

# I DBMS sono ... efficienti

- Cercano di utilizzare al meglio le risorse di spazio di memoria (principale e secondaria) e tempo (**di esecuzione e di risposta**)
- Tecniche specializzate per l'implementazioni dei DBMS con investimenti e competizione

# Perché (non) utilizzare MS Access?

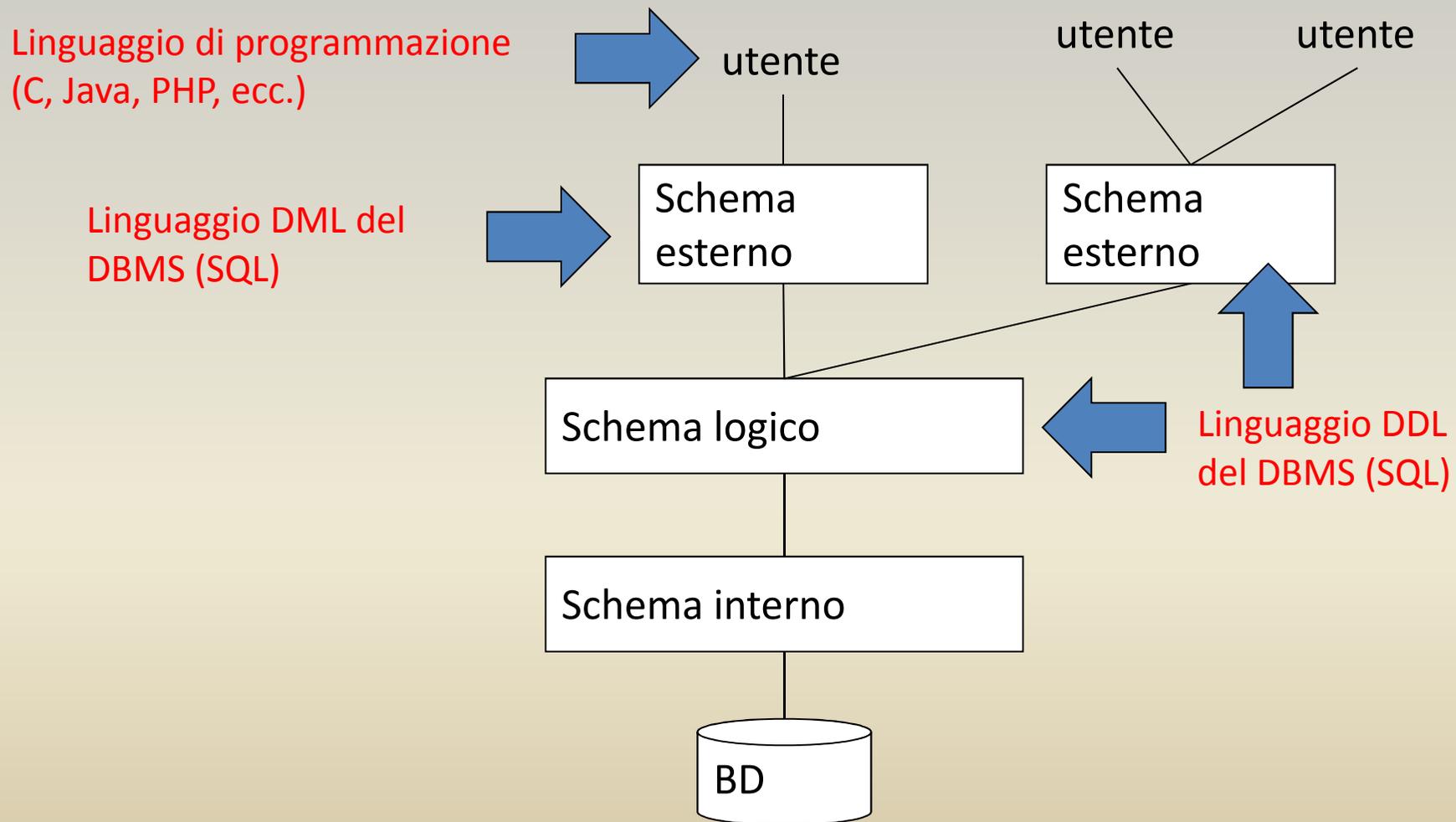
## **Vantaggi:**

- potete creare facilmente il database utilizzando MS Office dal vostro pc, disponendo di una interfaccia facile e intuitiva
- oltre a dover configurare i permessi di scrittura sulla cartella che ospita il db e sul db stesso non avete bisogno di altro
- potete effettuare un back up del database quando desiderate semplicemente copiando il file .MDB

