTI,  
    FOREIGN KEY (COD-CORSO) REFERENCES CORSO  
)
```



Approfondiremo il concetto di Foreign Key e la sua espressione nella prossima lezione

Chiavi alternative

- ❓ Come si esprime una chiave alternativa?
 - ❓ Specificando le clausole NOT NULL e UNIQUE

CREATE TABLE STUDENTE

(**MATR** CHAR(6) PRIMARY KEY,
 CF CHAR(16) NOT NULL UNIQUE,
 NOME VARCHAR (30) NOT NULL,
 CITTÀ VARCHAR (20),
 C-DIP CHAR (3))



Interrogazioni in SQL

Dichiaratività di SQL

- ❑ In SQL l'utente specifica **QUALE** informazione è di suo interesse ma non **COME** estrarla dai dati
- ❑ Il sistema costruisce una strategia di accesso (**QUERY OPTIMIZATION**)
- ❑ È l'aspetto più qualificante delle basi di dati relazionali

Struttura di SQL

- ❑ Basata sulla composizione di blocchi
 - ❑ **SELECT**
 - ❑ **FROM**
 - ❑ **WHERE**
- ❑ Ogni blocco ha il potere espressivo di una qualunque combinazione di selezioni, proiezioni e join

Esempio: gestione degli esami universitari

studente

MAT	NOME	CITTA'	C-DIP
R	Carlo	Bologna	Inf
123	Paola	Torino	Inf
415	Antonio	Roma	Log

esame

MAT R	COD- CORSO	DATA	VOTO
	1	2014-09-07	30
123	2	2015-01-08	28
123	2	2014-09-07	20

corso

COD- CORSO	TITOLO	DOCENTE
1	matematica	Barozzi
2	informatica	Natali

Interrogazioni semplici

SELECT *
FROM STUDENTE

MAT	NOME	CITTA'	C-DIP
R	Carlo	Bologna	Inf
123	Paola	Torino	Inf
415	Antonio	Roma	Log

702

SELECT *
FROM STUDENTE
WHERE C-DIP = 'Log'

MAT	NOME	CITTA	C-DIP
R	Antonio	'	Log

702

Roma

Selezione

```
SELECT *  
FROM STUDENTE  
WHERE NOME='Paola'
```

❓ Il risultato è una tabella (priva di nome) con schema:

❓ lo stesso schema di STUDENTE

❓ istanze :

❓ le tuple di STUDENTE che soddisfano il predicato di selezione

MAT	NOME	CITTA	C-DIP
R	Paola	'	Inf

Esempio di selezione

```
SELECT *  
FROM STUDENTE  
WHERE (CITTÀ='Torino') OR  
((CITTÀ='Roma') AND NOT (C-DIP='log'))
```

MAT	NOME	CITTÀ	C-DIP
R	Carlo	Bologna	Inf
123	Paola	Torino	Inf
415	Antonio	Roma	Log

702

Proiezione

```
SELECT MATR, CITTA'  
FROM STUDENTE
```

- Il risultato è una tabella (priva di nome) con schema:
 - gli attributi di proiezione
- istanze :
 - le tuple di STUDENTE ristrette a quegli attributi

MAT	CITTÀ
R	Bologna
123	Torino
415	Roma

Blocchi SQL per la modifica

❓ Tre operazioni elementari:

❓ Cancellazione: DELETE

❓ inserimento: INSERT

❓ modifica: UPDATE

❓ Cancellazione

❓ DELETE FROM STUDENTE WHERE MATR = '678678'

Inserimento

INSERT INTO STUDENTE

VALUES

('456878', 'Giorgio Rossi', 'Bologna', 'Logistica e Produzione')
[, (...)]

INSERT INTO BOLOGNESI

(
 SELECT *
 FROM STUDENTE
 WHERE CITTÀ = 'Bologna'
)

Modifica

UPDATE ESAME

SET VOTO = 30

WHERE DATA = 2014-04-01

UPDATE ESAME

SET VOTO = VOTO + 1

WHERE MATR = '787989'