



**UNIVERSITÀ
DEL SALENTO**

Infomatica

**Corso di Laurea in Scienze Biologiche
Dip.to di Scienze Tecnologie Biologiche ed Ambientali
(cfu 6)**

Prof. Salvatore Mancarella

salvatore.mancarella@unisalento.it

Basi di dati



Base di dati

- Con **base di dati** (o **banca dati**, a volte abbreviato con la sigla **DB** dall'inglese *database*) in informatica si indica un insieme di dati strutturati ovvero omogeneo per contenuti e formato, memorizzati in un computer,
- rappresentando di fatto la versione digitale di un archivio dati o schedario.



Storia

- All'inizio della storia dell'informatica, la grande maggioranza dei programmi specializzati consentivano l'accesso a una singola base di dati, per guadagnare in velocità di esecuzione pur perdendo in flessibilità.
- Oggi, invece, i moderni sistemi possono essere utilizzati per compiere operazioni su un gran numero di basi di dati differenti.
- Dagli anni settanta le basi di dati hanno subito un enorme sviluppo sia in fatto di quantità di dati memorizzati sia in fatto di tipi di architetture adottate.
 - Architettura centralizzata, **DBMS**;
 - Architettura **data Warehouse**.



Definizione di un DBMS

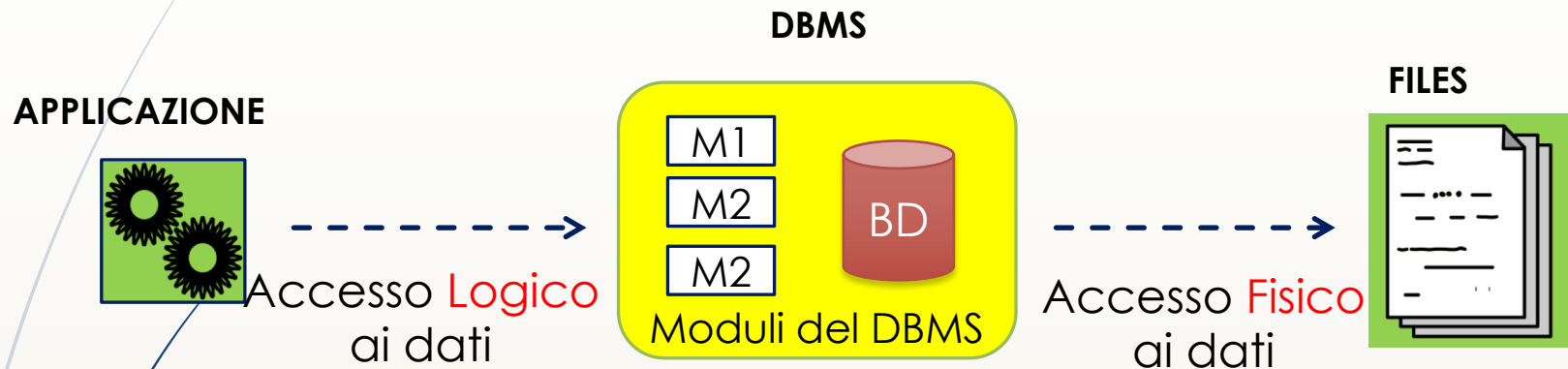
Un **DBMS** è un sistema software che è in grado di gestire collezioni di dati *grandi, condivise e persistenti, in maniera efficiente e sicura.*

Alcune funzionalità

- Creazione di una base di dati e memorizzazione su memoria secondaria
- Accesso in lettura/scrittura ai dati
- Condivisione di dati tra diversi utenti/applicazioni
- Protezione dei dati da accessi non autorizzati
- Reliability dei dati in caso di guasti (hardware/software)

