

Corso di Basi di Dati A.A. 2020/21  
Progetto "Gestione Progetti ANSA"

Federico Matteoni  
Mat. 530257

29/01/2021

# 1 Descrizione del dominio

Si vuole progettare un sistema di distribuzione di notizie agli utenti.

Ogni notizia ha un titolo, autore, testo, data, ora e un'insieme di uno o più argomenti.

Ogni argomento ha un nome e può essere di tipo Premium o Economy.

Un utente può essere un autore e poter inviare articoli, firmandoli con proprio nome e cognome.

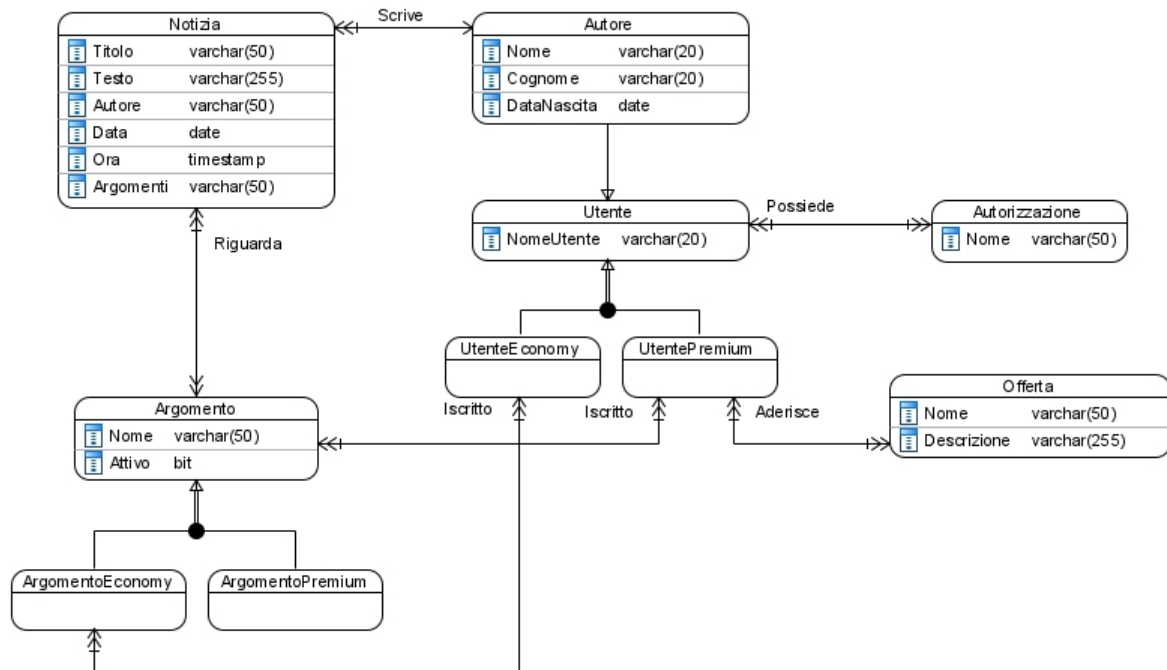
Ogni utente può essere Premium o Economy e iscriversi agli argomenti per cui ricevere notizie.

Un utente Premium può iscriversi ad ogni argomento, mentre un utente Economy può iscriversi solo agli argomenti Economy.

Un utente Premium può ricevere offerte/sconti speciali.

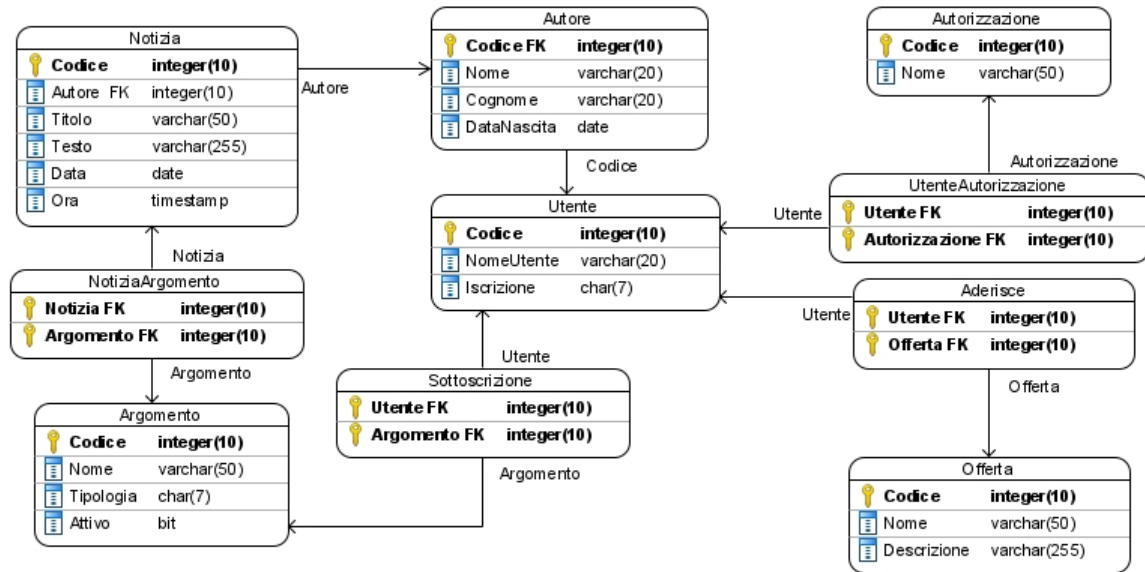
Alcuni utenti hanno delle autorizzazioni speciali.

