

1. ABSTRACT

Si rappresenta un negozio che vende articoli musicali quali vinili, cassette audio e CD di vari generi. Ciascun album musicale è prodotto da uno o più artisti (nel caso di una band si memorizza solo il nome della stessa) che collaborano con un producer per incidere i brani. I producer appartengono ad una etichetta discografica alla quale gli artisti si appoggiano per creare il proprio album; le etichette discografiche sono specializzate in uno o più generi musicali. Periodicamente il negozio organizza degli eventi giornalieri legati ad un genere: durante l'evento tutti i prodotti di quel genere sono scontati. Si memorizzano due tipi di cataloghi: quello degli articoli che sono presenti in negozio e quello degli articoli ordinati nel caso in cui fossero esauriti. Gli articoli esposti possono essere raggruppati in bundle o acquistati singolarmente, mentre quelli in arrivo possono essere solo prenotati fornendo alcuni dati personali e un acconto. Si memorizzano infine gli scontrini per il calcolo delle tasse.

2. ANALISI DEI REQUISITI

Si vuole creare un database per gestire un negozio che vende articoli musicali, tenendo traccia anche delle informazioni legate al prodotto, degli articoli in vendita e in arrivo, prenotati e venduti. Inoltre è possibile organizzare degli eventi giornalieri.

In particolare gli articoli musicali sono descritti da:

- Codice articolo
- Nome
- Descrizione

Il codice è legato all'album e all'articolo, quindi più copie dello stesso articolo musicale hanno lo stesso codice (nel catalogo viene poi indicata la quantità). Ad esempio due CD dello stesso album hanno lo stesso codice, ma un CD e un vinile dello stesso album hanno codici diversi.

Un bundle è un particolare articolo che contiene più articoli musicali, anche di vario tipo e genere.

Ciascun articolo è distinto in vinile, CD e cassetta audio dove:

- Vinile possiede:
 - Giri
 - Condizione copertina
 - Condizione disco
- CD possiede:
 - RPM
 - Formato logico
- Cassetta audio possiede:
 - Formato
 - Tipo nastro

Ogni articolo è prodotto da uno o più artisti.

L'artista è caratterizzato da:

- Nome d'arte
- Nazionalità
- Nome
- Cognome

Nel caso di una band si memorizza solo il nome della stessa e la nazionalità, *Nome* e *Cognome* sono quindi inutilizzati.

Il singolo artista collabora con un producer per album il quale ha il compito di registrare e riarrangiare le canzoni.

Il producer è identificato da:

- Codice fiscale
- Nome
- Cognome

Ogni producer lavora per una specifica etichetta discografica specializzata su uno o più generi musicali.

L'etichetta discografica è composta da:

- Nome
- Sede

Gli eventi giornalieri promossi dal negozio sono descritti da:

- Data
- Sconto
- Numero articoli venduti (esclusiva degli eventi passati)

Particolare attenzione si ha per l'evento corrente che velocizza le operazioni di vendita. L'entità è legata infatti al catalogo e calcola per ogni articolo appartenente all'evento il prezzo scontato per il giorno corrente.

Si memorizzano in un catalogo le seguenti informazioni per ciascun articolo:

- Prezzo
- Quantità
- Prezzo d'acquisto

Nel caso in cui un articolo venga esaurito in negozio viene automaticamente ordinato segnando la quantità richiesta.

Alla vendita degli articoli è prevista l'emissione di uno scontrino che viene memorizzato per accertamenti fiscali.

Gli scontrini sono composti da:

- Data
- Ora

I Clienti sono descritti da:

- E-mail
- Nome
- Numero di telefono

Si tiene traccia solo dei clienti che effettuano una prenotazione.

Il cliente può effettuare una prenotazione di un articolo nel caso in cui non fosse disponibile in negozio. Lascia quindi un acconto che verrà poi detratto dal pagamento, una e-mail e un numero di telefono per contattarlo quando l'ordine sarà arrivato in negozio. Se sono già state prenotate tutte le copie dell'articolo la prenotazione viene rifiutata.

Le prenotazioni sono descritte da:

- Data
- Ora
- Acconto

3. PROGETTAZIONE CONCETTUALE

3.1 Lista Entità

Il Database è formato dalle seguenti tabelle. Tutti i campi sono NOT NULL tranne quelli specificati.