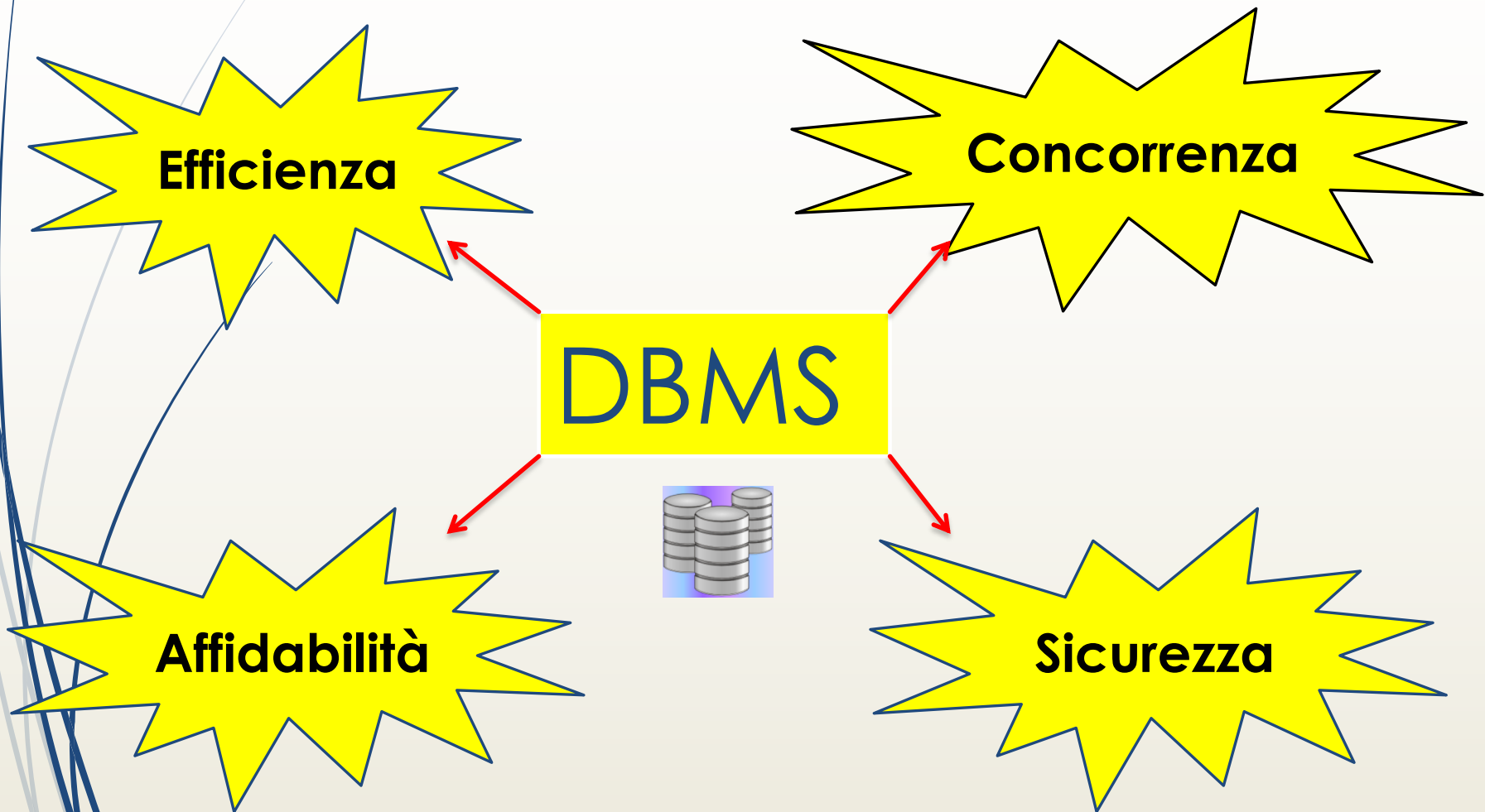


Separazione Dati/Applicazioni con DBMS

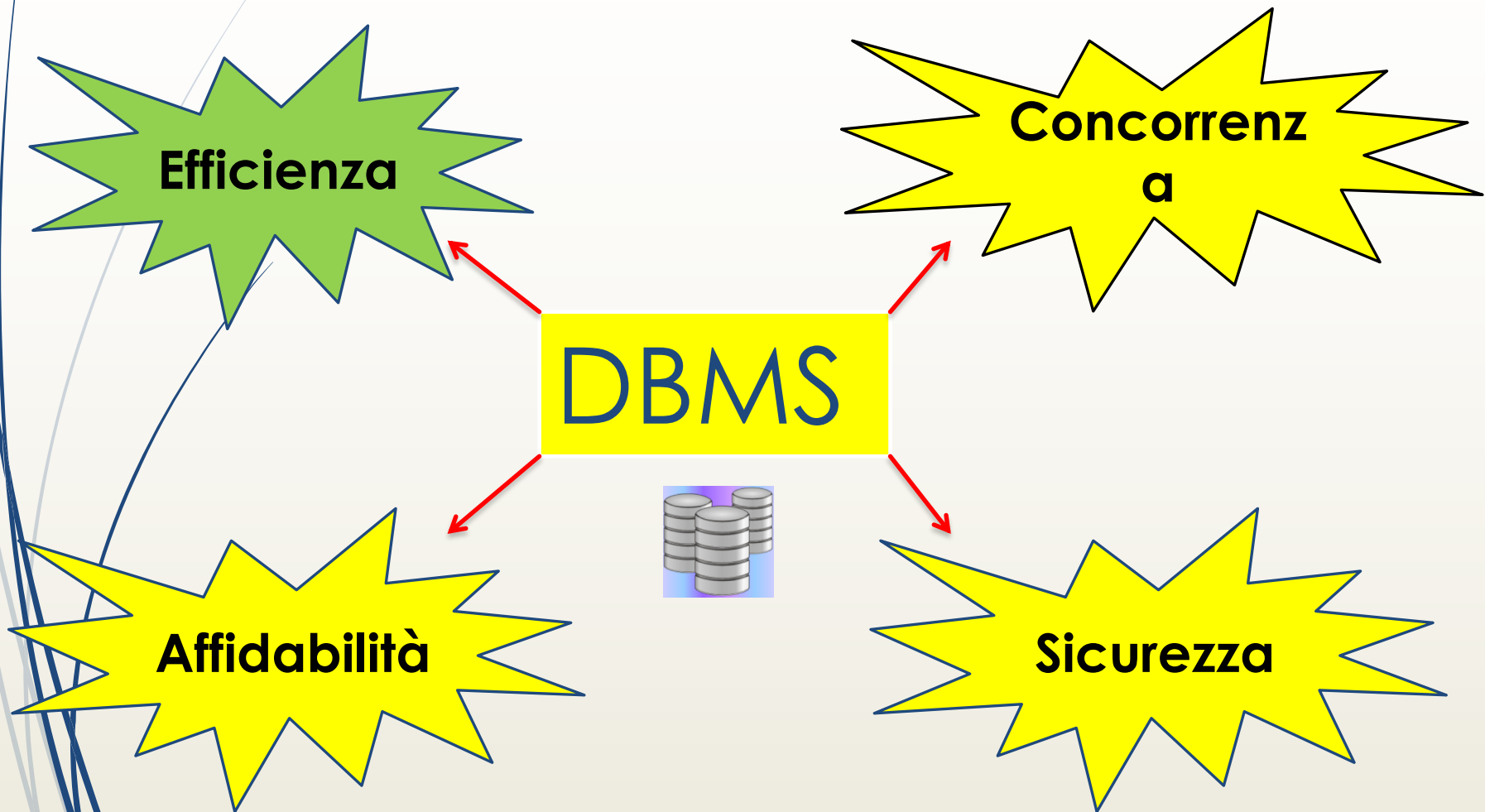


- Tramite i DBMS, è possibile implementare un **paradigma di separazione di dati ed applicazioni ...**
- Le applicazioni non necessitano di conoscere la struttura fisica dei dati (es. come e dove sono memorizzati su disco) ma **solo la struttura logica** (cosa rappresentano).

Le caratteristiche di un DBMS



Le caratteristiche di un DBMS



Caratteristiche di un DBMS:

Efficienza

- **Efficienza** di un DBMS nella gestione dei dati :
- DBMS forniscono adeguate **strutture dati** per organizzare i dati all'interno dei file, e per supportare le operazioni di ricerca/aggiornamento.
- In genere, parliamo di **strutture dati ad albero** o **tabelle hash**.