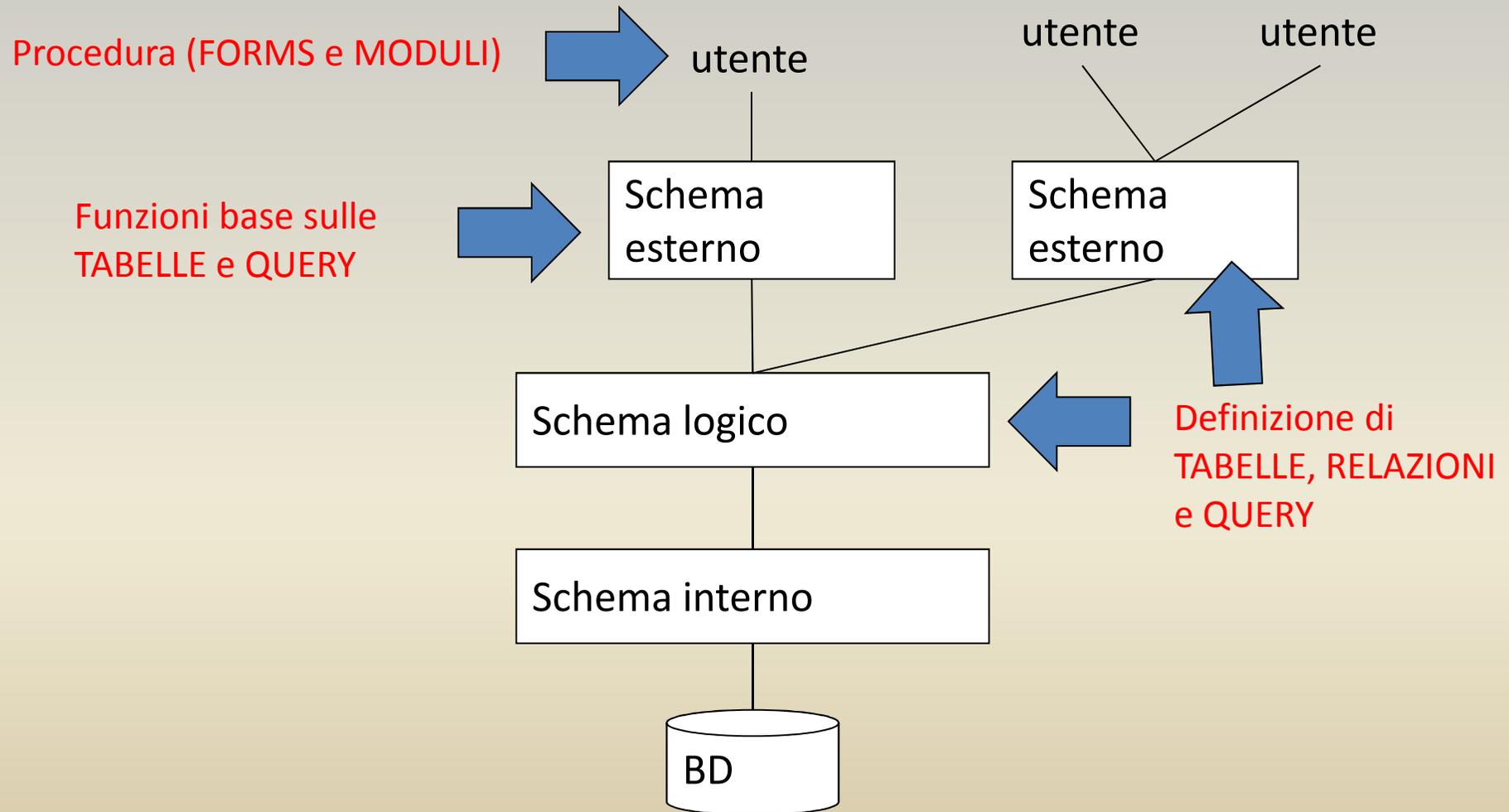
## **Svantaggi:**

- limitazione dichiarata da Microsoft sul numero di connessioni contemporanee al database, che è di 15 (teorico). In pratica con 3 siamo al limite.
- lentezza delle prestazioni in caso di “molti” record oppure multiutenza.