## 2 Schema Concettuale



Powered By QV/visual Paradigm Community Edition

### 3 Schema Logico Relazionale



Powered By Visual Paradigm Community Edition

#### Notazione testuale

Notizia(Codice, Autore\*, Titolo, Testo, Data, Ora)

NotiziaArgomento(Notizia\*, Argomento\*)

Argomento(Codice, Nome, Tipologia, Attivo)

Utente(Codice, NomeUtente, Iscrizione)

Autore(Codice\*, Nome, Cognome, DataNascita)

Sottoscrizione(Utente\*, Argomento\*)

Autorizzazione(Codice, Nome)

UtenteAutorizzazione(Utente\*, Autorizzazione\*)

Offerta(Codice, Nome, Descrizione)

Aderisce(Utente\*, Offerta\*)

#### Vincoli non catturati graficamente

Ogni utente riferito dalla tabella Aderisce ha il campo "Iscrizione" con valore pari a "premium"

Ogni Argomento con campo "Tipologia" con valore "premium" può essere legato solamente ad un Utente con campo "Tipologia" con valore "premium".

## 4 Query SQL

1. Uso di proiezione, join e restrizione

La lista dei titoli di tutte le notizie di cronaca.

```
SELECT Notizia.Titolo
FROM Notizia
JOIN NotiziaArgomento ON Notizia.Codice = NotiziaArgomento.Argomento
JOIN Argomento ON NotiziaArgomento.Argomento = Argomento.Codice
WHERE Argomento.Nome = "Cronaca"
```

2. Uso di group by con having, where e sort

La classifica dei giorni del 2020 con più di 5 notizie.

```
SELECT Data, COUNT(*) AS Notizie
FROM Notizia
WHERE Data.YEAR = 2020
GROUP BY Data
HAVING Notizie > 5
ORDER BY Notizie
```

3. Uso di join, group by con having e where

La lista degli autori nati dopo il 2000 con più di 10 notizie pubblicate.

```
SELECT Nome, Cognome, COUNT(*) AS Notizie
FROM Autore
JOIN Notizia ON Notizia.Autore = Autore.Codice
WHERE DataNascita.YEAR >= 2000
GROUP BY Autore.Codice
HAVING Notizie > 10
```

4. Uso di select annidata con quantificazione esistenziale

La lista degli utenti premium che hanno aderito ad almeno un'offerta.

```
SELECT NomeUtente
FROM Utente
WHERE EXISTS
    (SELECT Utente
    FROM Aderisce
    WHERE Utente.Codice = Aderisce.Utente)
```

5. Uso di select annidata con quantificazione universale

La lista degli autori che hanno pubblicato solo notizie di cronaca.

```
SELECT Nome, Cognome
FROM Autore
WHERE Autore.Codice NOT IN
    (SELECT Notizia.Autore
    FROM Notizia
    JOIN NotiziaArgomento ON Notizia.Codice = NotiziaArgomento.
        Argomento
    JOIN Argomento ON NotiziaArgomento.Argomento = Argomento.Codice
    WHERE Argomento.Nome <> "Cronaca")
```