The diagram shows a table on the left and a list of records on the right. Arrows indicate the mapping between the table rows and the records. The table has columns for names and values. The records are numbered and contain names. Some records are highlighted in green or orange to match the table rows.

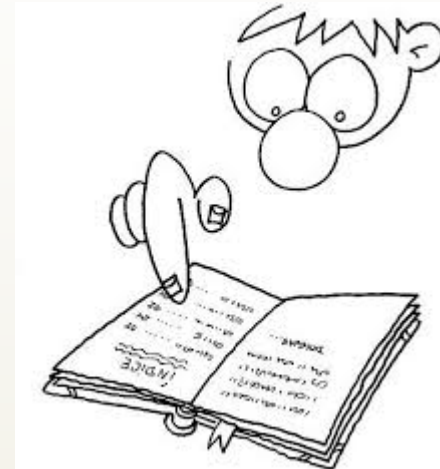
Aceto	
Aldo	
Asola	
Baco	
...	
...	

10021	Abate	
14322	Abete	
00002	Acaro	
03421	Aceto	
00003	Adone	
20000	Africa	
65001	Ago	
76199	Aldo	
00001	Amari	
40000	Amato	
54002	Ando	
00004	Asola	
...	...	
34001	Baba	
54200	Bacardi	
65401	Bacchi	
54320	Baco	
...	...	

Caratteristiche di un DBMS: Efficienza

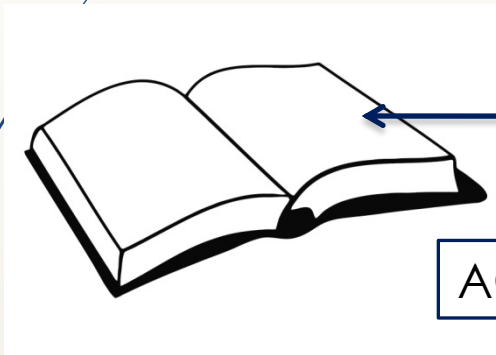
- **Indice** → struttura che contiene informazioni sulla posizione di memorizzazione delle tuple sulla base del valore del campo chiave.

- A che serve un indice?



Caratteristiche di un DBMS: Efficienza (INDICE)