# Architettura standard (ANSI/SPARC) a tre livelli per DBMS

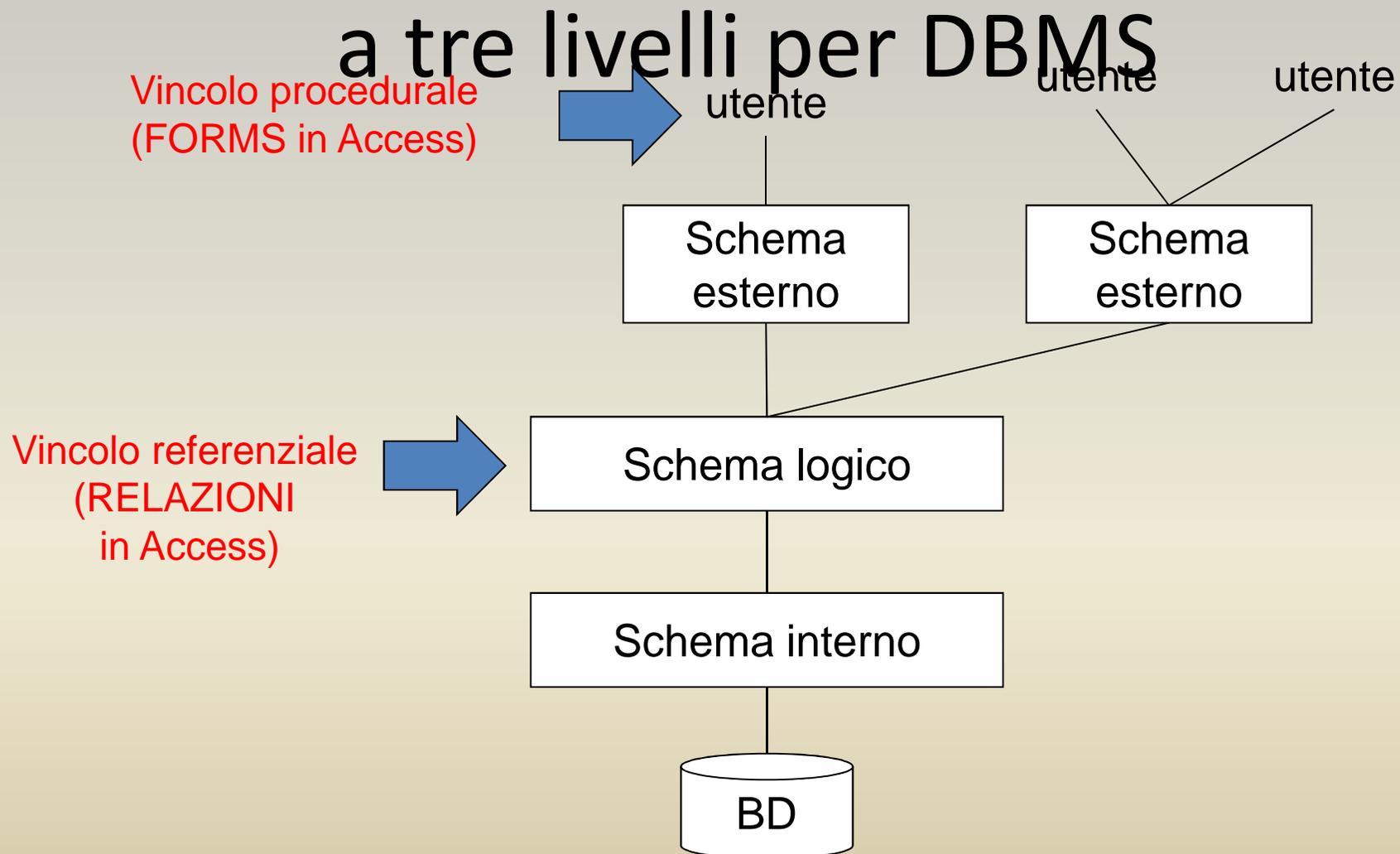


# Architettura standard (ANSI/SPARC) MS-ACCESS

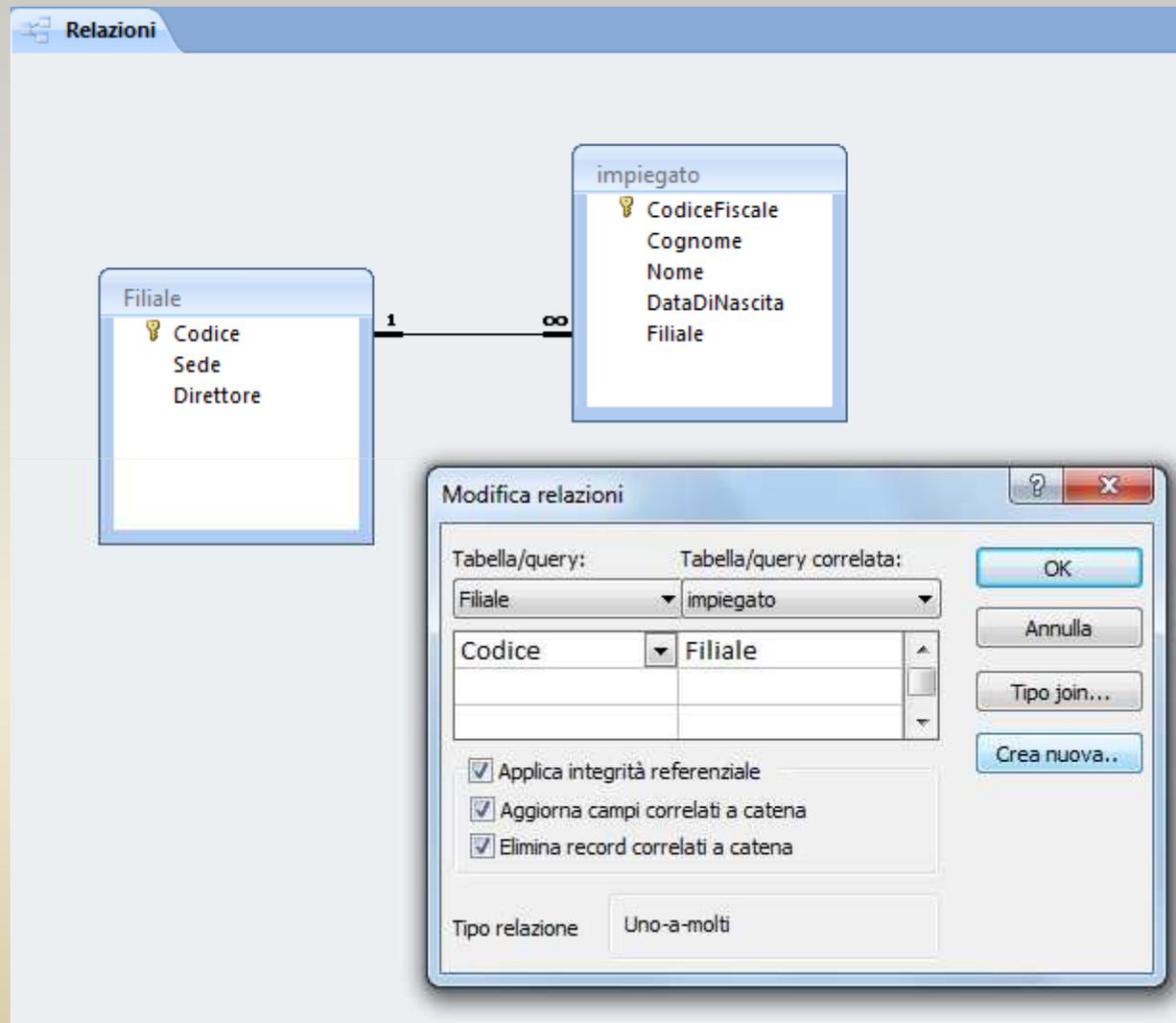


# Architettura standard (ANSI/SPARC)

## a tre livelli per DBMS



# Vincolo referenziale



# STRUTTURA TABELLA IMPIEGATO

Nome campo	Tipo dati
CodiceFiscale	Testo
Cognome	Testo
Nome	Testo
DataDiNascita	Data/ora
Filiale	Testo

Generale	Ricerca
Visualizza controllo	Casella combinata
Tipo origine riga	Tabella/query
Origine riga	SELECT Filiale.Codice, Filiale.Sede FROM Filiale;
Colonna associata	1
Numero colonne	2
Intestazioni colonne	No
Larghezza colonne	0cm;5cm
Righe in elenco	16
Larghezza elenco	Automatica
Solo in elenco	Sì
Consenti modifiche a ele	No
Maschera di modifica voc	



Vincolo pseudo-procedurale

Ripercussioni sul data entry ma non vincolo sui dati

# FORM IMPIEGATI



Filiale

Codice:

Sede:

Direttore:

impiegati:



impiegato

CodiceFiscale	Cognome	Nome	DataDiNascita	Filiale
SLVPTR54B578H	Salvi	Pietro	15/04/1954	Milano 2
RSSNDR60T524S	Rossi	Andrea	18/03/1960	Milano 2
				Milano 3
				Roma
				Los Angeles



Vincolo  
procedurale

# Stuttura FORM IMPIEGATI

impiegati

Intestazione maschera

impiegato

CodiceFiscale      Cognome      Nome      DataDiNascita      Filiale

Corpo

CodiceFiscale      Cognome      Nome      DataDiNascita      Filiale

Piè di pagina maschera

Finestra delle proprietà

Tipo di selezione: Casella combinata

Filiale

Formato    Dati    Evento    Altro    Tutte

Formato	
Posizioni decimali	Automatiche
Visibile	Sì
Numero colonne	2
Larghezza colonne	0cm;5cm

Finestra delle proprietà

Tipo di selezione: Casella combinata

Filiale

Formato    Dati    Evento    Altro    Tutte

Origine controllo	Filiale
Origine riga	SELECT Filiale.Codice, Filiale.Sede FROM Filiale;
Tipo origine riga	Tabella/query
Colonna associata	1

Vincolo procedurale

# Corso di Basi di Dati

- Descrivere in linguaggio naturale le informazioni organizzate nella base dati in figura.
- Individuare le chiavi e i vincoli di integrità referenziale che sussistono nella base di dati di figura e che è ragionevole assumere siano soddisfatti da tutte le basi di dati sullo stesso schema.
- Individuare anche gli attributi sui quali possa essere sensato ammettere valori nulli.

## PAZIENTI

Cod	Cognome	Nome
A102	Necchi	Luca
B372	Rossini	Piero
B543	Missoni	Nadia
B444	Missoni	Luigi
S555	Rossetti	Gino

## RICOVERI

Paziente	Inizio	Fine	Reparto
A102	2/05/94	9/05/94	A
A102	2/12/94	2/01/95	A
S555	5/10/94	3/12/94	B
B444	1/12/94	2/01/95	B
S555	5/10/94	1/11/94	A

## MEDICI

Matr	Cognome	Nome	Reparto
203	Neri	Piero	A
574	Bisi	Mario	B
431	Bargio	Sergio	B
530	Belli	Nicola	C
405	Mizzi	Nicola	A
201	Monti	Mario	A