- **Articolo musicale:** rappresenta un prodotto
 - Codice articolo: *string*
 - Nome: *string*
 - Descrizione: *string*
- **Vinile:** rappresenta un prodotto specifico, il vinile
 - Giri: *enum*
 - Condizione copertina: *enum*
 - Condizione disco: *enum*
- **CD:** rappresenta un prodotto specifico, il CD
 - RPM: *int*
 - Formato logico: *enum*
- **Cassetta audio:** rappresenta un prodotto specifico, la cassetta audio
 - Formato: *enum*
 - Tipo nastro: *enum*
- **Artista:** rappresenta un artista o una band
 - Nome d'arte: *string*
 - Nazionalità: *string*
 - Nome: *string* (NULL)
 - Congome: *string* (NULL)
- **Producer:** rappresenta un producer di una etichetta discografica
 - Codice fiscale: *string*
 - Nome: *string*
 - Cognome: *string*
- **Etichetta discografica:** rappresenta l'etichetta discografica che incide l'album
 - Nome: *string*
 - Sede: *string*
- **Genere:** rappresenta il genere musicale dei prodotti in vendita
- **Rock:** rappresenta il genere rock
- **Punk:** rappresenta il genere punk
- **Pop:** rappresenta il genere pop
- **Jazz:** rappresenta il genere jazz
- **Heavy metal:** rappresenta il genere heavy metal
- **Rap:** rappresenta il genere rap
- **Country:** rappresenta il genere country
- **Raggae:** rappresenta il genere raggae
- **Blues:** rappresenta il genere blues
- **Evento:** rappresenta un evento giornaliero organizzato dal negozio
 - Data: *date*
 - Sconto: *double*
 - Numero articoli venduti: *int* (NULL)
- **Evento corrente:** rappresenta l'evento organizzato nella giornata corrente
 - Prezzo scontato: *double*
- **Catalogo:** rappresenta il catalogo di tutti i prodotti in possesso dal negozio
 - Prezzo: *double*
 - Quantità: *int*
 - Prezzo d'acquisto: *double*
- **Catalogo in vendita:** rappresenta il catalogo di tutti i prodotti in esposizione

- **Catalogo in arrivo:** rappresenta il catalogo di tutti i prodotti ordinati che devono arrivare in negozio
 - Quantità: *int*
- **Scontrino:** rappresenta lo scontrino emesso ad ogni vendita
 - Momento: attributo composto da: Data e Ora
 - Prezzo totale: *double*
 - Numero articoli: *int*
- **Cliente:** rappresenta il cliente che ordina dei prodotti
 - E-mail: *string*
 - Nome: *string*
 - Numero telefono: *string*
- **Prenotazione:** rappresenta una prenotazione effettuata da un cliente su degli articoli in arrivo in negozio
 - Momento: attributo composto da: Data e Ora
 - Acconto: *double*

3.2 Lista Relazioni

Articolo musicale-Articolo musicale: bundle (1:N)

- Un articolo musicale può appartenere al più ad un bundle
- Un bundle contiene più articoli musicali

Articolo musicale-Artista: produzione (N:N)

- Un articolo musicale può essere prodotto da zero o più artisti
- Un artista può produrre uno o più album (quindi più articoli musicali)

Artista-Producer: collaborazione (N:N)

- Un artista collabora con uno o più producer per diversi album
- Uno stesso producer può collaborare con più artisti diversi

Producer-Etichetta discografica: registrazione canzone (1:N)

- Un producer registra una canzone presso una sola etichetta discografica
- All'interno della stessa etichetta discografica lavorano uno o più producer

Articolo musicale-Etichetta discografica: incisione (1:N)

- Un album (quindi il corrispettivo articolo musicale) è inciso da una sola etichetta discografica (o da nessuna se si tratta di un bundle)
- Una stessa etichetta discografica incide uno o più album (quindi più articoli musicali)

Etichetta discografica-Genere: occupazione (N:N)

- Ogni etichetta discografica si occupa di uno o più generi
- Lo stesso genere può essere trattato da più etichette discografiche

Articolo musicale-Genere: appartenenza (1:N)

- Un articolo musicale appartiene ad un singolo genere musicale (vincolo)
- Uno stesso genere è associato a uno o più articoli

Evento-Genere: tipologia (1:N)

- Un Evento è specializzato in un singolo evento
- Un genere può specializzare uno o più eventi

Evento corrente-Catalogo: appartenenza (1:1)

- Un evento corrente elenca gli articoli scontati appartenenti al catalogo
- Un articolo del catalogo può appartenere o meno all'evento corrente

Articolo musicale-Catalogo: presenza (1:1)

- Un articolo musicale può essere presente nel catalogo
- Nel catalogo è presente una sola istanza per articolo

Catalogo di Vendita-Scontrino: registro (N:N)

- Un articolo (presente in diverse quantità) può comparire su più scontrini
- Uno scontrino registra la vendita di uno o più articoli