6. Uso di subquery di confronto quantificato usando una subquery

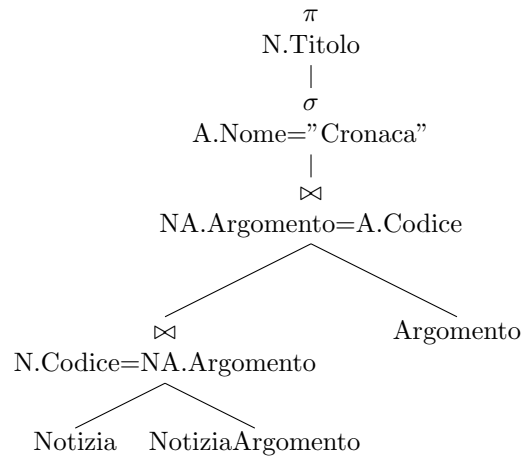
La lista degli utenti con più di metà delle autorizzazioni possibili.

```
SELECT NomeUtente, COUNT(UtenteAutorizzazione.Autorizzazione) AS
    Autorizzazioni
FROM Utente
JOIN UtenteAutorizzazione ON Utente.Codice = UtenteAutorizzazione.Utente
WHERE (Autorizzazioni*2) > (SELECT COUNT(*) FROM Autorizzazione)
GROUP BY Utente.Codice
```

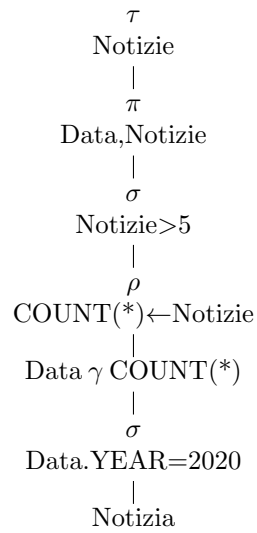
## 5 Piani di accesso

1. Scrivere un piano di accesso logico delle query 1, 2 e 3  
Per ragioni di spazio, laddove implicito i nomi delle tabelle sono abbreviati con le iniziali.

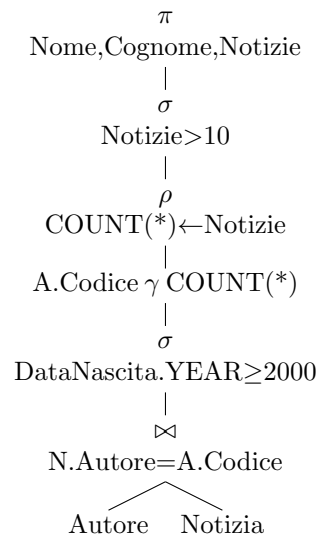
1.



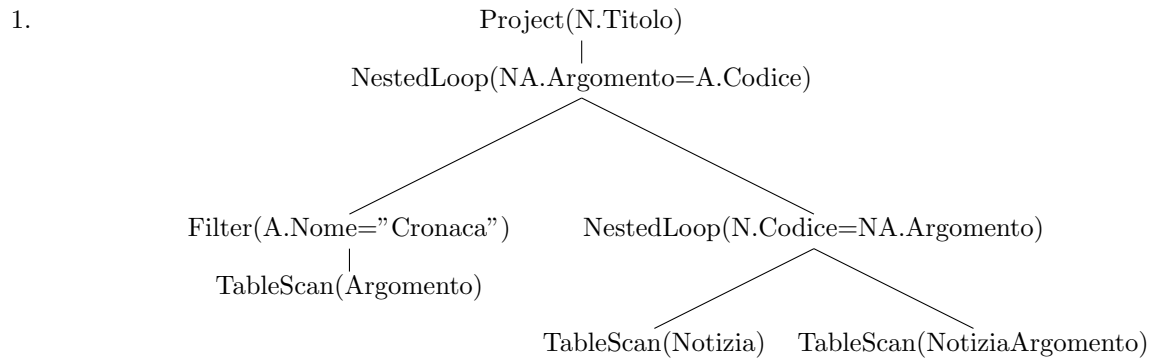
2.



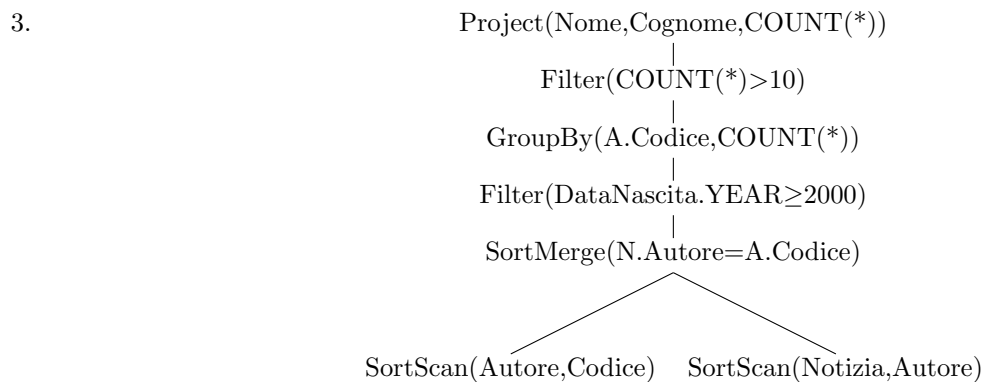
3.