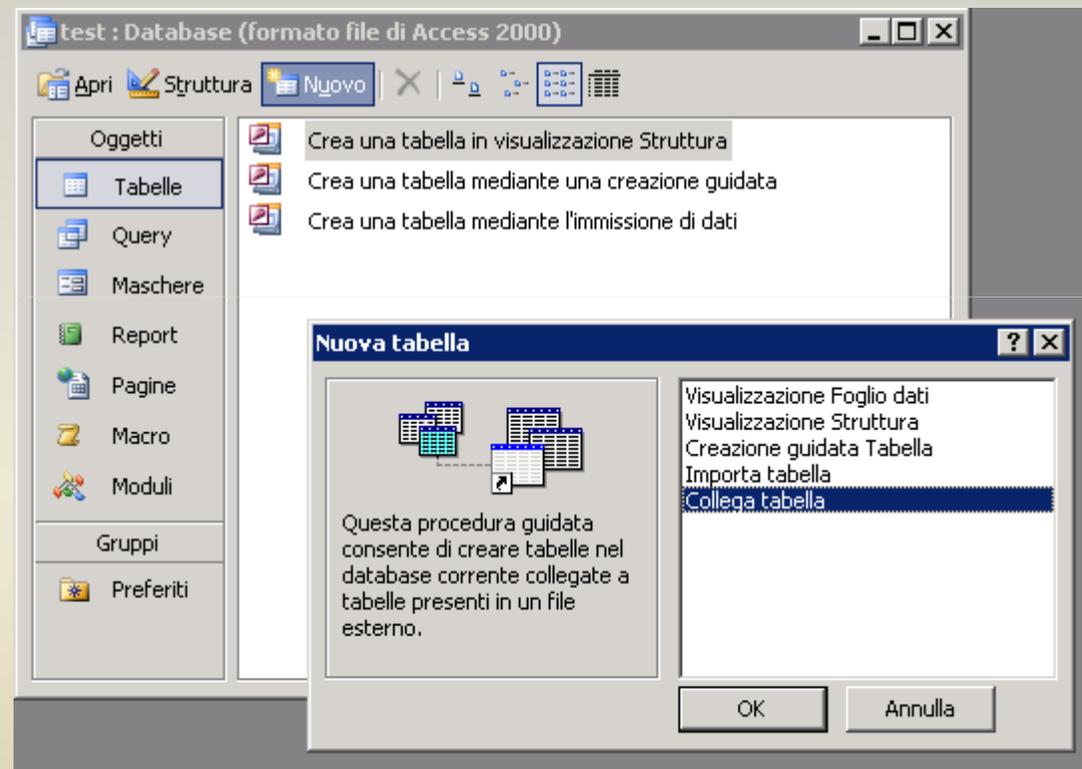
## REPARTI

Cod	Nome	Primario
A	Chirurgia	203
B	Medicina	574
C	Pediatria	530

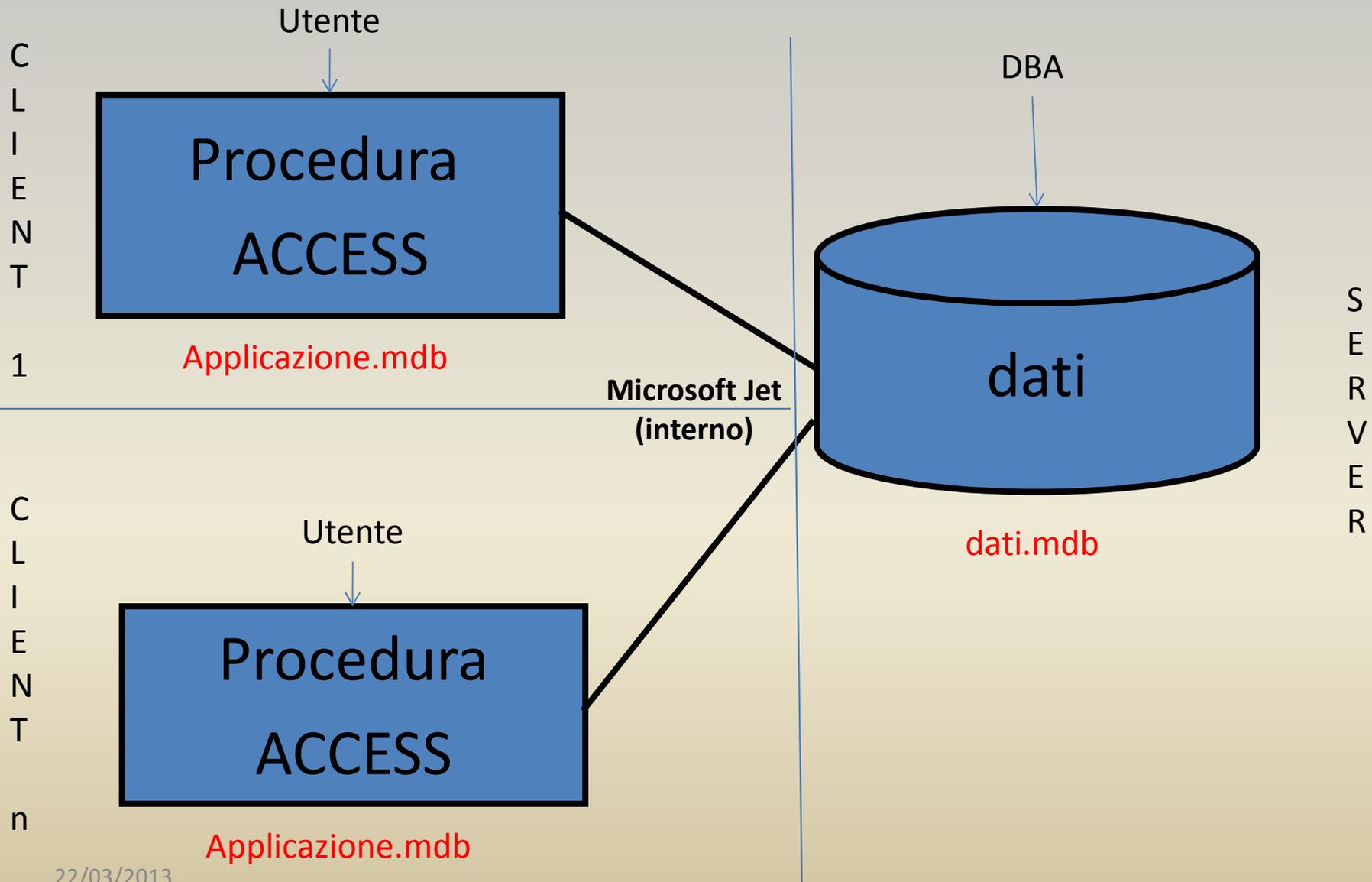
# Corso di Basi di Dati

- La scelta fatta sulla relazione RICOVERI presume che un paziente possa essere ricoverato solo una volta nello stesso giorno. Se supponiamo che questa ipotesi non venga soddisfatta, e che un paziente possa essere ammesso due o più volte nello stesso giorno, la relazione non sarebbe corretta. Infatti due o più ricoveri nello stesso giorno e nello stesso reparto dovrebbero avere anche la stessa data di dimissione, e così sarebbe rappresentata nella stessa riga nella relazione.
- I vincoli di integrità che esistono nel database sono tra l'attributo "Paziente" in RICOVERI e "Cod" in PAZIENTI, tra "Reparto" nella relazione RICOVERI e "Cod" nella relazione REPARTI, tra "Primario" in REPARTI e "Matr" nella relazione MEDICI e infine tra "Reparto" in MEDICI e "Cod" in REPARTI.
- I valori nulli possono essere ammessi negli attributi "Cognome" e "Nome" nella relazione PAZIENTI, "Fine" nella relazione RICOVERI, "Cognome" e "Nome" nella relazione MEDICI e "Nome" nella relazione REPARTI. Tutti questi attributi non sono chiavi e non hanno nessun vincolo di integrità referenziale.

