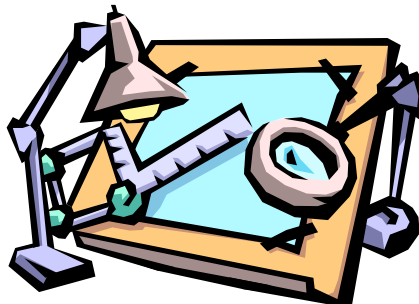


Progettazione di una base di dati



Dario Maio

<http://bias.csr.unibo.it/maio/>

Progettazione di basi di dati



1

Ciclo di vita di un S.I. (1)

- La progettazione di una base di dati è una delle attività principali del processo di sviluppo dei **sistemi informativi (SI)**.

ciclo di vita

- + definizione strategica (**strategic study**);
- + pianificazione (**information system planning**);
- + analisi dell'organizzazione (**business analysis**);
- + progettazione del sistema (**system design**);
- + progettazione esecutiva (**construction design**);
- + realizzazione e collaudo in fabbrica (**construction and workbench test**);
- + installazione (**installation**);
- + collaudo del sistema installato (**test of installed system**);
- + esercizio (**operation**);
- + evoluzione (**evolution**);
- + messa fuori servizio (**phase out**);
- + post mortem.

Progettazione di basi di dati



2



Ciclo di vita di un S.I. (2)

■ Definizione strategica

- ✦ Si assumono decisioni sulle aree aziendali che devono essere oggetto di automazione.

■ Pianificazione

- ✦ Si definiscono gli obiettivi e si evidenziano i fabbisogni; viene condotto uno **studio di fattibilità** per individuare possibili strategie d'attuazione e avere una prima idea dei costi, dei benefici e dei tempi.

■ Analisi dell'organizzazione (specificazione dei requisiti)

- ✦ Si formalizzano i requisiti, avvalendosi di tecniche di modellazione della realtà e si producono **macro-specifiche** per la fase di progettazione.

■ Progettazione del sistema

- ✦ Si interpretano i requisiti in una soluzione architettuale di massima. Sono prodotte **specifiche formali** indipendenti in linea teorica dai particolari strumenti che saranno usati per la costruzione del sistema.

■ Progettazione esecutiva

- ✦ Le specifiche del passo precedente sono rese vincolanti per lo staff addetto alla realizzazione, descrivendo la struttura dei componenti dell'architettura hardware, software e di rete. Queste specifiche devono essere tali da poter dar luogo, attraverso il ricorso a strumenti di sviluppo opportuni, a un prodotto funzionante.

Progettazione di basi di dati

3



Cosa e Come modellare

- La progettazione di una **base di dati** è una delle attività principali del processo di sviluppo di un **sistema informativo**.
- Il processo di analisi è incrementale e porta **per passi successivi** alla stesura di un insieme di documenti in grado di rappresentare un **modello dell'organizzazione** e comunicare, **in modo non ambiguo**, una descrizione esauriente, coerente e realizzabile dei vari **aspetti statici, dinamici e funzionali** di un SI.

Aspetti ontologici da modellare:

- ✦ **conoscenza concreta** (oggetti e fatti specifici, le loro caratteristiche e le loro interrelazioni);
- ✦ **conoscenza astratta** (fatti generali che descrivono la conoscenza concreta e regolano il modo in cui essa può evolvere);
- ✦ **conoscenza procedurale** (modalità per modificare la conoscenza concreta e ricavare altri fatti);
- ✦ **dinamica** (modalità per l'evoluzione nel tempo della conoscenza);
- ✦ **comunicazione** (modalità per accedere alla conoscenza).

Progettazione di basi di dati

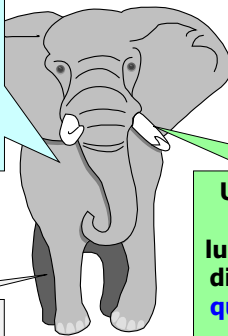
4

Oggetti, Funzioni, Stati

- ✚ Gli **oggetti** possono essere descritti a partire da termini molto generici (es. **edificio**) fino ad arrivare a livello di dettaglio specifici (es. **Palazzo Mazzini Marinelli, via Sacchi, 3, Cesena, ...**).
- ✚ Le **funzioni** possono essere espresse inizialmente in modo vago (es. **controllare il livello di gas nocivi nell'aria**) e successivamente precisate (es. **la programmazione del livello di soglia per l'allarme della centralina è attivata, dopo aver digitato un Pin, premendo il pulsante P**).
- ✚ Gli **stati** possono essere descritti a un elevato livello di astrazione (es. **la centralina è in stato di errore**) o specificati in maggior dettaglio (es. **è acceso il segnalatore d'errore nel sensore S**).

La favola dei tre ciechi...

Il secondo cieco toccò la pelle ispida dell'elefante. «Senza dubbio», disse, «si tratta di...cosa? Un porcospino? **Ma certo è proprio un porcospino!**»



Un cieco toccò l'estremità appuntita di una delle lunghe zanne dell'elefante e disse «**Ah! Ciò che abbiamo qui è un toro. Posso sentire le sue corna.**»

Il terzo cieco tastò una delle massicce zampe dell'elefante e disse: «**Questo in cui ci siamo imbattuti deve essere un albero.**»

Meccanismi di Astrazione



- Molteplici sono le relazioni in gioco fra oggetti, funzioni e stati e molteplici i livelli di possibile dettaglio:
- L'analista deve far ricorso a tecniche che gli consentano di organizzare, e interrogare, la conoscenza sul problema via via acquisita.
- I principali meccanismi di astrazione usati durante il processo di analisi per costruire una base di conoscenza sul problema sono **classificazione**, **generalizzazione**, **aggregazione**, **proiezione**.
- N.B. Nel seguito, negli esempi, l'attenzione sarà focalizzata principalmente sulla modellazione di oggetti.

Progettazione di basi di dati



7

Classificazione

- ✚ La classificazione consente di raggruppare in classi oggetti, funzioni, o stati in base alle loro proprietà. Ad esempio la **classe dei computer**.