➤ ESEMPIO

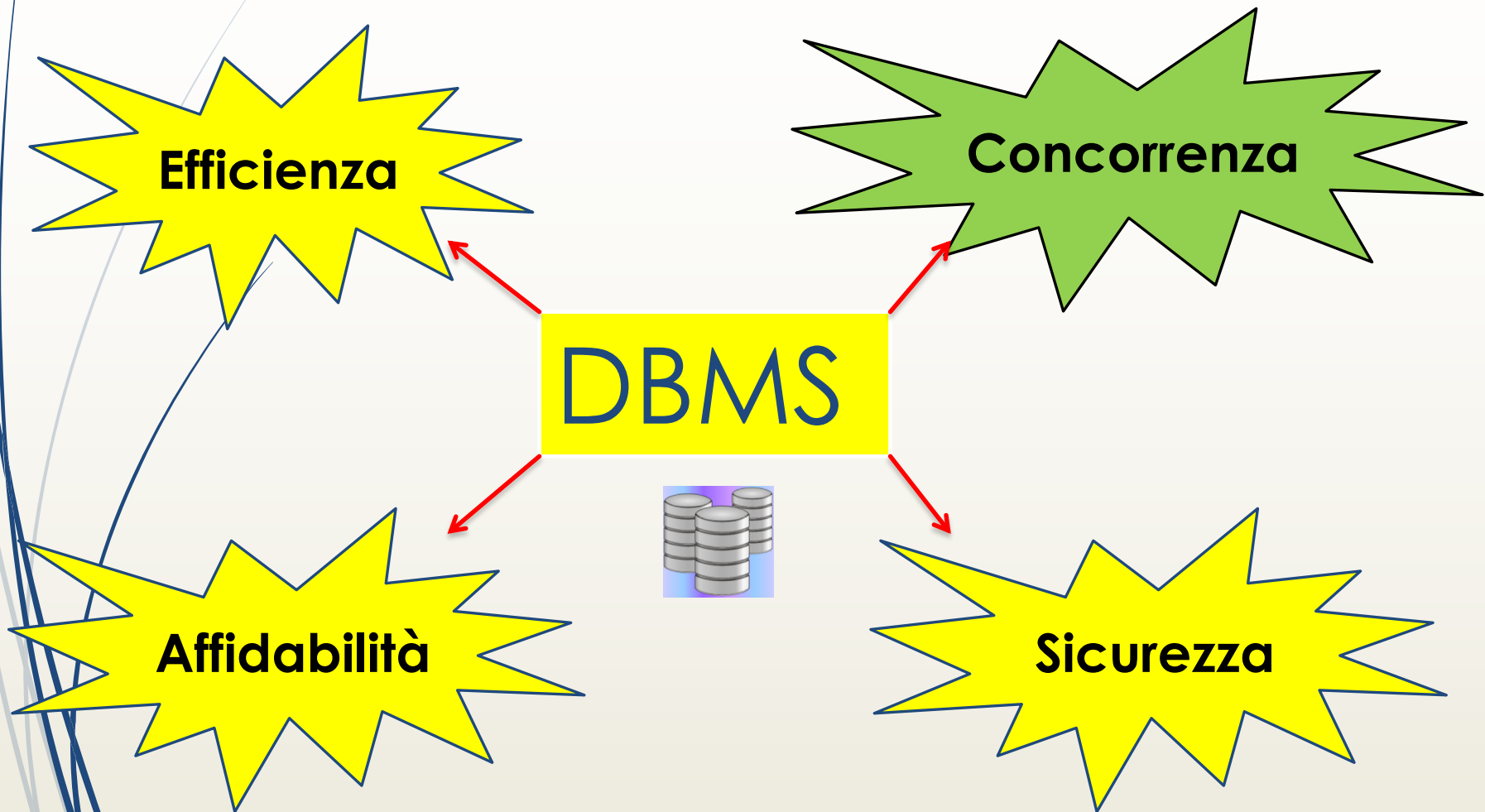


ACCESSO DIRETTO

Indice

Introduzione	1
Capitolo 1	20
Capitolo 2	40
Capitolo 3	60
Conclusioni	65

Le caratteristiche di un DBMS



Caratteristiche di un DBMS: Gestione della Concorrenza

- In molti sistemi è fondamentale **gestire operazioni concorrenti** di accesso ai dati



Caratteristiche di un DBMS: Gestione della Concorrenza

➤ ESEMPIO



Processing di più di 7.7 milioni di pagamenti ogni giorno

La maggior parte dei DBMS forniscono un **livello di granularità di locking** più fine di quello convenzionale (a livello di tabella, pagina, o singola entry).

Caratteristiche di un DBMS:

Gestione della Concorrenza

- Al tempo stesso, un DBMS deve garantire il fatto che **accessi da parte di applicazioni diverse non interferiscano tra loro**, lasciando il sistema in uno **stato inconsistente**.



Caratteristiche di un DBMS: Gestione della Concorrenza

Es. Sistema informativo dei conti bancari

- 2 richieste da gestire al tempo t :
 - Prelievo di 100 euro dal conto X
 - Prelievo di 80 euro dal conto X
- Saldo del conto X at tempo t : 120 euro

Caratteristiche di un DBMS: Gestione della Concorrenza

ESEMPIO di ESECUZIONE

(non corretta!!)