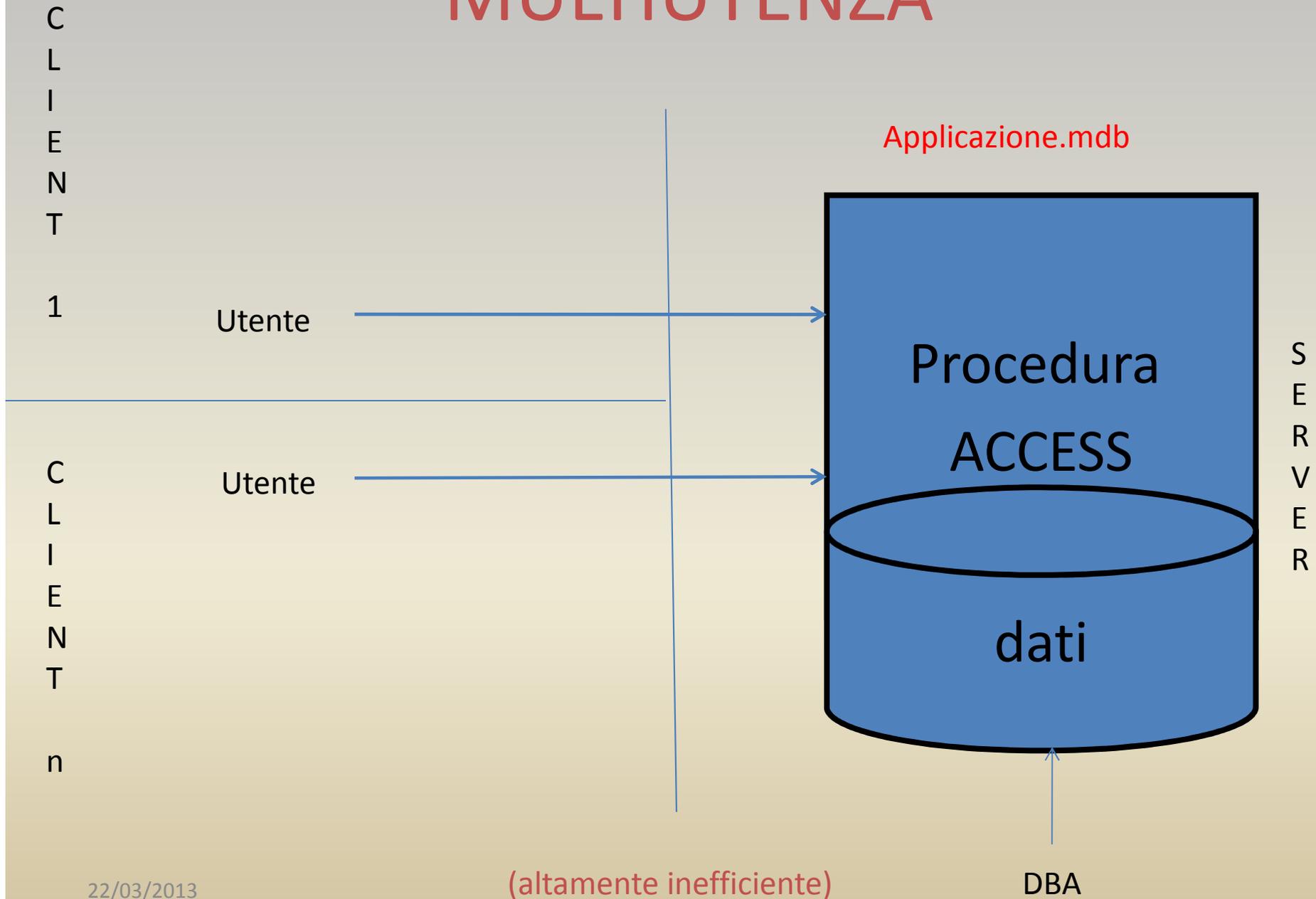
# Uso di MS Access (DBMS + Appl.)



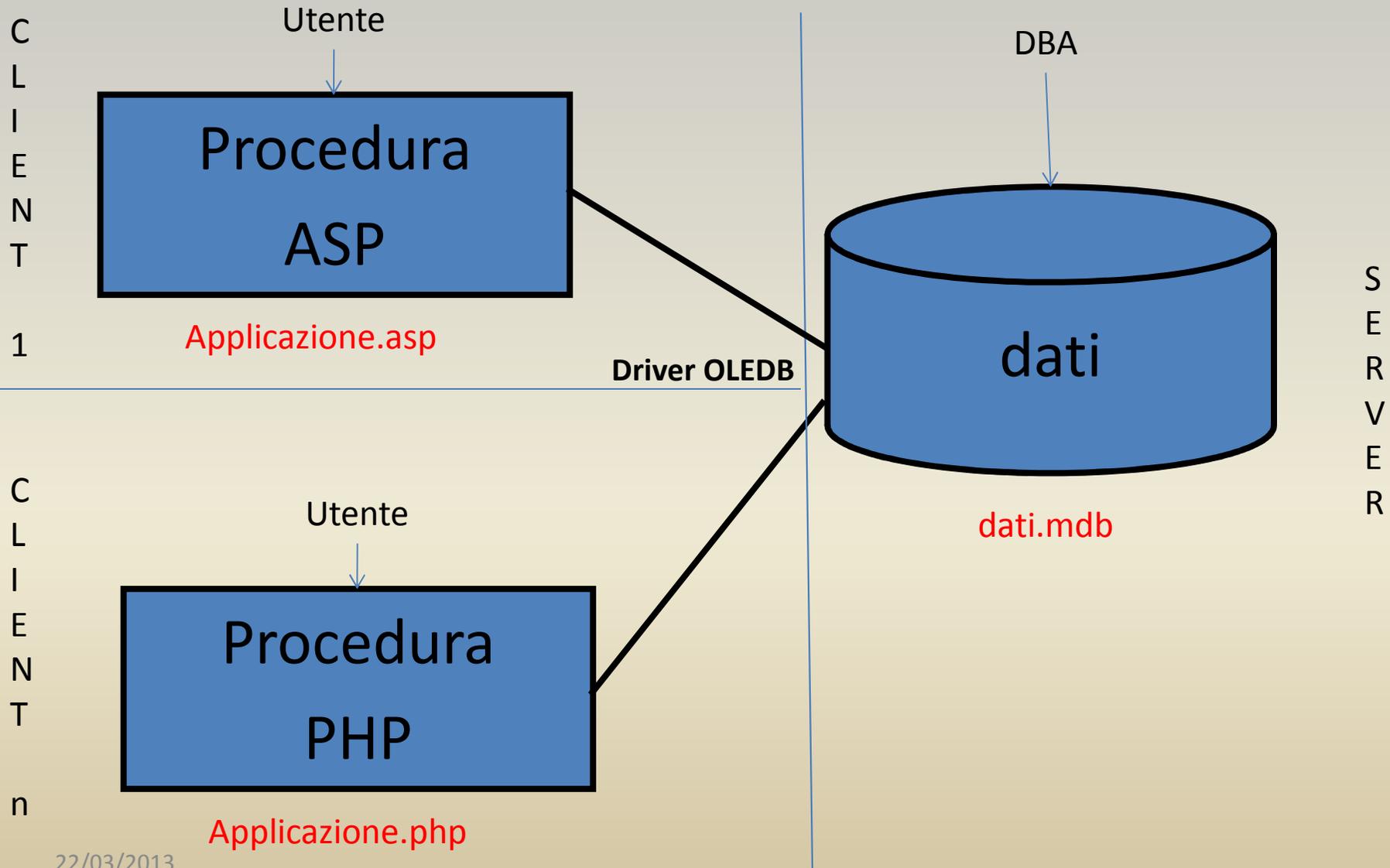
# Uso di MS Access (DBMS + Appl.)