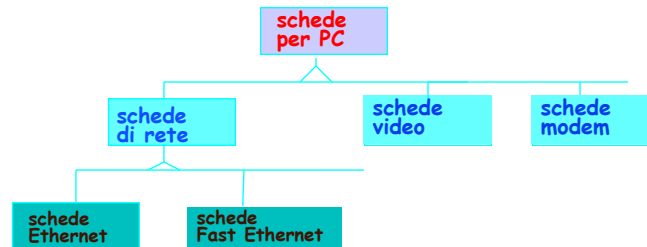
Progettazione di basi di dati



8

Generalizzazione

- La **generalizzazione** cattura le relazioni **è un** ovvero permette di astrarre le caratteristiche comuni fra più classi definendo **superclassi**.



- Ciò che caratterizza una scheda per PC è comune anche a ogni suo sottoinsieme, ovvero ogni sottoclasse **eredita** dalla superclasse ma può anche avere caratteristiche proprie.
- La **specializzazione** è il processo inverso della generalizzazione.

Progettazione di basi di dati

9

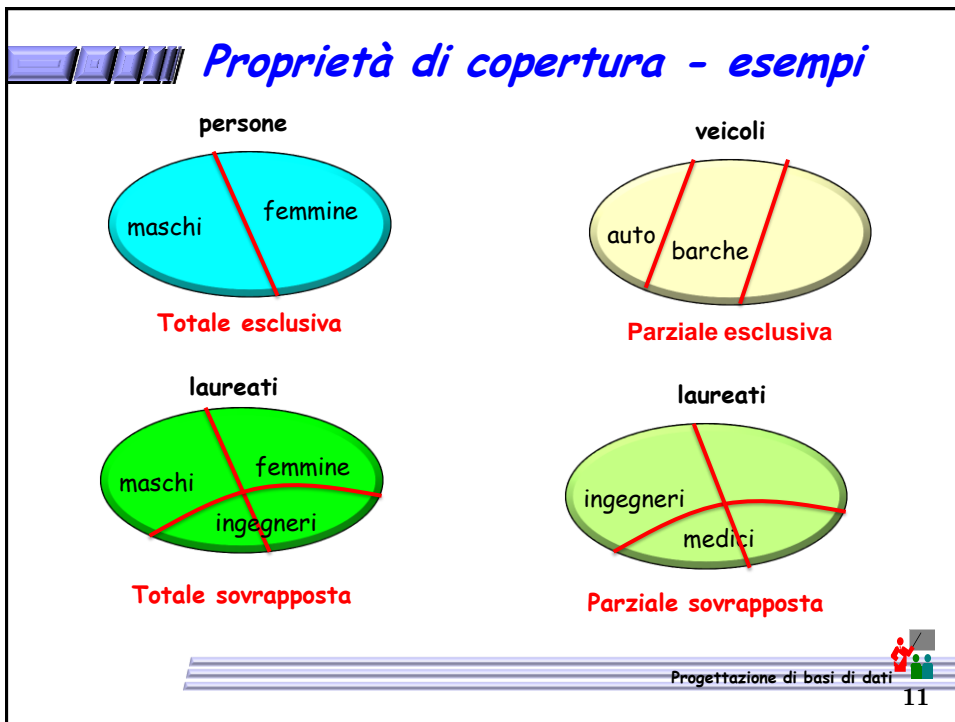
Copertura delle generalizzazioni

- Le generalizzazioni si caratterizzano per due dimensioni indipendenti
- Confronto fra unione delle specializzazioni e classe generalizzata
 - totale** se la classe generalizzata è l'unione delle specializzazioni;
 - parziale** se la classe generalizzata contiene l'unione delle specializzazioni.
- Confronto fra le classi specializzate
 - esclusiva** se gli insiemi delle specializzazioni sono fra loro disgiunti;
 - sovrapposta (overlapped)** se può esistere un'intersezione non vuota fra insiemi delle specializzazioni.
- Sono ovviamente possibili le quattro combinazioni

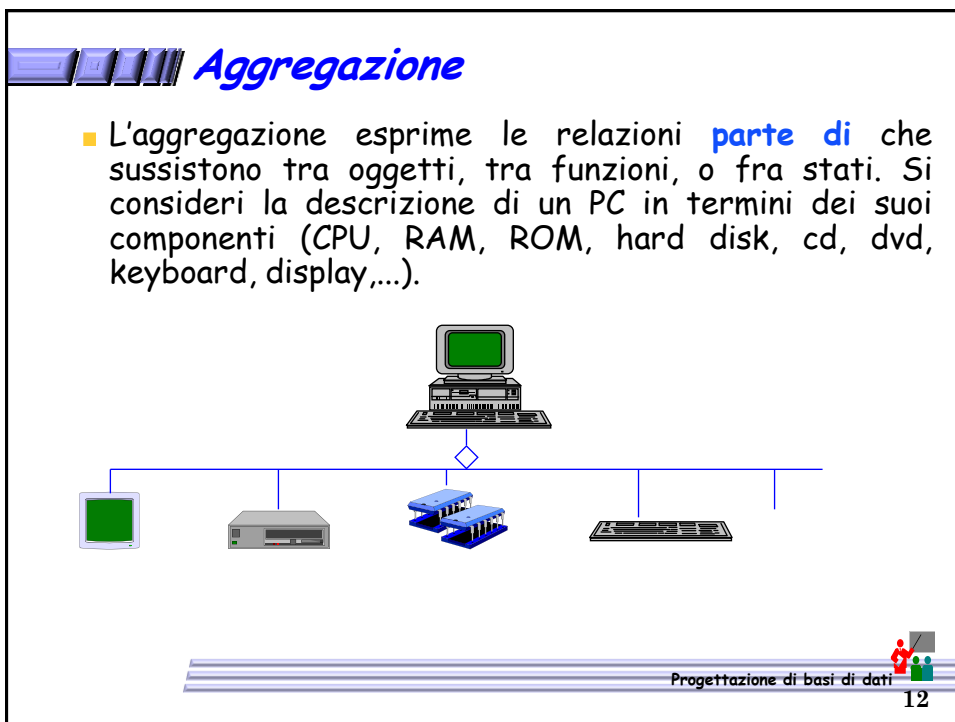
(t,e) (p,e) (t,o) (p,o)

Progettazione di basi di dati

10



11



12

Proiezione

- La proiezione cattura la vista delle relazioni strutturali fra gli oggetti, le funzioni, gli stati.
- Ad esempio, nel descrivere il funzionamento di un certo personal computer può essere necessario distinguere il **punto di vista** dell'operatore, dell'installatore, del programmatore.