Attributi:

- Quantità: *int*

Scontrino-Cliente: ricezione (1:N)

- Ogni scontrino può essere associato al più ad un cliente che ha effettuato una prenotazione
- Un cliente riceve uno o più scontrini

Prenotazione-Cliente: effettuazione (1:N)

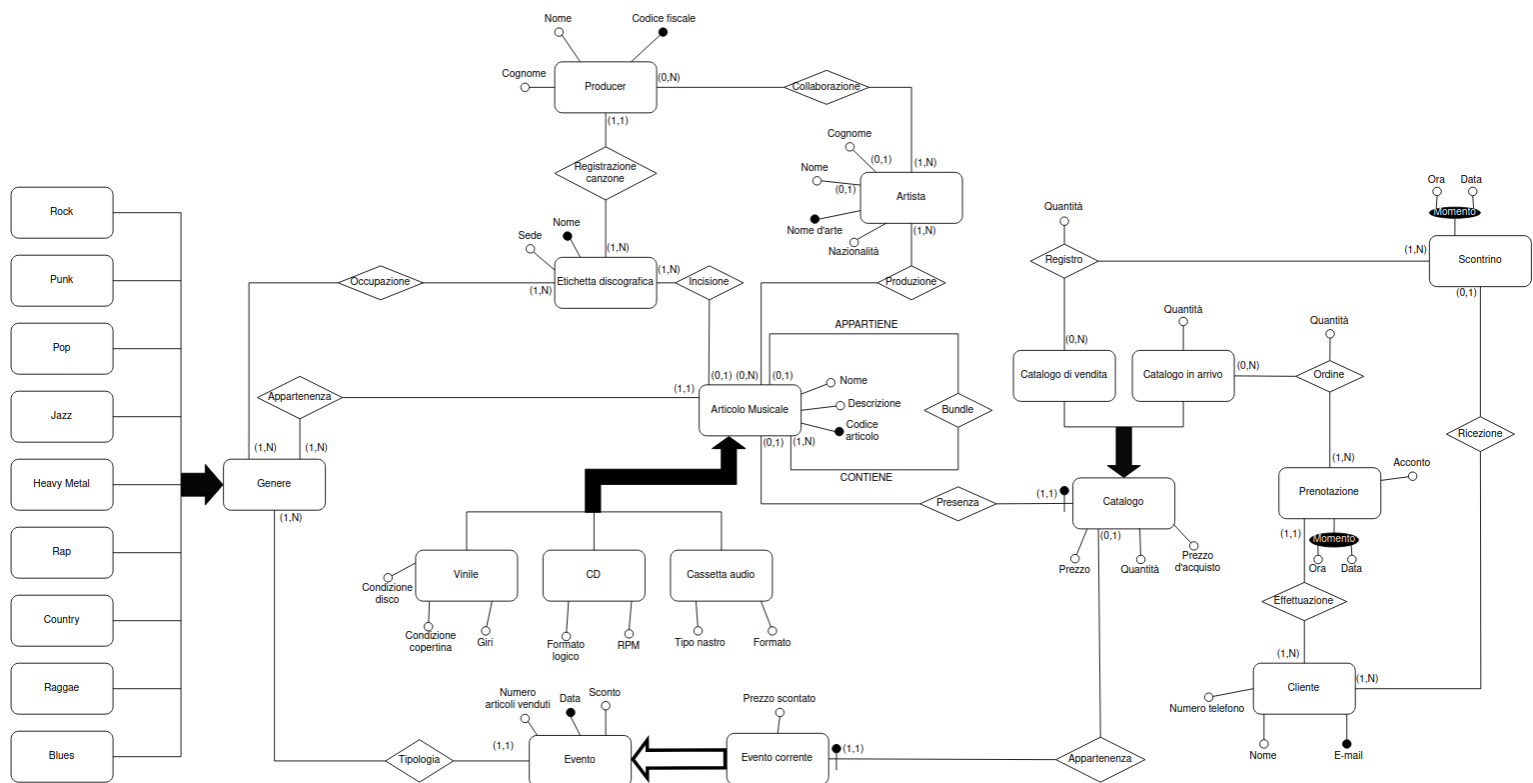
- Ogni prenotazione è effettuata da un solo cliente
- Un cliente può effettuare una o più prenotazioni

Catalogo in arrivo-Prenotazione: ordine (N:N)

- Un articolo (presente in diverse quantità) può essere presente in più prenotazioni
- Una prenotazione prevede l'ordine di uno o più articoli
 - Quantità: *int*

