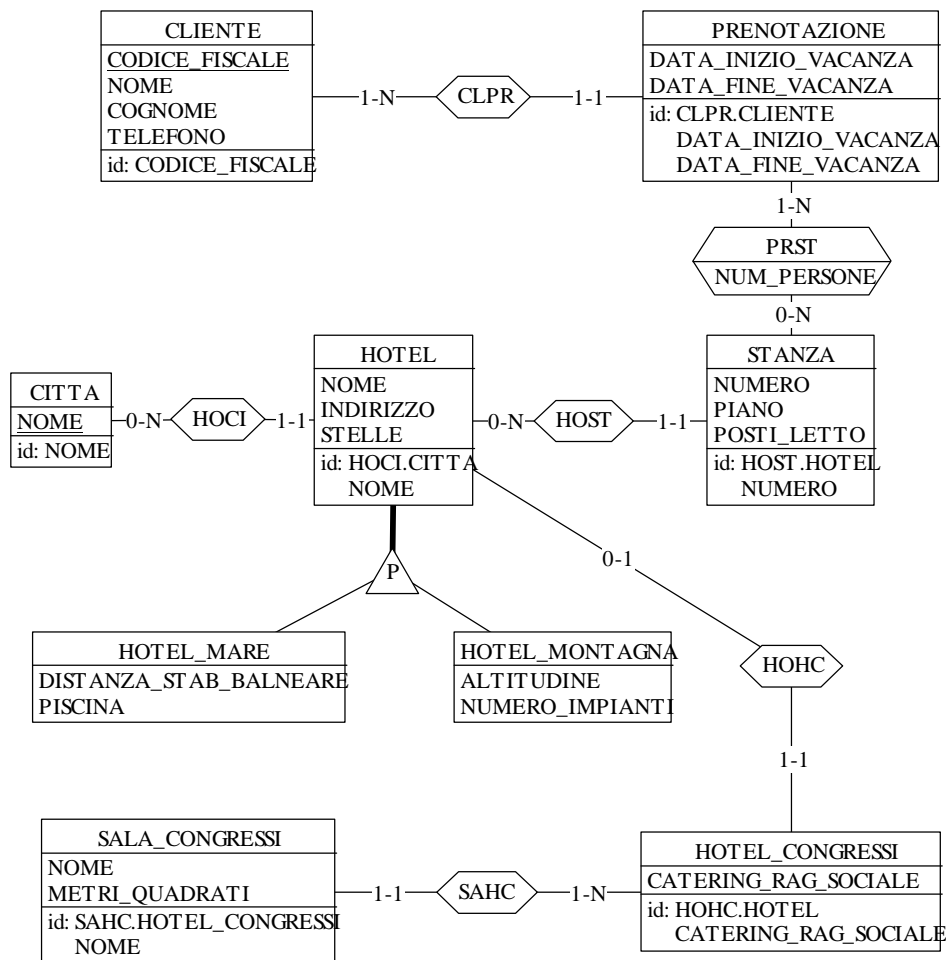


Tempo a disposizione: 2 ore

1) Progettazione concettuale



Commenti

- A causa dei limiti di DB-MAIN, la relazione di subset tra HOTEL\_CONGRESSI e HOTEL viene rappresentata mediante un'associazione 1-1
- Si noti che il vincolo che le stanze oggetto di una prenotazione facciano tutte parte di uno stesso hotel non è rappresentabile nel modello E-R

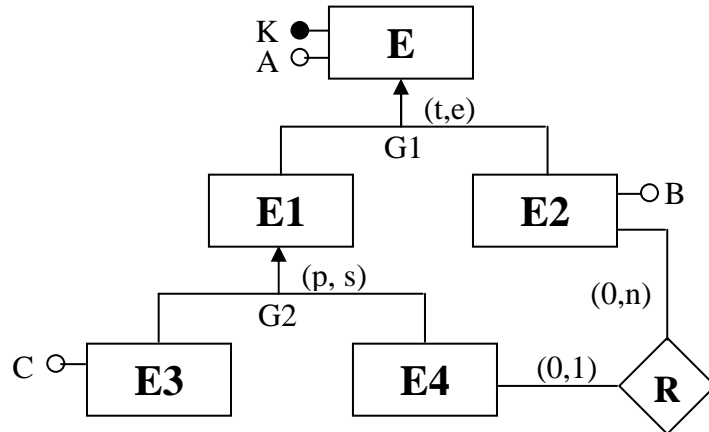
## Sistemi Informativi L-B - Prova integrata

20 Aprile 2004

### Risoluzione

#### 2) Progettazione logica

Dato lo schema concettuale in figura



e considerando che:

- Tutti gli attributi sono di tipo INT
- Le operazioni fanno propendere per un:
  - collasso verso il basso della gerarchia G1
  - collasso verso l'alto della gerarchia G2
- R non viene tradotta come relazione a sè stante
- Ad E2 (tramite R) sono associate istanze di E4 che non sono anche istanze di E3

si progettano gli opportuni schemi relazionali e si definiscono tali schemi facendo uso dell'SQL di DB2; per gli eventuali vincoli non esprimibili a livello di schema si predispongono opportune query di verifica da eseguire prima di effettuare inserimenti di tuple, allo scopo di evitare che tali inserimenti violino i vincoli stessi

```
CREATE TABLE E1 (  
    K INT NOT NULL PRIMARY KEY,  
    A INT NOT NULL,  
    C INT,  
    SEL3 SMALLINT NOT NULL CHECK (SEL3 IN (0,1)), -- 1 se appartiene a E3  
    SEL4 SMALLINT NOT NULL CHECK (SEL4 IN (0,1)), -- 1 se appartiene a E4  
    KR INT REFERENCES E2,  
    CONSTRAINT CHECK_HIERARCHY  
        CHECK ( ((SEL3=1 AND C IS NOT NULL) OR (SEL3=0 AND C IS NULL))  
              AND (KR IS NULL OR SEL4=1) ),  
    CONSTRAINT CHECK_R  
        CHECK (C IS NULL OR KR IS NULL);
```

```
CREATE TABLE E2 (  
    K INT NOT NULL PRIMARY KEY,  
    A INT NOT NULL,  
    B INT NOT NULL);
```

-- Query che, data una tupla (k, ...) che si vuole inserire in E1, verifica che k non sia già presente  
-- in E2. Una query analoga va eseguita per l'inserimento di una tupla (k, ...) in E2

```
SELECT * -- Ok se non restituisce nessuna tupla  
FROM E2  
WHERE E2.K = k;
```

#### 3) Indici

Considerando la casistica vista a lezione, le differenze fra i diversi tipi di indici sono così riassumibili:

- Clustered / Unclustered: un indice è detto clustered se è costruito sul campo su cui i record nel file dati sono mantenuti ordinati, altrimenti è detto unclustered.
- Primary / Secondary: un indice è detto primary (primario) se è costruito su un campo a valori non ripetuti (chiave relazionale), altrimenti è detto secondary (secondario).
- Dense / Sparse: negli indici dense (densi) il numero di puntatori è pari al numero di record del file dati, negli indici sparse (sparsi) è minore (tipicamente uno per pagina dati).
- Single-level / Multi-level: indicizzando ricorsivamente un indice si ottiene una struttura definita multi-level; se un indice non subisce ulteriori indicizzazioni, esso viene definito single-level.