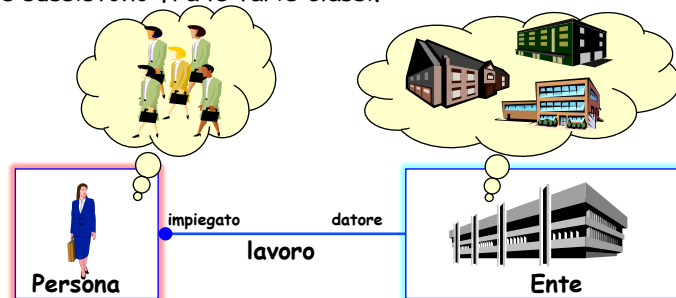


Progettazione di basi di dati

13

Associazioni

- Oltre ai meccanismi citati è importante modellare le associazioni che sussistono fra le varie classi.



- Le **associazioni** (corrispondenze tra classi) sono di fatto **aggregazioni** di cui le classi sono le componenti.
- Dal punto di vista della modellazione, è importante caratterizzare queste corrispondenze in termini di **vincoli di cardinalità** (un impiegato in quanti enti può lavorare?).

Progettazione di basi di dati

14



Vincoli di cardinalità

- ✦ Sia A un'associazione fra $C1$ e $C2$
 - ✦ $\text{min-card}(C1, A)$: cardinalità minima di $C1$ in A
 - ✦ è il minimo numero di corrispondenze nell'associazione A alle quali ogni istanza di $C1$ deve partecipare;
 - ✦ $\text{max-card}(C1, A)$: cardinalità massima di $C1$ in A
 - ✦ è il massimo numero di corrispondenze nell'associazione A alle quali ogni istanza di $C1$ può partecipare.
- ✦ **Vincoli di cardinalità minima**
 - ✦ **partecipazione opzionale**: $\text{min-card}(C1, A) = 0$
 - ✦ alcuni elementi di $C1$ possono non essere associati tramite A a elementi di $C2$;
 - ✦ **partecipazione obbligatoria (totale)**: $\text{min-card}(C1, A) > 0$
 - ✦ a ogni elemento di $C1$ deve essere associato, tramite A , almeno un elemento di $C2$.

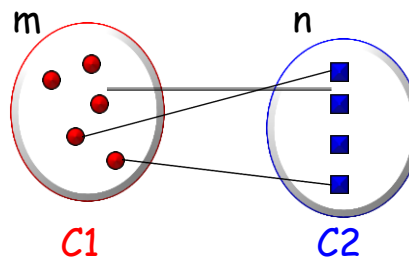


Vincoli di cardinalità massima(1)

Uno a uno (One-to-one)

$$\text{max-card}(C1, A) = 1$$

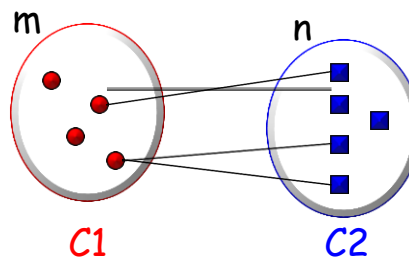
$$\text{max-card}(C2, A) = 1$$



Uno a molti (One-to-many)

$$\text{max-card}(C1, A) = n$$

$$\text{max-card}(C2, A) = 1$$

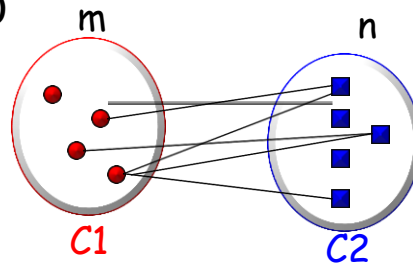


Vincoli di cardinalità massima(2)

Molti a molti (Many-to-many)

$$\text{max-card}(C1,A) = n$$

$$\text{max-card}(C2,A) = m$$



$\text{max-card}(C1,A) = n$ significa che il massimo numero di partecipazioni di un elemento di $C1$ in A è limitato soltanto dal numero di elementi in $C2$

$\text{max-card}(C2,A) = m$ significa che il massimo numero di partecipazioni di un elemento di $C2$ in A è limitato soltanto dal numero di elementi in $C1$

Vincoli di cardinalità massima(3)

- Il numero massimo totale delle corrispondenze nell'associazione tra $C1$ (con m istanze) e $C2$ (con n istanze) è dunque:

✚ Uno a uno (One-to-one) : $\min(m,n)$

✚ Uno a molti (One-to-many) : n

✚ Molti a molti (Many-to-many) : $n \times m$



Aggregazione n-aria

- Quanto visto si estende naturalmente al caso di n classi.



- ✚ *Ogni corso ha non più di 3 lezioni settimanali*
- ✚ $\text{min-card}(\text{Corsi}, \text{Lezioni}) = 1; \text{max-card}(\text{Corsi}, \text{Lezioni}) = 3$.
- ✚ Ragionare sui vincoli è in generale meno immediato che nel caso binario.
- ✚ *Ogni corso si tiene in non più di 2 aule*
- ✚ Non riguarda i giorni, quindi non riguarda Lezioni!