# MULTIUTENZA



# Uso di MS Access (solo DBMS)



# Esempi di connessione ad un DB ACCESS

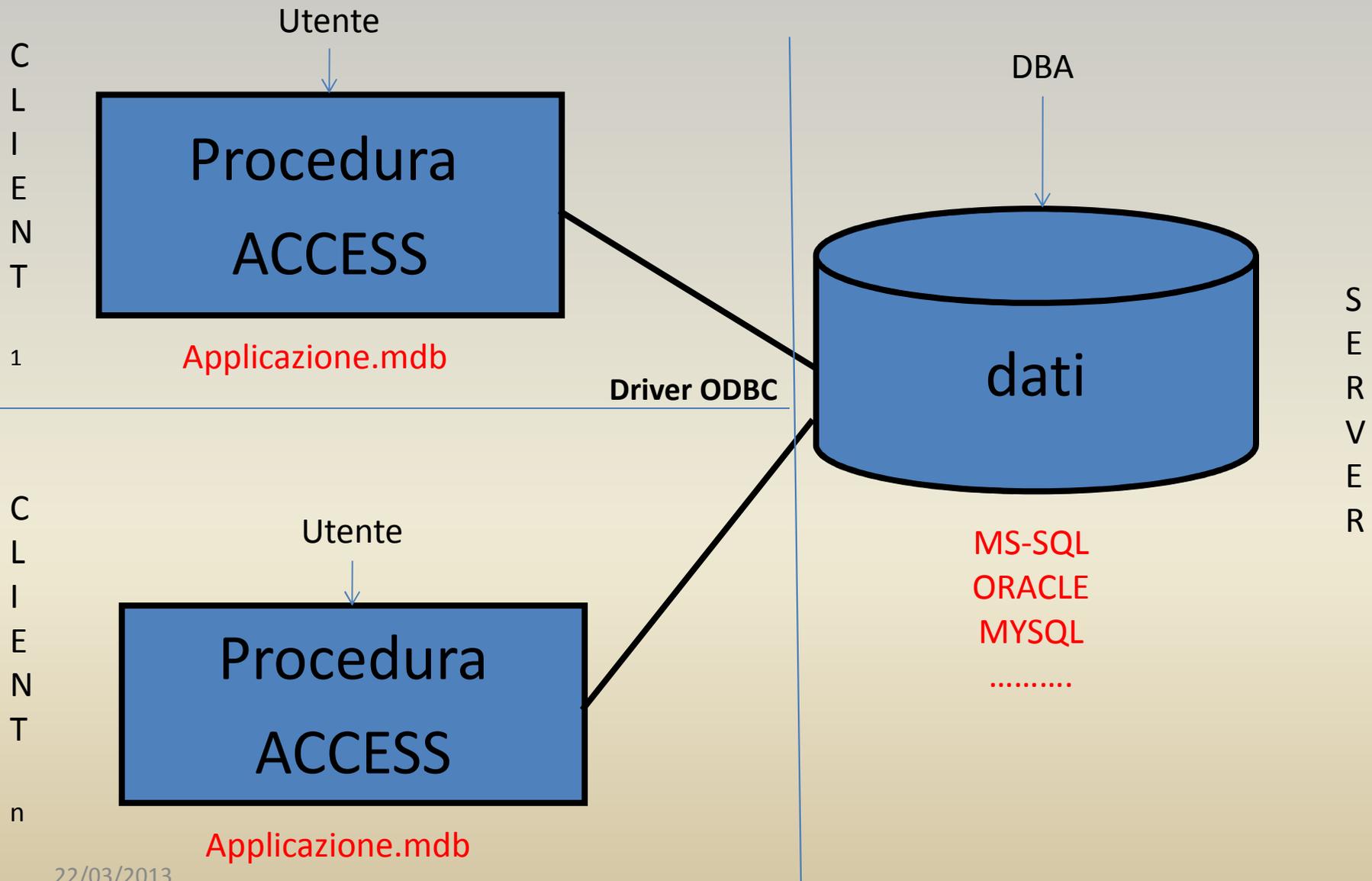
ASP

```
<%  
Set conn = Server.CreateObject("ADODB.Connection")  
conn.Open "DRIVER={Microsoft Access Driver (*.mdb)};DBQ=" &  
server.MapPath("/database/esempio.mdb")  
sql = "SELECT * FROM Tabella1"  
Set rs = Server.CreateObject("ADODB.Recordset")  
rs.Open sql, conn  
%>
```

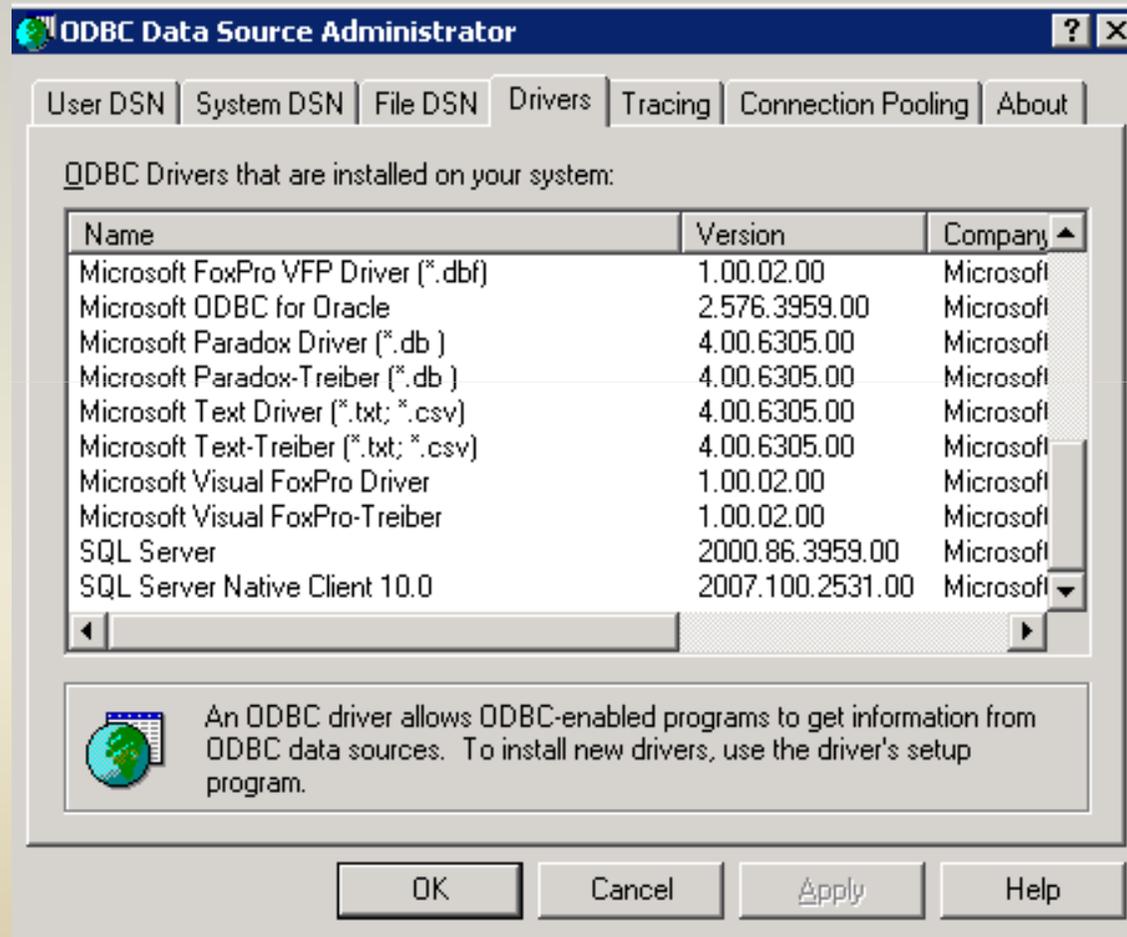
PHP

```
$connessione = new COM("ADODB.Connection");  
$stringa_di_connessione = "DRIVER={Microsoft Access Driver (*.mdb)};DBQ=" .  
realpath("testdb/testdb.mdb");  
$connessione->Open($stringa_di_connessione);  
$interrogazione = "select * from categorie";  
$recordset = new COM("ADODB.Recordset");
```