3.3 Lista Generalizzazioni

- **Articolo musicale** è una generalizzazione totale ed esclusiva di **Vinile**, **CD** e **Cassetta audio**
- **Genere** è una generalizzazione totale ed esclusiva di **Rock**, **Punk**, **Pop**, **Jazz**, **Heavy Metal**, **Rap**, **Country**, **Reggae** e **Blues**
- **Catalogo** è una generalizzazione totale ed esclusiva di **Catalogo di vendita** e **Catalogo in arrivo**
- **Evento** è una generalizzazione parziale ed esclusiva di **Evento Corrente**



4. PROGETTAZIONE LOGICA

4.1 Analisi delle ridondanze

Analizzando lo schema E-R si può notare che l'attributo "Prezzo scontato" della relazione "Evento corrente" potrebbe essere calcolato applicando lo sconto all'articolo in vendita se presente nell'evento. Inoltre l'entità "Evento corrente" è essa stessa una ridondanza che va popolata all'inizio di ogni evento e ripulita al termine.

Bisogna quindi analizzare le operazioni riguardanti questo attributo e l'entità per capire se eliminarli.

| Concetto | Costrutto | Volume |
|-----------------|-----------|--------|
| Evento corrente | Entità | 800 |

- **Operazione 1 (1 a settimana):** Creazione della tabella degli articoli appartenenti all'evento corrente
- **Operazione 2 (400 articoli ad evento):** Vendere un articolo scontato dall'evento

Con Ridondanza

- **Operazione 1**

| Concetto | Costrutto | Accesso | Tipo |
|---------------------|-----------|---------|------|
| Evento | Entità | 800 | L |
| Tipologia | Relazione | 800 | L |
| Genere | Entità | 800 | L |
| Appartenenza | Relazione | 800 | L |
| Articolo musicale | Entità | 800 | L |
| Presenza | Relazione | 800 | L |
| Catalogo di vendita | Entità | 800 | L |
| Evento corrente | Entità | 800 | S |

Costo: $800 \times 7 = 5600$ accessi in lettura, $800 \times 2 = 1600$ accessi in scrittura

Costo settimanale: 7200

- **Operazione 2**

| Concetto | Costrutto | Accesso | Tipo |
|---------------------|-----------|---------|------|
| Catalogo di vendita | Entità | 1 | L |
| Appartenenza | Relazione | 1 | L |
| Evento corrente | Entità | 1 | L |
| Registro | Relazione | 1 | S |
| Scontrino | Entità | 1 | S |

Costo: 3 accessi in lettura, $2 \times 2 = 4$ accessi in scrittura

Costo ad evento: $3 \times 400 + 4 \times 400 = 2800$

Senza Ridondanza

- **Operazione 1**

Eliminando la tabella il costo è nullo.

Costo: 0 accessi in lettura, 0 accessi in scrittura

Costo settimanale: 0

- **Operazione 2**

| Concetto | Costrutto | Accesso | Tipo |
|---------------------|-----------|---------|------|
| Catalogo di vendita | Entità | 1 | L |
| Appartenenza | Relazione | 1 | L |
| Articolo musicale | Entità | 1 | L |
| Appartenenza | Relazione | 1 | L |
| Genere | Entità | 1 | L |
| Tipologia | Relazione | 1 | L |
| Evento | Entità | 1 | L |
| Registro | Relazione | 1 | S |
| Scontrino | Entità | 1 | S |

Costo: 7 accessi in lettura, $2 \times 2 = 4$ accessi in scrittura

Costo ad evento: $7 \times 400 + 4 \times 400 = 4400$

Considerando le due operazioni risulta una stima di accessi pari a 10000 se si mantiene l'entità "Evento corrente", mentre eliminandola si otterrebbe una stima di accessi pari a 4400. Risulta quindi conveniente eliminare l'entità e valutare se l'articolo che si sta vendendo è presente nell'evento tramite il confronto tra le entità e in caso applicare lo sconto.

Analizzando lo schema E-R si può notare che l'attributo "Numero articoli venduti" della relazione "Evento passato" potrebbe essere calcolato come somma degli articoli venduti (cioè presenti negli scontrini emessi) durante l'evento.

Bisogna quindi analizzare le operazioni riguardanti questo attributo per capire se eliminarlo.