Tassonomia dei metodi di analisi

- analisi orientata alle **funzioni**
- analisi orientata agli **oggetti**
- analisi orientata agli **stati**

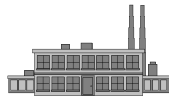


- ✚ L'orientamento di un metodo verso un aspetto è determinato dall'approccio seguito durante l'analisi e dal diverso accento riposto nella modellazione della realtà.
- ✚ La tendenza attuale consiste nell'integrare modelli di rappresentazione nati per finalità diverse; si veda ad esempio il linguaggio di modellazione **UML** (**U**nified **M**odeling **L**anguage), che si è affermato recentemente come *standard de facto*.



Analisi orientata alle funzioni

- L'obiettivo è rappresentare un sistema come:
 - ✚ una rete di processi;
 - ✚ un insieme di flussi informativi tra processi.
- ✚ Ciò corrisponde alla progressiva costruzione di una **gerarchia funzionale**.
- ✚ Sinonimi di funzione:
 - ✚ processo, bolla, attività, trasformazione, transazione



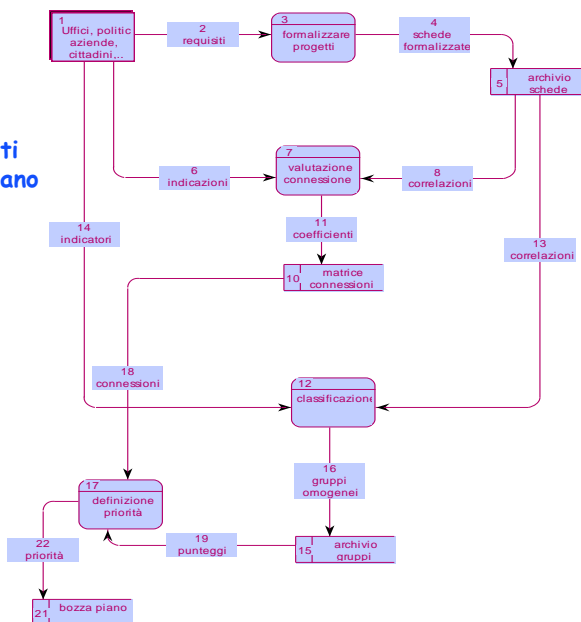
Progettazione di basi di dati

21



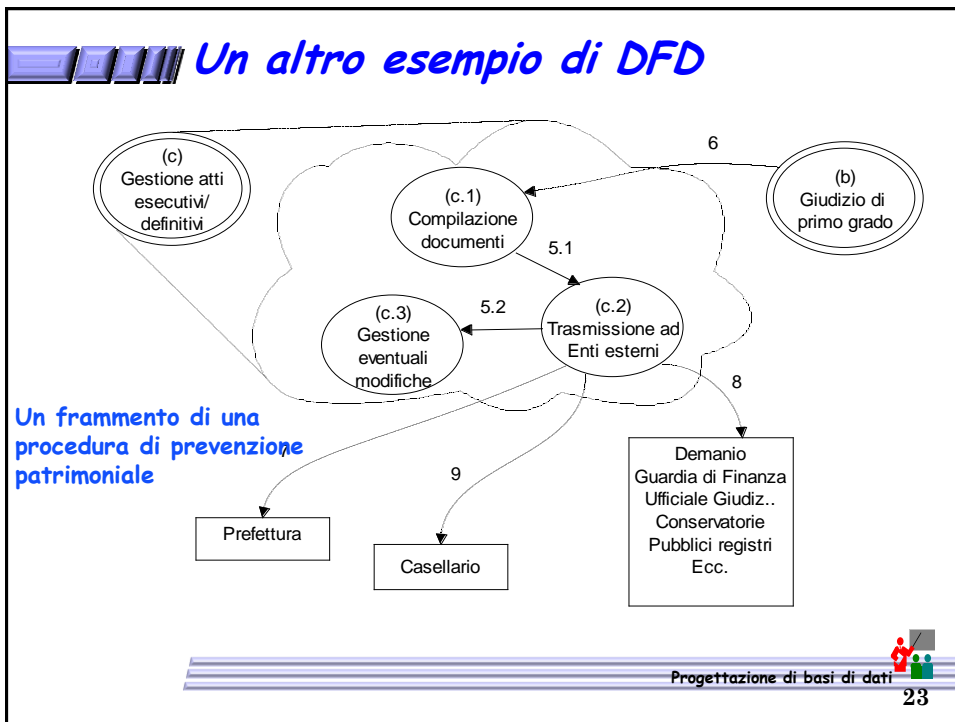
DFD

Un DFD
(Data Flow Diagram)
per la scelta di progetti
relativi al traffico urbano



Progettazione di basi di dati

22



Analisi orientata agli oggetti (1)

- L'enfasi è posta:
 - ✦ sull'identificazione degli oggetti;
 - ✦ sulle interrelazioni tra oggetti.

- Nel tempo e le proprietà strutturali degli oggetti osservati restano abbastanza stabili, mentre l'uso che degli oggetti si fa può mutare in modo sensibile.

Progettazione di basi di dati

24

Analisi orientata agli oggetti (2)

- In letteratura sono classificati come metodi di analisi a oggetti approcci di natura molto diversa:
- ✚ formalismo **Entity-Relationship** arricchito;
- ✚ formalismi **a oggetti**, dove la nozione di oggetto modella sia aspetti strutturali sia aspetti comportamentali;
- ✚ **tecnica JSD** (Jackson System Development) in cui il concetto di entità è più restrittivo rispetto alla nozione di oggetto e per certi versi più ricco rispetto al concetto di entità del modello E/R.

....