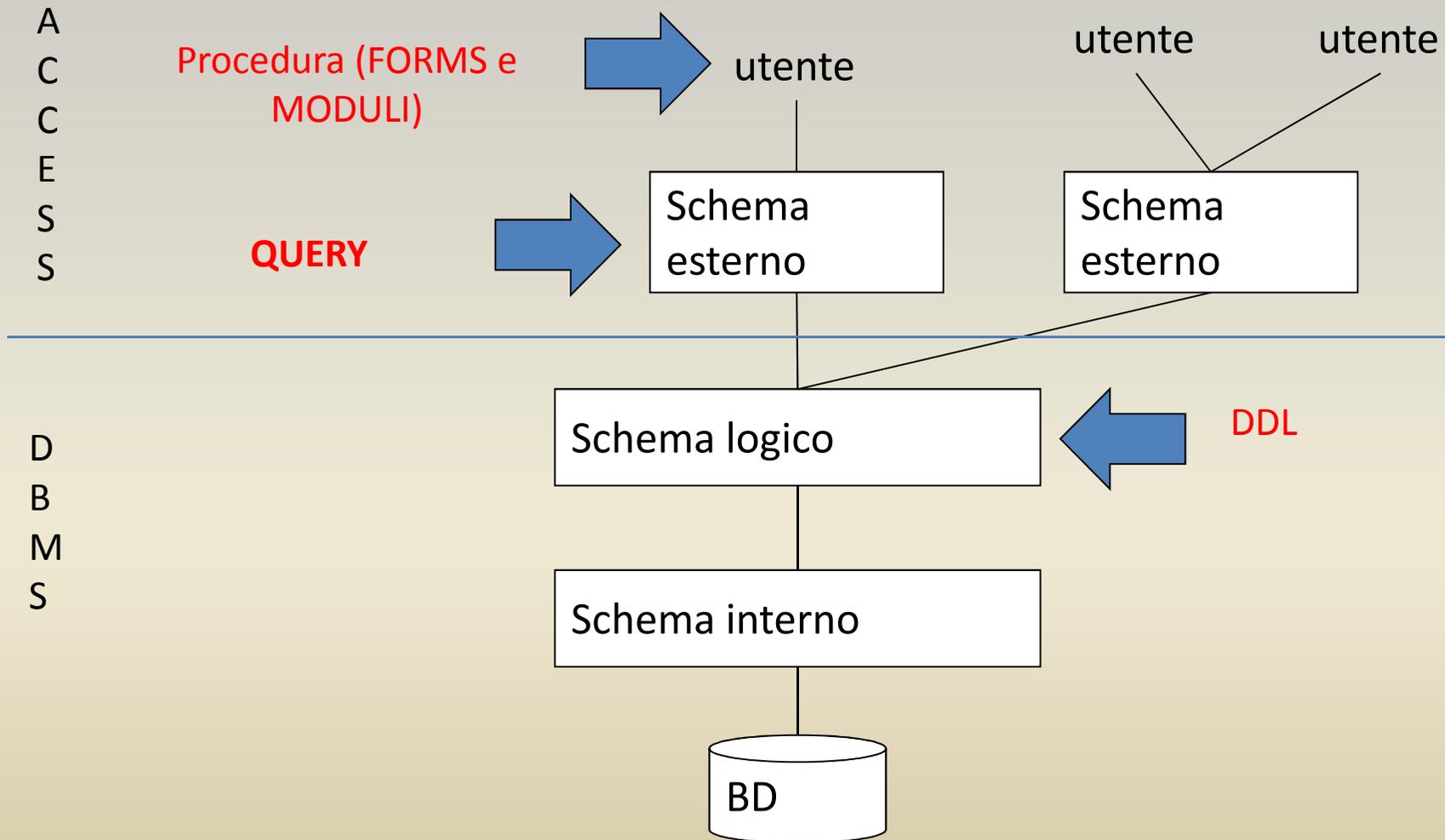
# Uso di MS Access (Appl.)