- **Operazione 1 (400 articoli ad evento):** Memorizzare la vendita di un articolo scontato dall'evento
- **Operazione 2 (2 a settimana):** Visualizza il numero di articoli venduti per un evento passato (in media si emettono 200 scontrini ciascuno contenente 2 articoli scontati dall'evento)

Con Ridondanza

- **Operazione 1**

| Concetto | Costrutto | Accesso | Tipo |
|---------------------|-----------|---------|------|
| Catalogo di vendita | Entità | 1 | L |
| Presenza | Relazione | 1 | L |

| | | | |
|---------------------|-----------|---|---|
| Articolo musicale | Entità | 1 | L |
| Appartenenza | Relazione | 1 | L |
| Genere | Entità | 1 | L |
| Tipologia | Relazione | 1 | L |
| Evento | Entità | 1 | L |
| Evento | Entità | 1 | S |
| Catalogo di vendita | Entità | 1 | S |
| Registro | Relazione | 1 | S |
| Scontrino | Entità | 1 | S |

Costo: $400 \times 7 = 2800$ accessi in lettura, $400 \times 2 \times 4 = 3200$ accessi in scrittura

- **Operazione 2**

| Concetto | Costrutto | Accesso | Tipo |
|----------|-----------|---------|------|
| Evento | Entità | 1 | L |

Costo: 1 accesso in lettura

Costo settimanale: $1 \times 2 = 2$

Senza Ridondanza

- **Operazione 1**

| Concetto | Costrutto | Accesso | Tipo |
|---------------------|-----------|---------|------|
| Catalogo di vendita | Entità | 1 | L |
| Presenza | Relazione | 1 | L |
| Articolo musicale | Entità | 1 | L |
| Appartenenza | Relazione | 1 | L |
| Genere | Entità | 1 | L |
| Tipologia | Relazione | 1 | L |
| Evento | Entità | 1 | L |
| Catalogo di vendita | Entità | 1 | S |
| Registro | Relazione | 1 | S |
| Scontrino | Entità | 1 | S |

Costo: $400 \times 7 = 2800$ accessi in lettura, $400 \times 2 \times 3 = 2400$ accessi in

scrittura

- **Operazione 2**

| Concetto | Costrutto | Accesso | Tipo |
|---------------------|-----------|---------|------|
| Scontrino | Entità | 200 | L |
| Registro | Relazione | 400 | L |
| Catalogo di vendita | Entità | 400 | L |
| Presenza | Relazione | 400 | L |
| Articolo Musicale | Entità | 400 | L |
| Appartenenza | Relazione | 400 | L |
| Genere | Entità | 400 | L |
| Tipologia | Relazione | 400 | L |
| Evento | Entità | 400 | L |

Costo: $200 + 400 \times 8 = 3400$ accessi in lettura

Costo settimanale: $3400 \times 2 = 6800$

Considerando le due operazioni risulta una stima di accessi pari a 6002 se si mantiene la ridondanza di "Numero articoli venduti", mentre eliminandola si otterrebbe una stima di accessi pari a 12000. Risulta quindi conveniente mantenere l'attributo sulla relazione "Evento".

4.2 Eliminazioni delle generalizzazioni

Genere: È una generalizzazione totale ed esclusiva legata a 3 relazioni: molti a molti con "Etichetta discografica", 1 a molti con "Articolo musicale", 1 a molti con "Evento". Dal momento che tutte le entità figlie non hanno attributi propri né relazioni si è deciso di accorpare le classi figlie nella classe padre inserendo un attributo "Tipo".

Articolo Musicale: È una generalizzazione totale ed esclusiva legata a 7 relazioni:

- 1 a molti con "Etichetta discografica"
- molti a molti con "Artista"
- 1 a molti con "Articolo musicale"
- 1 a 1 con "Catalogo"
- molti a molti con "evento"
- 1 a molti con "Genere" tramite la relazione "Evento passato"
- 1 a molti con "Genere" tramite la relazione "Evento futuro"

Poiché le entità figlie hanno molti attributi propri si è deciso di creare delle relazioni di cardinalità 1 a 1 tra queste e l'entità padre. Questa soluzione permette di evitare valori NULL e di creare bundle contenenti articoli di diversa tipologia.

Catalogo: È una generalizzazione totale ed esclusiva legata con una relazione 1 a 1 con "Articolo musicale". L'entità figlia "Catalogo di vendita" è legata con una relazione molti a molti con "Scontrino" mentre la seconda entità figlia "Catalogo in arrivo" è legata con una relazione molti a molti con "Prenotazione". Poiché le entità figlie non posseggono attributi propri si è deciso di accorparle nell'entità padre; così facendo inoltre si preserva l'identificatore esterno di "Catalogo". La quantità degli articoli ordinati dal negozio viene rinominata in "Quantità in arrivo".

Evento: È una generalizzazione parziale ed esclusiva legata con una relazione 1 a molti a “Genere”, tuttavia in seguito all’analisi delle ridondanze è risultato conveniente eliminare l’entità “Evento corrente”.