2. Scrivere un piano di accesso fisico efficiente per i tre piani di accesso logico al punto 1 che non fanno uso di indici e (opzionale) verificare se la sort prima della group by può essere evitata  
Per ragioni di spazio, laddove implicito i nomi delle tabelle sono abbreviati con le iniziali.



Posso evitare la Sort prima della GroupBy ordinando immediatamente la tabella Notizia in fase di scansione.

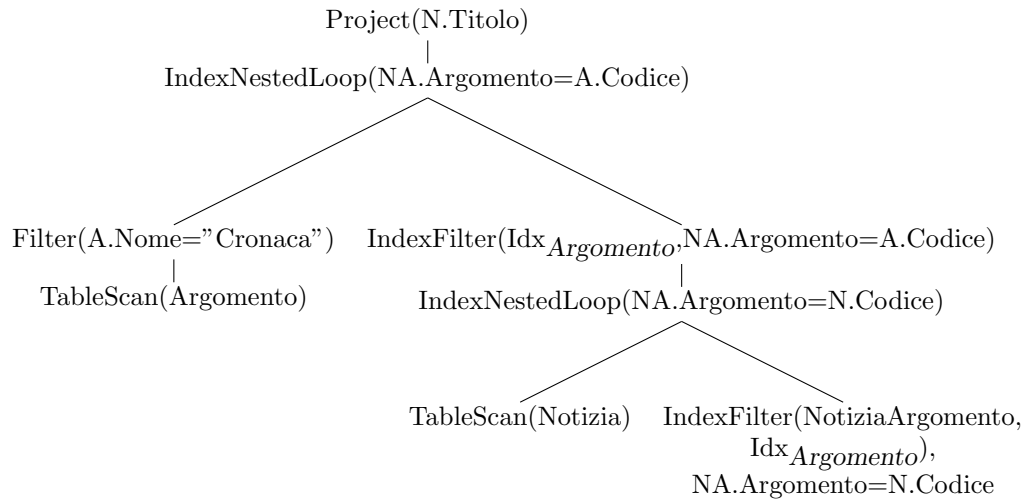


Posso evitare la Sort prima della GroupBy ordinando immediatamente la tabella Autore in fase di scansione.

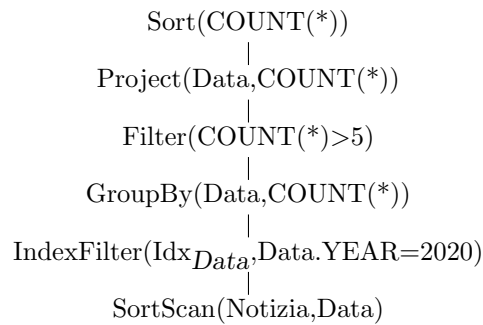
3. Scrivere un piano di accesso fisico efficiente per i tre piani di accesso logico al punto 1 che fanno uso di due indici (o comunque del numero massimo di indici possibili) e (opzionale) verificare se la sort prima della group by può essere evitata

Per ragioni di spazio, laddove implicito i nomi delle tabelle sono abbreviati con le iniziali.

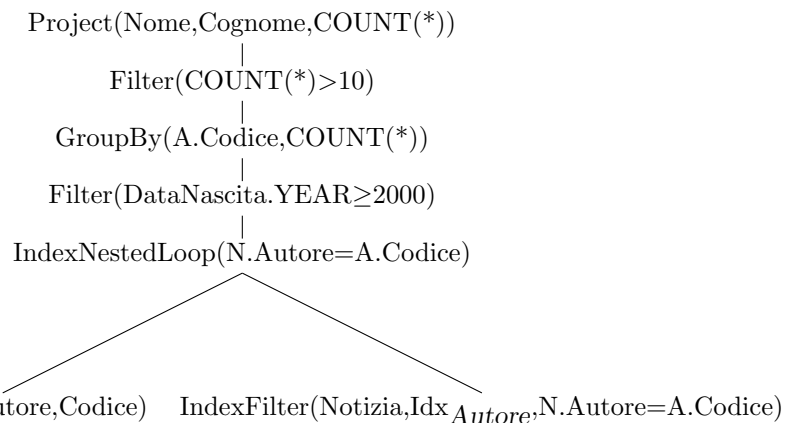
1.



2.



3.



Posso evitare la Sort prima della GroupBy ordinando immediatamente la tabella Autore in fase di scansione.