# Uso di MS Access (Appl.)

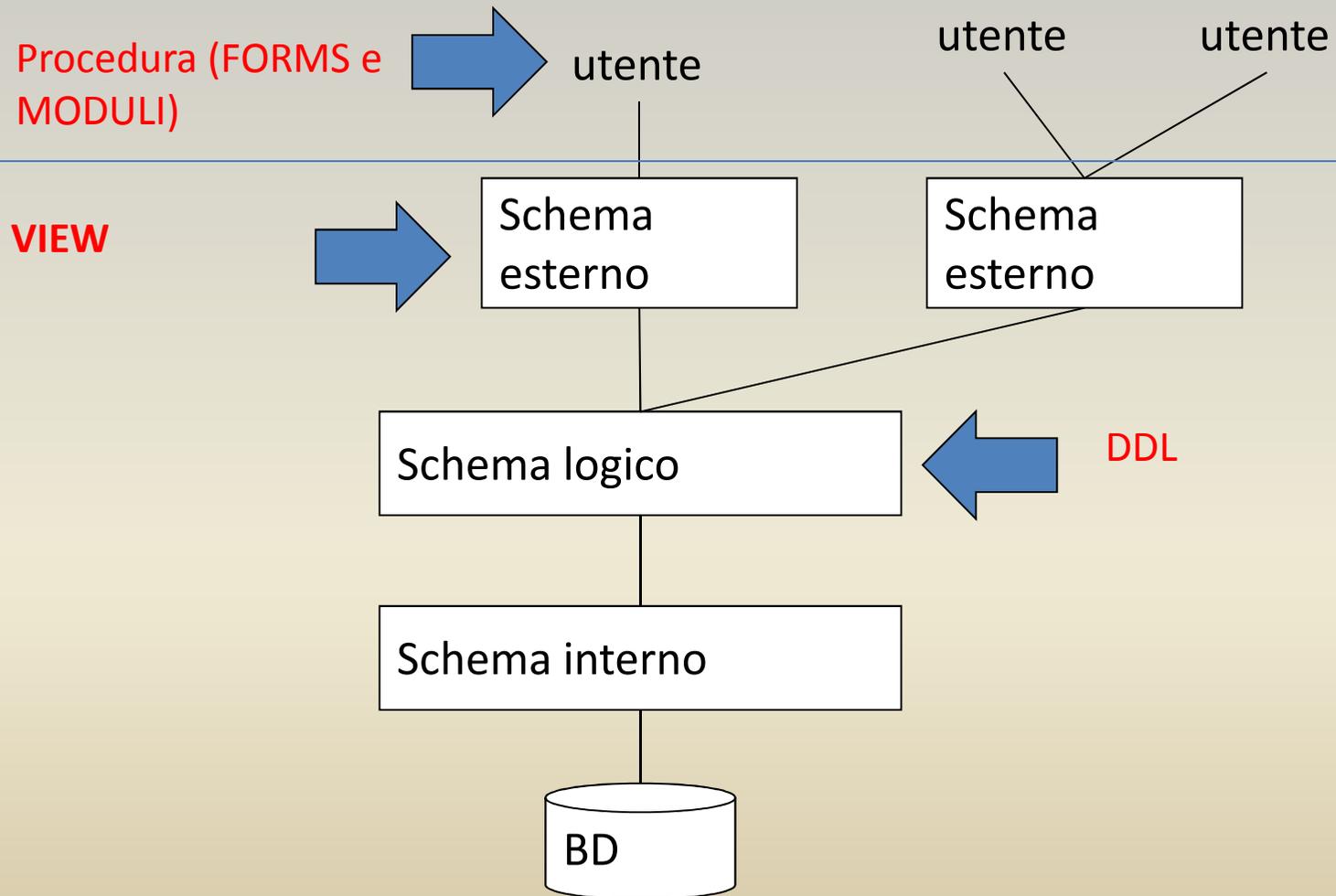


# MS-ACCESS Appl. (soluzione 1 – meno efficiente)



# MS-ACCESS Appl. (soluzione 2 – più efficiente)

A  
C  
C  
E  
S  
S



Copy of laure@ndi : Database (formato file di Access 2000)

Apri | Struttura | Nuovo

Oggetti

- Tabelle
- Query
- Maschere
- Report
- Pagine
- Macro
- Moduli

Gruppi

- Preferiti

Crea una tabella in visualizzazione Struttura	dbo_DidLau_Sottocommissioni
Crea una tabella mediante una creazione guidata	dbo_DidLau_Tabella cariche
Crea una tabella mediante l'immissione di dati	dbo_DidLau_V_Laureati
account	dbo_DidLau_V_TipoSottocommissione
dbo_Did_CDS	dbo_DidTiro_Progetti
dbo_Did_CorsiInCdS	dbo_DidTiro_RelAssegnazioni
dbo_Did_Docenti	dbo_DidTiro_Richieste
dbo_Did_Insegnamenti	dbo_ESSE3_V_CarriereLaureandi
dbo_Did_IstanzaCorso	dbo_Esse3_V_cds
dbo_Did_Mutuazioni	dbo_Gen_Gruppi
dbo_Did_Plurimodulari	dbo_Gen_Organizzazione
dbo_Did_RelCorsiIstanze	dbo_Gen_Persone
dbo_Did_Studenti	dbo_Gen_RelPersoneOrganizzazione
dbo_Did_TipiInsegnamento	dbo_Gruppi_V_DisponibilitaIAP
dbo_DID_V_CdSDocenti	dbo_Gruppi_V_DisponibilitaINF
dbo_DidLau_Candidati	dbo_Gruppi_V_DisponibilitaWEA
dbo_DidLau_CarriereLaureandi	parametri
dbo_DidLau_Commissioni lauree/diploma	<b>tabella cariche</b>
dbo_DidLau_Composizione commissioni	Tabella esami prosecuzioni
dbo_DidLau_Composizione sottocommissioni	tabella gruppi-gen_gruppo
dbo_DidLau_Disponibilita	Tabella Incarichi
dbo_DidLau_GruppoCdS	tabella stato militare
dbo_DidLau_Lauree	tmp_CarriereLaureandi
dbo_DidLau_LegendaDisponibilita	tmp_Gruppi_V_DisponibilitaIAP
dbo_DidLau_Prosecuzioni	tmp_Gruppi_V_DisponibilitaINF
dbo_DidLau_Relatori	tmp_Gruppi_V_DisponibilitaWEA
dbo_DidLau_Sessioni	

22/03/2013

# Operazioni standard sui dati

- **Inserimento**
- **Ricerca**
- **Modifica**
- **Cancellazione**
- **Selezione**

# USO DEGLI INDICI

Esempio: supponiamo di cercare un nome fra i seguenti

Giuseppe

Irene

Fabio

Carlo

Lola

Elena

Fabio

la procedura da fare è:

1. leggi il primo nome della lista
2. se nome trovato fermati
3. leggi prossimo nome torna a 2

Nel caso migliore leggo 1 volta. Nel caso peggiore leggo 7 volte.

Mediamente leggo  $\frac{1+7}{2} = 4$  volte

2

Se il dato non è presente si casca sempre nel caso peggiore.

# USO DEGLI INDICI

Supponiamo ora di organizzare l'elenco precedente in ordine alfabetico:

Carlo  
Elena  
Fabio  
Fabio  
Giuseppe  
Irene  
Lola

Se la procedura è la solita:

1. leggi il primo nome della lista
2. se nome trovato fermati
3. leggi prossimo nome torna a 2

Si ha lo stesso numero di letture. Cambia però il caso del dato non presente che rientra nelle letture medie.

# USO DEGLI INDICI

Supponiamo ora cambiare la procedura

Carlo  
Elena  
Fabio  
Fabio  
Giuseppe  
Irene  
Lola

La nuova procedura è:

1. leggi nome centrale
2. se nome trovato fermati
3. se nome > cercato considera la metà superiore del file
4. se nome < cercato considera la metà inferiore del file
5. vai a 1

Nel caso migliore leggo 1 volta. Nel caso peggiore leggo 3 volte.

Mediamente leggo  $\frac{1+3}{2} = 2$  volte

# USO DEGLI INDICI

## PROBLEMA:

Il caso si presenta più complicato quando aumenta il numero dei campi e il numero delle ricerche possibili.

1)Giuseppe	Rossi
2)Irene	Bianchi
3)Fabio	Verdi
4)Carlo	Bianchi
5)Lola	Neri
6)Elena	Bianchi
7)Fabio	Rossi

Se si vuol ricercare sia per nome che per cognome non si può dare un'organizzazione unica ai dati e non si può neanche riordinarli prima di ciascuna ricerca.

# USO DEGLI INDICI

## SOLUZIONE:

Si usano degli *indici*: archivi a parte nei quali si ha il riepilogo dei dati organizzati e la maniera di recuperare l'intero record nell'archivio principale che rimane inalterato

<b>Nome</b>		<b>Cognome</b>	
Carlo	4	Bianchi	2, 4, 6
Elena	6	Neri	5
Fabio	3, 7	Rossi	1, 7
Giuseppe	1	Verdi	3
Irene	2		
Lola	5		

# In ACCESS (struttura tabella)

Medici	
Nome campo	Tipo dati
Matr	Numerico
Cognome	Testo
Nome	Testo
Reparto	Testo

Generale	
Dimensione campo	20
Formato	
Maschera di input	
Etichetta	
Valore predefinito	
Valido se	
Messaggio errore	
Richiesto	No
Consenti lunghezza zero	Sì
Indicizzato	Sì (Duplicati ammessi)
Compressione Unicode	Sì
Modalità IME	Nessun controllo
Modalità frase IME	Nessuna conversione
Smart tag	