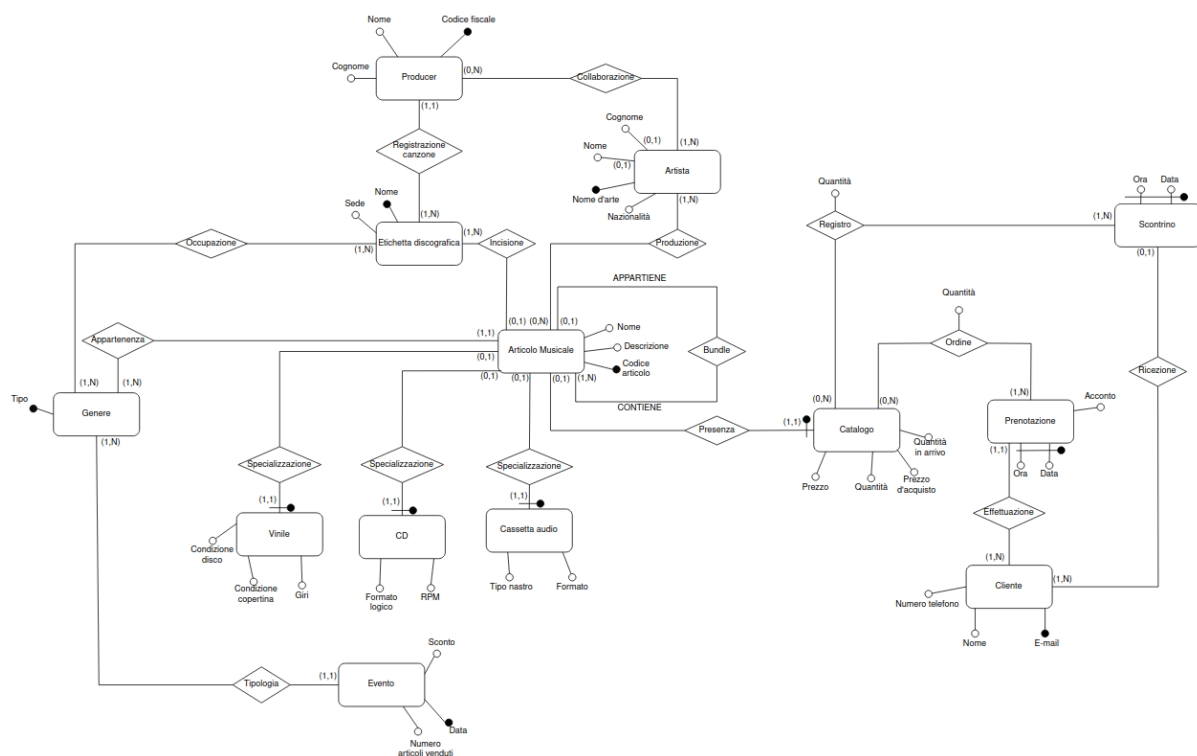
4.3 Partizionamento/Accorpamento di Entità e Relationship

Le entità Prenotazione e Scontrino possiedono entrambe di un attributo composto “Momento” i quali devono essere separati in due attributi singoli “Data” e “Ora”.

4.4 Scelta di identificatori primari

La decomposizione di “Momento” porta alla creazione di nuovi identificatori costituiti da “Data” e “Ora” sia nell’entità “Scontrino” che in “Prenotazione”.

4.5 Diagramma ER ristrutturato



4.6 Descrizione schema relazione

Si ricorda che gli attributi contrassegnati da “*” possono essere NULL.

ArticoloMusicale(CodiceArticolo, Nome, Descrizione, Genere, EtichettaDiscografica*, CodiceBundle*)

Vinile(ArticoloMusicale, Giri, CondizioneDisco, CondizioneCopertina)

CD(ArticoloMusicale, RPM, FormatoLogico)

CassettaAudio(ArticoloMusicale, Formato, TipoNastro)

Produzione(ArticoloMusicale, Artista)

Artista(NomeArte, Nome*, Cognome*, Nazionalità)

Collaborazione(Artista, Producer)

Producer(CodiceFiscale, Nome, Cognome, EtichettaDiscografica)

EtichettaDiscografica(Nome, Sede)

Occupazione(EtichettaDiscografica, Genere)

Genere(Tipo)

Evento(Data, Genere, Sconto, NumeroArticoliVenduti*)

Catalogo(ArticoloMusicale, Prezzo, Quantità, PrezzoAcquisto, Quantità in arrivo)

Registro(DataScontrino, OraScontrino, Catalogo, Quantità)

Scontrino(Data, Ora, Cliente*)

Cliente(Email, Nome, NumeroTelefono)