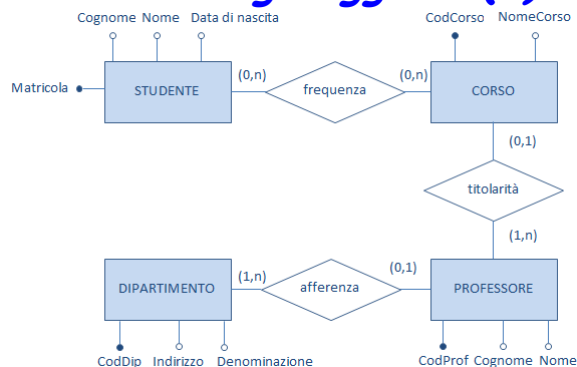
Progettazione di basi di dati

25

Analisi orientata agli oggetti (3)

uno schema
Entity-Relationship

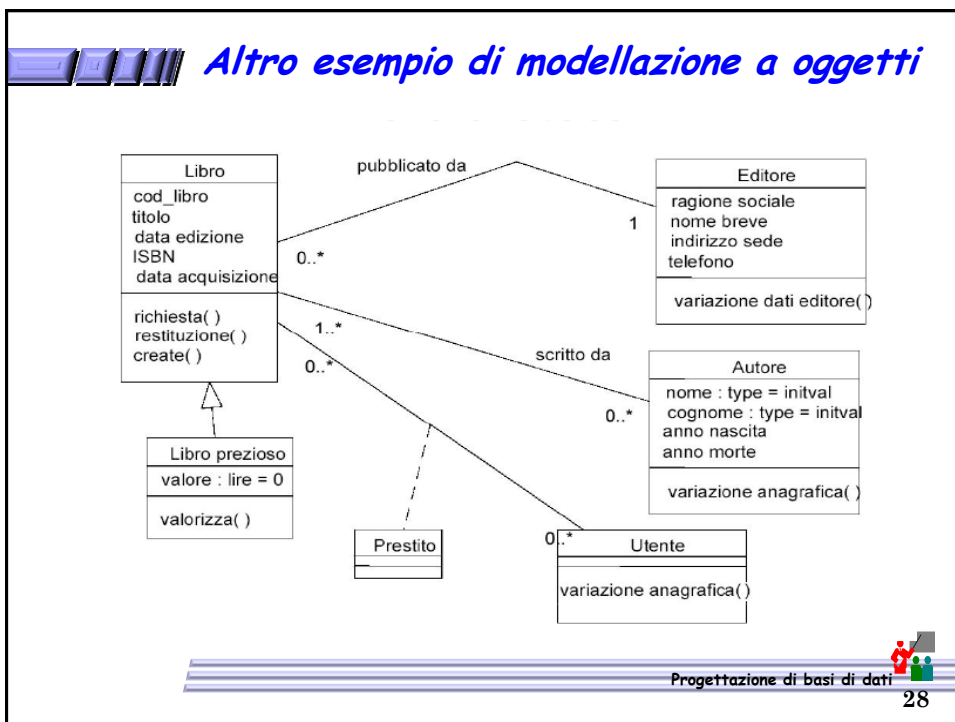
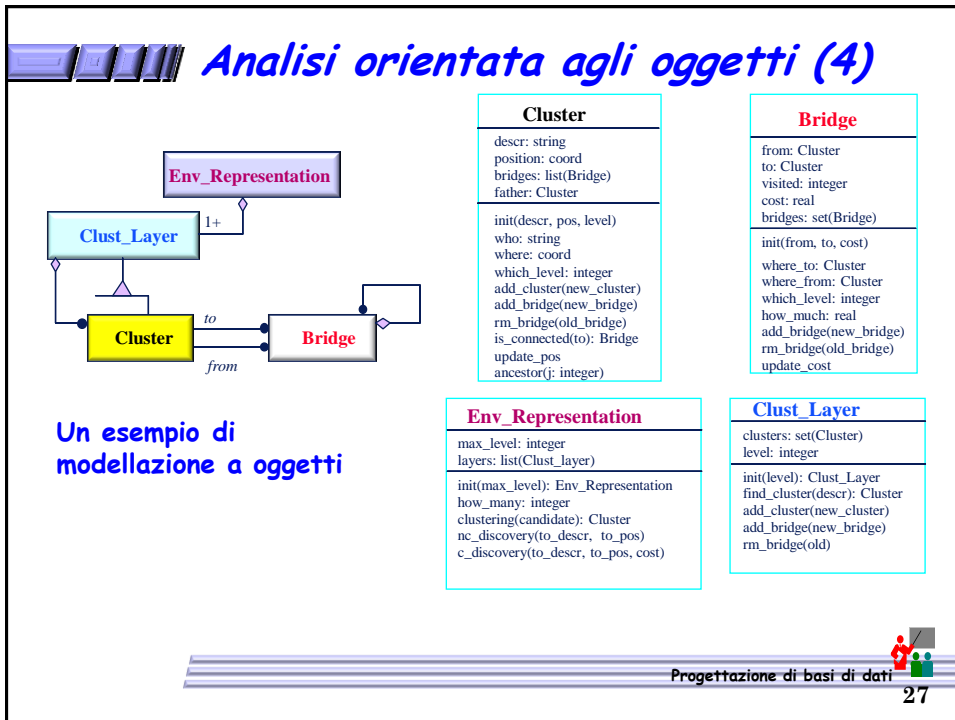
N.B. Snapshot di un A.A

