

- Attraverso opportuni linguaggi che estendono SQL si possono definire due **componenti procedurali** nello schema:
- Procedure memorizzate (**stored procedures**)
- **Trigger**
- Fanno parte dello schema e non dei programmi applicativi
- Con opportuni diritti le procedure memorizzate possono essere richiamate dagli utenti della base di dati

```
CREATE FUNCTION contaStudenti IS
  DECLARE
    numStudenti INTEGER;
  BEGIN
    SELECT COUNT(*) INTO numStudenti FROM STUDENTI;
    RETURN (numStudenti);
  END
```

- I trigger si basano sul paradigma evento-condizione-azione (ECA):

```
CREATE TRIGGER Nome
  PrimaODopoDi Evento {, Evento}
  ON Tabella [WHEN Condizione]
  [Granularità]
  Azione
  PrimaODopoDi := BEFORE | AFTER
  Evento := INSERT | DELETE | UPDATE OF Attributi
  Granularità := FOR EACH ROW | FOR EACH STATEMENT
```

```
CREATE TRIGGER ControlloStipendio
  BEFORE INSERT ON Impiegati
  DECLARE
    StipendioMedio FLOAT
  BEGIN
    SELECT avg(Stipendio) INTO StipendioMedio
    FROM Impiegati
    WHERE Dipartimento = :new.Dipartimento;
    IF :new.Stipendio > 2 * StipendioMedio
    THEN RAISE_APPLICATION_ERROR(-2061, 'Stipendio alto')
    END IF;
  END;
```

- Proprietà essenziale dei trigger: terminazione
- Utilità dei trigger
 - Trattare vincoli non esprimibili nello schema
 - Attivare automaticamente azioni sulla base di dati quando si verificano certe condizioni

- Un ruolo può essere visto come un utente o un gruppo
- Quando si crea un ruolo si può inserire in un altro ruolo (quindi si "ereditano" i privilegi del ruolo in cui si inserisce)
- CREATE ROLE segretarie
- ... si forniscono dei privilegi a "segretari"
- CREATE ROLE marta LOGIN WITH PASSWORD 'pippo' IN ROLE segretarie

- Chi crea lo schema della BD è l'unico che può fare CREATE, ALTER e DROP
- Chi crea una tabella stabilisce i modi in cui altri possono farne uso:
 - GRANT Privilegi ON Oggetto TO Utenti [WITH GRANT OPTION]

- Tipi di privilegi:
 - SELECT: lettura di dati
 - INSERT [(Attributi)]: inserire record (con valori non nulli per gli attributi)
 - DELETE: cancellazione di record
 - UPDATE [(Attributi)]: modificare record (o solo gli attributi)
 - REFERENCES [(Attributi)]: definire chiavi esterne in altre tabelle che riferiscono gli attributi.
- WITH GRANT OPTION: si possono trasferire i privilegi ad altri utenti.

CONTROLLO DEGLI ACCESSI (cont.)

9

- Chi definisce una tabella o una VIEW ottiene automaticamente tutti i privilegi su di esse, ed è l'unico che può fare un DROP e può autorizzare altri ad usarla con GRANT.
- Nel caso di viste, il "creatore" ha i privilegi che ha sulle tabelle usate nella definizione.
- Le autorizzazioni si annullano con il comando:
 - REVOKE [GRANT OPTION FOR] Privilegi ON Oggetto FROM Utenti [CASCADE]
- Quando si toglie un privilegio a U, lo si toglie anche a tutti coloro che lo hanno avuto solo da U.