Prenotazione(Data, Ora, Cliente, Acconto)

Ordine(Data, Ora, Catalogo, Quantità)

4.7 Vincoli di integrità referenziale

ArticoloMusicale.CodiceBuddle -> ArticoloMusicale.CodiceArticolo
ArticoloMusicale.EtichettaDiscografica -> EtichettaDiscografica.Nome
Vinile.ArticoloMusicale -> ArticoloMusicale.CodiceArticolo
CD.ArticoloMusicale -> ArticoloMusicale.CodiceArticolo
CassettaAudio.ArticoloMusicale -> ArticoloMusicale.CodiceArticolo
Produzione.ArticoloMusicale -> ArticoloMusicale.CodiceArticolo
Produzione.Artista -> Artista.NomeArte
Collaborazione.Artista -> Artista.NomeArte
Collaborazione.Producer -> Producer.CodiceFiscale
Producer.EtichettaDiscografica -> EtichettaDiscografica.Nome
Occupazione.EtichettaDiscografica -> EtichettaDiscografica.Nome
Occupazione.Genere -> Genere.Tipo
Evento.Genere -> Genere.Tipo
Catalogo.ArticoloMusicale -> ArticoloMusicale.CodiceArticolo
(Registro.DataScontrino, Registro.OraScontrino) -> (Scontrino.Data, Scontrino.Ora)
Registro.Catalogo -> Catalogo.ArticoloMusicale
Scontrino.Cliente -> Cliente.Email
Prenotazione.Cliente -> Cliente.Email
(Ordine.Data, Ordine.Ora) -> (Prenotazione.Data, Prenotazione.Ora)
Ordine.Catalogo -> Catalogo.ArticoloMusicale

4.8 Vincoli non esprimibili in ER

- Un articolo musicale o è presente in una sola delle vecchie entità figlie (Vinile, CD, Cassetta audio) o in nessuna perché è un bundle
- Un articolo racchiuso in un bundle non è vendibile singolarmente, quindi non è presente nel catalogo
- I bundle non sono incisi da nessuna etichetta discografica e non sono stati prodotti da nessun artista, gli altri articoli invece possiedono entrambi
- I bundle non possono contenere altri bundle, quindi l'attributo "codice bundle" è sempre NULL
- "Nome" e "Cognome" dell'entità "Artista" sono entrambi NULL se si rappresenta una band oppure entrambi assumono un valore se si rappresenta un cantante
- Il genere di un articolo deve corrispondere a un genere dell'etichetta discografica che lo ha prodotto. Inoltre un artista deve collaborare con un producer che lavora per una etichetta discografica che tratta il genere musicale dell'album da incidere (coerenza tra i valori in tutte le relazioni citate)
- Gli scontrini che non sono associati a nessun cliente sono stati emessi a seguito di una vendita conclusa in negozio, quelli invece assegnati ad una e-mail sono stati emessi a seguito di una prenotazione conclusa
- Il numero di ordini riguardo un articolo derivanti dalle prenotazioni dei clienti non deve superare la quantità in arrivo dell'articolo nel catalogo (la richiesta di prenotazione del cliente viene rifiutata perché le copie dell'articolo in arrivo sono già state tutte prenotate)
- L'acconto di ciascuna prenotazione non può superare la somma dei prezzi degli articoli ordinati.

Glossario nomi


| Precedente | Nuovo | Descrizione |
|--------------------|----------------|---|
| Nome d'arte | NomeArte | Non è permesso l'uso dell'apostrofo |
| Prezzo d'acquisto | PrezzoAcquisto | Non è permesso l'uso dell'apostrofo |
| Quantità in arrivo | QuantitaArrivo | Maggiore leggibilità |
| Occupazione | Settore | Rendere più significativo il nome della tabella |

5. SQL

1. L'artista che ha venduto di più (in termini di prezzi degli articoli venduti) nel mese di febbraio dell'anno corrente.

```
drop view if exists venditearticolo;
create view venditearticolo (articolomusicale, somma) as
select pa.articolomusicale, sum(pa.prezzotot) as somma -- prezzi delle vendite di febbraio per articolo
from (select cat.articolomusicale, reg.quantita * cat.prezzo as prezzotot -- prezzi per vendita di ogni articolo
      from (select * -- vendite di febbraio
            from scontrino as s
            where s.data >= '2024-02-01' and s.data <= '2024-02-29') as vf, registro as reg, catalogo as cat
      where vf.data = reg.datascontrino and vf.ora = reg.orascontrino and reg.catalogo = cat.articolomusicale) as pa
group by pa.articolomusicale;

select prod.artista
from (select va.articolomusicale, va.somma -- articolo musicale con somma massima
      from venditearticolo as va
      where va.somma = (select max(somma) from venditearticolo)) as art, produzione as prod
where art.articolomusicale = prod.articolomusicale
```