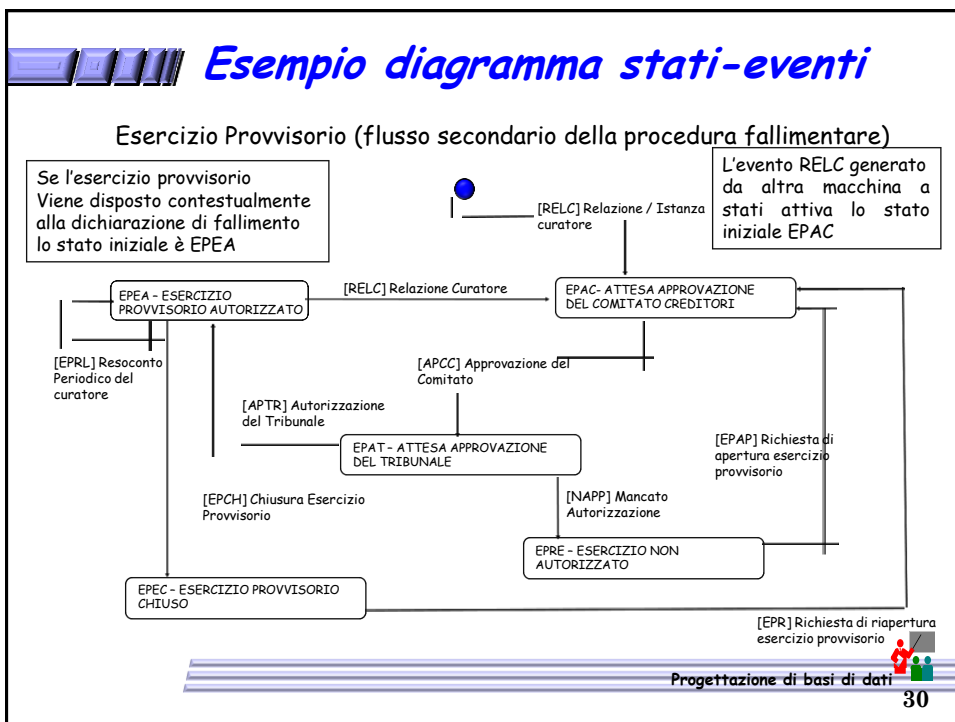
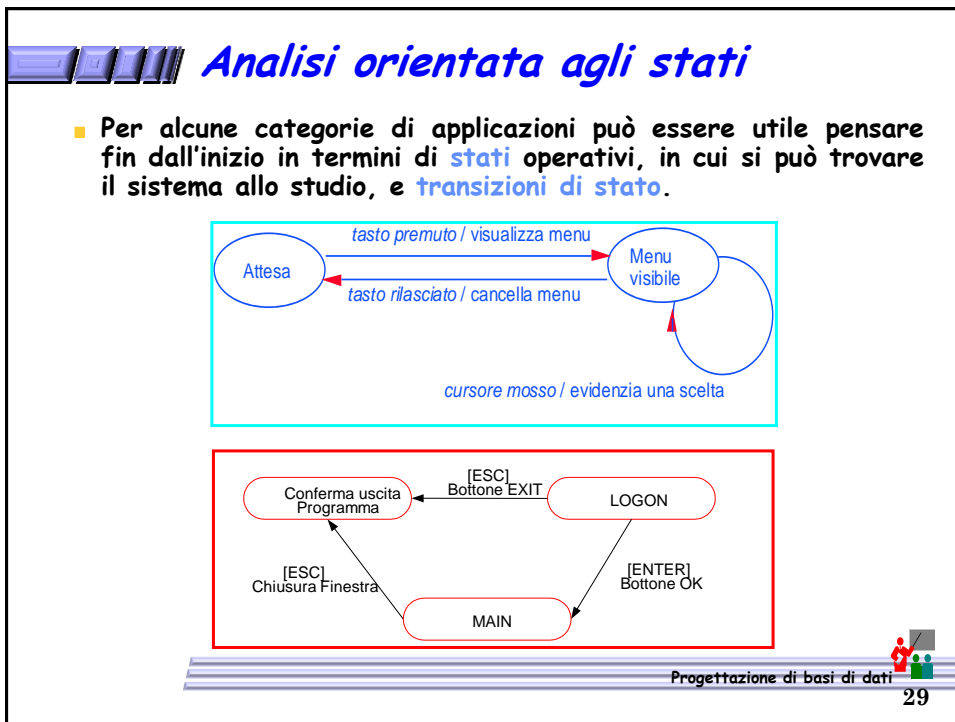


- ✚ Uso di un dizionario dati, con finalità di memorizzare in modo uniforme i vari tipi di dati riscontrati durante l'analisi e le relazioni che sussistono fra essi e le altre risorse coinvolte nel sistema informativo.
- ✚ Impiego successivo o concomitante di altri strumenti di modellazione. Ad esempio integrazione schemi E/R con DFD.

Progettazione di basi di dati

26





Tipologia delle applicazioni

applicazioni orientate agli oggetti

- l'aspetto più significativo è costituito dalle informazioni, le funzioni svolte non sono molto complesse.

applicazioni orientate alle funzioni

- la complessità risiede nel tipo di trasformazione input-output operata.

applicazioni orientate al controllo

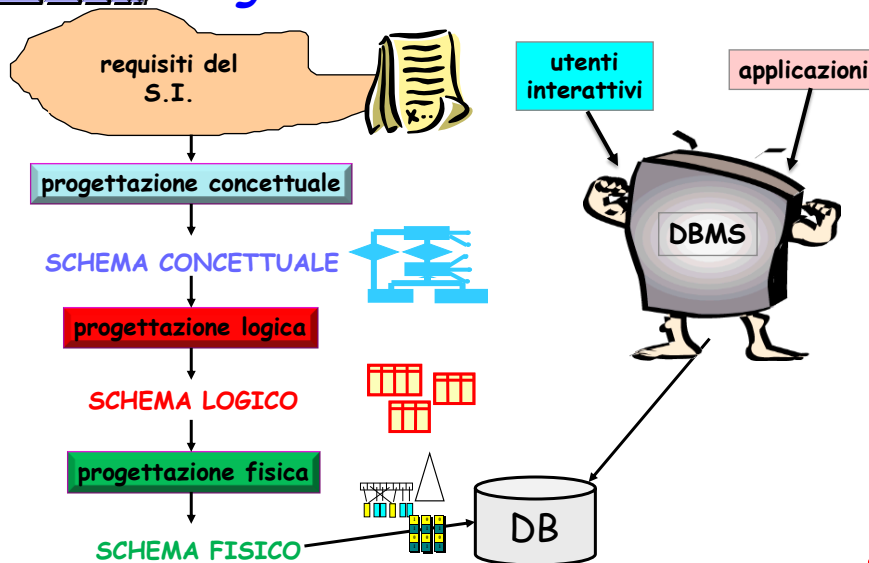
- l'aspetto più significativo da modellare è la sincronizzazione fra diverse attività cooperanti nel sistema.