OP1

Leggi X
Calcola X-100
Scrivi X

OP2

Leggi X
Calcola X-80
Scrivi X

Schedule:

OP1: Leggi X
OP2: Leggi X
OP1: Calcola X-100
OP2: Calcola X-80
OP1: Scrivi X
OP2: Scrivi X

Valore X:

120

120

120

120

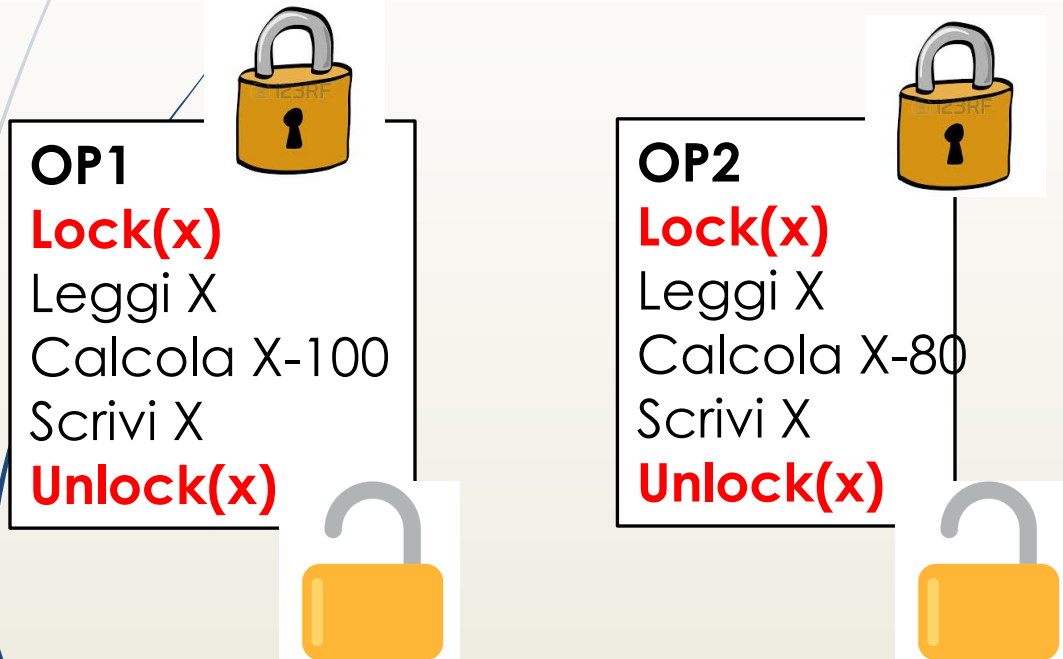
20

40 (????)

Per prevenire tali situazioni, i DBMS implementano **algoritmi di controllo della concorrenza** che operazioni sui dati (**transazioni**) eseguite in concorrenza producano lo stesso risultato di un'esecuzione seriale.