ESEMPI DI GRANT

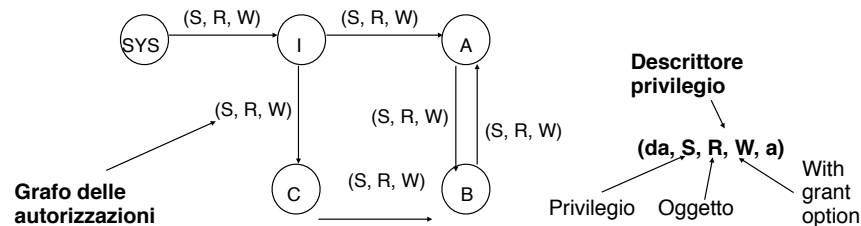
10

- GRANT INSERT, SELECT ON Esami TO Tizio.
- GRANT DELETE ON On Esami TO Capo WITH GRANT OPTION
 - Capo può cancellare record e autorizzare altri a farlo.
- GRANT UPDATE (voto) ON Esami TO Sicuro
 - Sicuro può modificare solo il voto degli esami.
- GRANT SELECT, INSERT ON VistaEsamiBD1 TO Albano
 - Albano può interrogare e modificare solo i suoi esami.

GRAFO DELLE AUTORIZZAZIONI

11

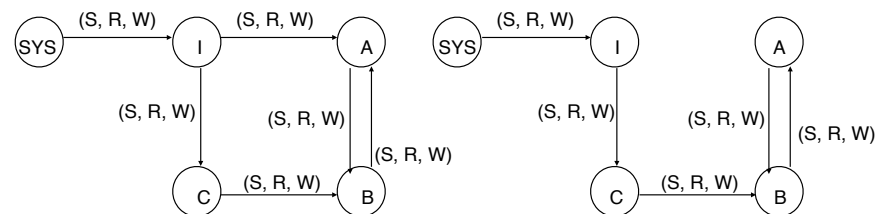
- L'utente I ha creato la tabella R e innesca la seguente successione di eventi:
 - I: GRANT SELECT ON R TO A WITH GRANT OPTION
 - A: GRANT SELECT ON R TO B WITH GRANT OPTION
 - B: GRANT SELECT ON R TO A WITH GRANT OPTION
 - I: GRANT SELECT ON R TO C WITH GRANT OPTION
 - C: GRANT SELECT ON R TO B WITH GRANT OPTION



GRAFO DELLE AUTORIZZAZIONI: PROPRIETA'

12

- Se un nodo N ha un arco uscente con un privilegio, allora esiste un cammino da SYSTEM a N con ogni arco etichettato dallo stesso privilegio + WGO.
- Effetto del REVOKE, ad es.
 - I: REVOKE SELECT ON R FROM A CASCADE
- e poi
 - I: REVOKE SELECT ON R FROM C CASCADE



- Cosa sono e a cosa servono
- Non è un comando standard dell'SQL e quindi ci sono differenze nei vari sistemi
 - CREATE INDEX NomeIdx ON Tabella(Attributi)
 - CREATE INDEX NomeIdx ON Tabella WITH STRUCTURE = BTREE, KEY = (Attributi)
 - DROP INDEX NomeIdx

- Alcuni esempi di tabelle, delle quali si mostrano solo alcuni attributi, sono:
 - Tabella delle password:
PASSWORD(username, password)
 - Tabella delle basi di dati:
SYSDB(dbname, creator, dbpath, remarks)
 - Tabella delle tabelle (type = view or table):
SYSTABLES(name, creator, type, colcount, filename, remarks)

- Alcuni esempi di tabelle, delle quali si mostrano solo alcuni attributi, sono:
 - Tabella degli attributi:
SYSCOLUMNS(name, tname, tbcreator, colno, coltype, lenght, default, remarks)
 - Tabella degli indici:
SYSINDEXES(name, tname, creator, uniquerule, colcount)
 - e altre ancora sulle viste, vincoli, autorizzazioni, etc. (una decina).

- Gli aspetti procedurali arricchiscono lo schema con operatori "di base" sui dati (procedure memorizzate) e permettono di esprimere vincoli d'integrità dinamici (trigger)
- I comandi GRANT / REVOKE + viste offrono ampie possibilità di controllo degli usi dei dati.
- Il catalogo permette di utilizzare tecniche di "riflessione" sui dati

- La padronanza di tutti questi meccanismi -- e di altri che riguardano aspetti fisici, affidabilità, sicurezza -- richiede una professionalità specifica (DBA).