- Un tipico S.I. di media complessità richiede durante la progettazione il ricorso a più strumenti di modellazione. **Il linguaggio di modellazione UML, complementato con metodi di progettazione di DB, risponde a questa esigenza.**

Progettazione di basi di dati

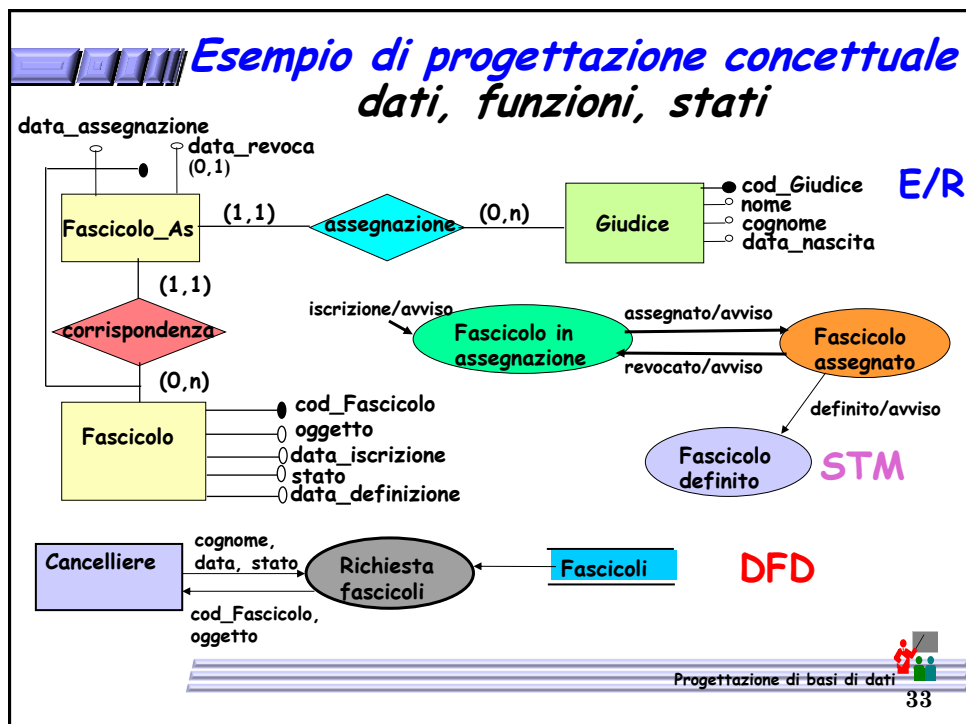
31

Progettazione della base di dati



Progettazione di basi di dati

32



Le fasi di progettazione di un DB

Progettazione logica (relazionale)

✚ **Obiettivo:** descrivere la base dati secondo il **modello logico** del DBMS

<u>cod Giudice</u>	cognome	nome	data_nascita
BNCGR668L21A944K	BIANCHI	GIORGIO	21-07-1968	
RSSNNA52A62A944V	ROSSI	ANNA	22-01-1952	
VRDMRC59M20F839X	VERDI	MARCO	20-08-1959	
.....				

GIUDICI

<u>cod_Fascicolo</u>	oggetto	data_isc	Stato	data_def
.....	
F023A12	O10	12-12-2001	AS		
F023A13	O05	13-12-2002	NA		
.....					

FASCICOLI

<u>Cod_Fascicolo</u>	<u>data_ass</u>	<u>data_rev</u>	<u>cod_Giudice</u>
F023A12	21-12-2001	null	RSSNNA52A62A944V	
F023A13	20-01-2002	20-05-2002	VRDMRC59M20F839X	
F023A13	02-06-2002	null	BNCGR669L21A944K	
.....	

ASSEGNAZIONI

Progettazione di basi di dati

34



Le fasi di progettazione di un DB

Progettazione fisica

- ✚ **Obiettivo:** scelta delle strutture di memorizzazione e degli indici (in parte dipendente dal particolare RDBMS)

Esempio per la tabella **FASCICOLI**:

- ✚ Indice su **cod_Fascicolo.FASCICOLI** (chiave primaria) clustered
- ✚ Indice su **stato.FASCICOLI** (chiave secondaria) unclustered
- ✚ Indice su **data_isc.FASCICOLI** (chiave secondaria) unclustered
- ✚

Tipicamente gli indici sono utili per i controlli di integrità referenziale e per migliorare l'efficienza delle query

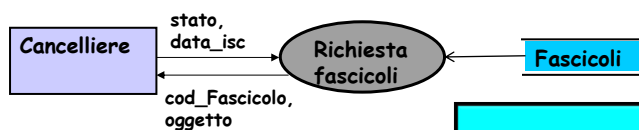
Progettazione di basi di dati

35



Progettazione di un'applicazione DB

- ✚ **Obiettivo:** realizzazione moduli software per le funzioni e per l'interfaccia utente



```
select cod_Fascicolo, oggetto
from FASCICOLI
where stato='NA'
and data_isc >= '01-01-2006'
```

stato	<input type="text" value="NA"/>
data_isc	<input type="text" value="01-01-2006"/> <input "="" type="button" value=">="/>

FASCICOLO	OGGETTO
F053F11	O09
F053H01	O11
F053G01	O06

Progettazione di basi di dati

36



Progettazione guidata dai dati

- Due aspetti di primaria importanza nella progettazione di un sistema informativo:
 - ✦ progettazione della base dati;
 - ✦ progettazione delle applicazioni.
- Il ruolo primario viene svolto dai dati, in quanto:
 - ✦ sono (strutturalmente) più stabili;
 - ✦ sono condivisi da più applicazioni.
- È quindi opportuno che la progettazione sia guidata focalizzando **innanzitutto l'attenzione sulla base di dati** e successivamente sulle applicazioni.



Metodi di progettazione

- Per progettare una base di dati (ma non solo) di buona qualità è opportuno adottare un metodo di progettazione in grado di:
 - ✦ definire le fasi in cui l'attività di progettazione si articola,
 - ✦ fornire criteri per scegliere tra diverse alternative,
 - ✦ rendere disponibili al progettista adeguati modelli di rappresentazione,
- e che sia di applicabilità generale e di facile uso.