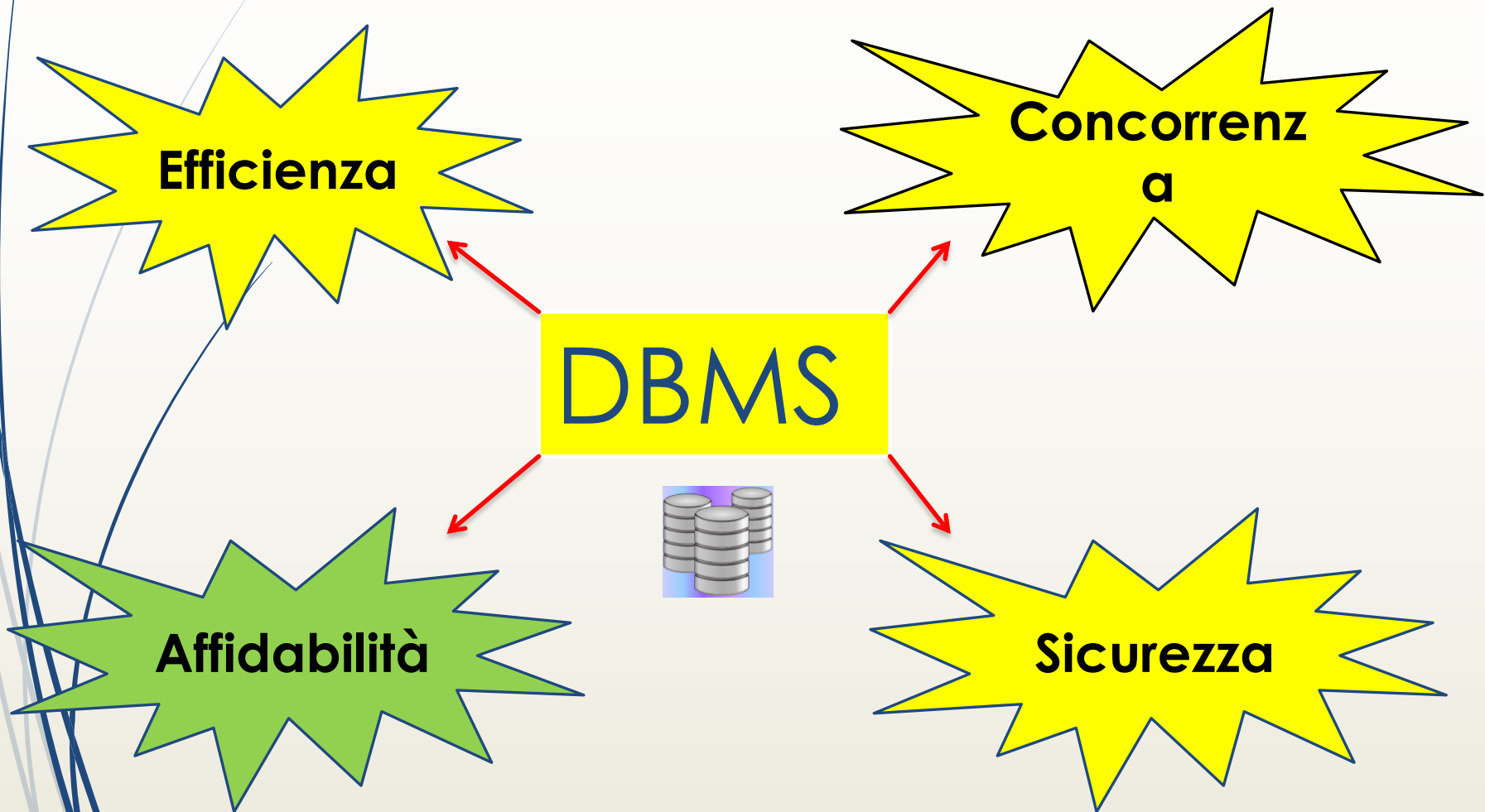
Caratteristiche di un DBMS: Gestione della Concorrenza

Lock Manager → componente del DBMS responsabile di **gestire i lock alle risorse del DB**, e di **rispondere alle richieste delle transazioni**.



➤ Utilizzo di **lock** in lettura/scrittura per accesso a risorse condivise (dati).

Le caratteristiche di un DBMS



Caratteristiche di un DBMS:

Affidabilità

Meccanismi di **Roll-back**

Esempio:

Alcune operazioni sui dati sono **particolarmente delicate**, e devono essere gestite in maniera opportuna, secondo la regola del **tutto o niente**.

Es. Trasferimento di denaro (100\$) dal conto X al conto Y.

Op1: $X = X - 100$

Op2: $Y = Y + 100$

CRASH!



Caratteristiche di un DBMS:

Affidabilità

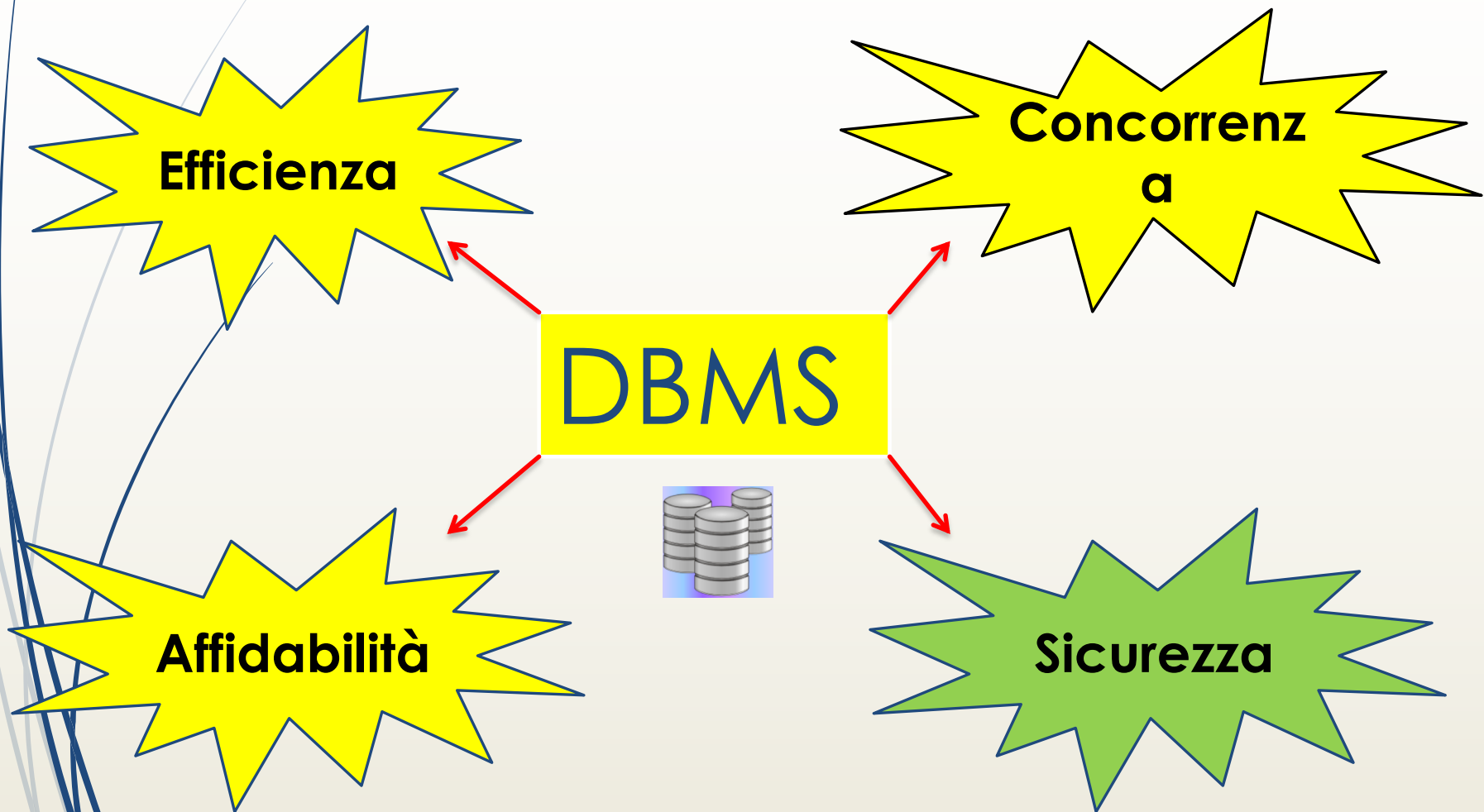
Meccanismi di Roll-back

➤ Alcune operazioni sui dati sono **particolarmente delicate**, e devono essere gestite in maniera opportuna, secondo la regola del **tutto o niente**.



➤ Per questo, i DBMS devono fornire appositi strumenti per **annullare operazioni non completate** e fare **roll-back** dello stato del sistema ...

Le caratteristiche di un DBMS



Caratteristiche di un DBMS:

Multi-utenza e Sicurezza

La maggior parte dei DBMS implementa **politiche di controllo degli accessi** ad i dati mediante **sistemi di permessi**:

- Quali **operazioni sono consentite** all'utente X?
- Quali **dati appartengono** all'utente X?

UTENTE X



UTENTE Y

Utente	OPERAZIONE	DATO	PERMESSO
Utente X	Lettura	Stipendio di X	Consentito
Utente X	Lettura	Stipendio di Y	Consentito
Utente Y	Scrittura	Stipendio di Y	Negato

Architettura a tre livelli

- In pratica, un DBMS può essere visto come un' **architettura software** a 3 livelli