Riepilogo: fasi di progettazione

- Il metodo introdotto prevede 3 fasi:
 - + progettazione concettuale
 - + progettazione logica
 - + progettazione fisica
- La fase di raccolta e analisi dei requisiti in pratica viene ad essere svolta congiuntamente a quella di progettazione concettuale.
- Ognuna delle fasi si basa su un **modello**, che permette di generare una rappresentazione formale (**schema**) della base di dati a un certo livello di astrazione (rispettivamente concettuale, logico e fisico):
 - + Schema concettuale
 - + Schema logico
 - + Schema fisico



Fase di raccolta e analisi dei requisiti

- Rappresenta la fase in cui sono raccolte e analizzate le specifiche informali ed eterogenee che i vari utenti forniscono circa le procedure da automatizzare mediante un DBMS:
 - + requisiti informativi: caratteristiche dei dati;
 - + requisiti sui processi: operazioni sui dati;
 - + requisiti sulla dinamica: evoluzione nel tempo;
 - + requisiti sui vincoli di integrità: proprietà dei dati e delle operazioni.
- Attività principali:
 - + costruzione di un glossario dei termini;
 - + eliminazione delle ambiguità (sinonimi, omonimi);
 - + raggruppamento dei requisiti "omogenei".
- Fase solo apparentemente semplice, nella realtà è spesso la più complessa poiché è difficilmente standardizzabile il processo che porta a
 - + **comprendere che cosa vogliono in realtà gli utenti!**



Fase di progettazione concettuale

- A partire dai requisiti informativi viene creato uno **schema concettuale**, cioè una descrizione formalizzata e integrata delle esigenze aziendali, espressa in modo **indipendente dal DBMS** adottato.
- A tale scopo si adotta un **modello concettuale**, che permette di fornire descrizioni ad alto livello indipendenti dall'implementazione.
- Lo schema concettuale è indipendente anche dal **tipo** di DBMS che sarà utilizzato (relazionale, gerarchico, ...).



Fase di progettazione logica

- ✚ Consiste nella traduzione dello schema concettuale nel modello dei dati del DBMS.
- ✚ Il risultato è uno **schema logico**, espresso nel DDL del DBMS.
- ✚ In questa fase si considerano anche aspetti legati a:
 - ✚ **integrità e consistenza (vincoli);**
 - ✚ **efficienza.**
- ✚ La progettazione logica si articola in due sotto-fasi:
 - ✚ **ristrutturazione dello schema concettuale;**
 - ✚ **traduzione verso il modello logico.**



Fase di progettazione fisica

- + In questa ultima fase si operano scelte spesso strettamente dipendenti dallo specifico DBMS utilizzato
- + Ad esempio, lo stesso schema logico può essere fisicamente rappresentato in modo diverso in DB2 e in Oracle, al fine di sfruttare al meglio le caratteristiche del particolare DBMS adottato.
- + Il risultato è lo **schema fisico**, che descrive le strutture di memorizzazione e accesso ai dati (tablespace, clustering, indici, ecc.).



Modelli dei dati: logici vs concettuali

- Un modello dei dati è una collezione di concetti che sono utilizzati per descrivere i dati, le loro associazioni, e i vincoli che questi devono rispettare.
 - + Un ruolo di primaria importanza nella definizione di un modello dei dati è svolto dai **meccanismi che possono essere usati per strutturare i dati** (cfr. i costruttori di tipo in un linguaggio di programmazione)
- Modelli logici:** utilizzati nei DBMS per l'organizzazione dei dati;
- utilizzati dai programmi - indipendenti dalle strutture fisiche.
- Modelli concettuali:** permettono di rappresentare i dati in modo indipendente da ogni particolare sistema;
- cercano di descrivere i concetti del mondo reale;
 - sono utilizzati nelle fasi preliminari di progettazione.
- Il più noto in ambito DB è il modello **Entity-Relationship**.

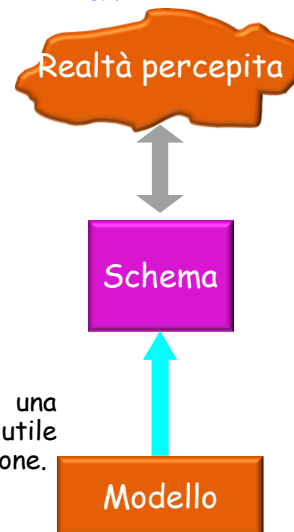
Modelli concettuali... un po' di storia

■ Nel tempo sono stati proposti numerosi modelli concettuali per la progettazione di basi di dati:

- ✦ modelli semantici, RM/T, ... [inizio anni '70]
- ✦ Entity-Relationship (E/R) (entità-associazione) [Chen 1976]
- ✦ IDEF1X (standard adottato dagli uffici governativi USA)
- ✦ UML (Universal Modelling Language) [1999]
- ✦

I modelli concettuali dei dati

- ✦ Lo scopo è pervenire a uno schema che rappresenti la realtà di interesse in modo indipendente dal DBMS.
- ✦ Si cerca un livello di **astrazione** "intermedio" tra sistema e utenti, che sia al tempo stesso:
 - ✦ Flessibile
 - ✦ Intuitivo
 - ✦ Espressivo
- ✦ caratteristiche che mancano ai modelli logici.
- ✦ I **modelli concettuali** prevedono tipicamente una rappresentazione grafica, che risulta anche utile come strumento di documentazione e comunicazione.





Sommario

- ❖ La progettazione di un sistema informativo è guidata dai dati, e si avvale di una metodologia che consta di diverse fasi.
- ❖ Ogni fase produce uno schema, facendo uso di uno specifico modello.
- ❖ Per la progettazione concettuale si fa uso di un modello concettuale che, astruendo da aspetti specifici dei DBMS, rappresenta un valido compromesso tra ciò che si dovrà realizzare e la realtà che si deve modellare.
- ❖ Tutti i modelli concettuali si basano su alcuni meccanismi di astrazione fondamentali: classificazione, aggregazione, generalizzazione, proiezione.