| | |
|---|---|
| | artista character varying (30)  |
| 1 | Eminem |

2. Gli articoli (codice, descrizione, etichetta discografica, codice bundle) che appartengono al prossimo evento in modo che si visionino prima tutti i bundle.

```
select codicearticolo, descrizione, etichettadiscografica, codicebundle
from (select *
      from evento
      where data = (select min(data) from evento where numeroarticolivenduti is null)) as prossimoevento
join articolomusicale on prossimoevento.genere = articolomusicale.genere
order by etichettadiscografica desc
```

| | codicearticolo [PK] character | descrizione character varying (200) | etichettadiscografica character varying (40) | codicebundle character |
|---|----------------------------------|--|---|---------------------------|
| 1 | LN234581 | Descrizione di un bellissimo bundle fatto su misura per te | [null] | [null] |
| 2 | HJ456792 | Descrizione di un bellissimo articolo musicale da comprare assolutame... | Sub Pop Records | LN234581 |
| 3 | ST012345 | Descrizione di un bellissimo articolo musicale da comprare assolutame... | Sub Pop Records | [null] |
| 4 | BA890147 | Descrizione di un bellissimo articolo musicale da comprare assolutame... | Sub Pop Records | [null] |
| 5 | JL012369 | Descrizione di un bellissimo articolo musicale da comprare assolutame... | Def Jam Recordings | LN234581 |
| 6 | LM890147 | Descrizione di un bellissimo articolo musicale da comprare assolutame... | Capitol Records | [null] |
| 7 | JI234581 | Descrizione di un bellissimo articolo musicale da comprare assolutame... | Atlantic Records | [null] |

3. Le etichette discografiche ordinate in modo decrescente secondo il numero di vinili incisi.

```
select ed.nome, count(*) as numerovinili
from vinile as vin, articolomusicale as am, etichettadiscografica as ed
where vin.articolomusicale = am.codicearticolo and am.etichettadiscografica = ed.nome
group by ed.nome
order by numerovinili desc
```

| | nome [PK] character varying (40) | numerovinili bigint | | | |
|---|-------------------------------------|------------------------|----|------------------|---|
| 1 | Sub Pop Records | 5 | 10 | XL Recordings | 2 |
| 2 | Interscope Records | 5 | 11 | Merge Records | 1 |
| 3 | Island Records | 3 | 12 | Matador Records | 1 |
| 4 | Domino Recording Company | 2 | 13 | Atlantic Records | 1 |
| 5 | Nonesuch Records | 2 | 14 | Dead Oceans | 1 |
| 6 | Epic Records | 2 | 15 | Columbia Records | 1 |
| 7 | Def Jam Recordings | 2 | 16 | ANTI-Records | 1 |
| 8 | Capitol Records | 2 | 17 | Ninja Tune | 1 |
| 9 | Universal Music Group | 2 | | | |

4. Le e-mail dei clienti che hanno effettuato delle prenotazioni per una spesa totale superiore a 100 euro; ordinare in modo decrescente secondo la spesa.

```
select sco.cliente, sum(cat.prezzo) spesatotale
from scontrino as sco, registro as reg, catalogo as cat
where sco.data = reg.datascontrino and sco.ora = reg.orascontrino and reg.catalogo = cat.articolomusicale and
sco.cliente is not null
group by sco.cliente
having sum(cat.prezzo) > 100
order by spesatotale desc
```

| | cliente character varying (40) | spesatotale numeric |
|---|-----------------------------------|------------------------|
| 1 | kristen85@example.com | 198.25 |
| 2 | unovak@example.net | 152.30 |
| 3 | brandyrusso@example.com | 126.30 |
| 4 | rachelcoleman@example.net | 106.90 |
| 5 | angel06@example.com | 106.00 |

5. Il producer e l'etichetta discografica che hanno collaborato alla produzione del CD "Golden Hour".

```
select producer.nome, producer.cognome, cdsel.etichettadiscografica
from (select *
      from articolomusicale as am join cd on am.codicearticolo = cd.articolomusicale
      where am.nome = 'Golden Hour') as cdsel, produzione, artista as art, collaborazione as collab, producer
where cdsel.codicearticolo = produzione.articolomusicale and produzione.artista = art.nomearte and
      art.nomearte = collab.artista and collab.producer = producer.codicefiscale
```

| | nome character varying (15) | cognome character varying (15) | etichettadiscografica character varying (40) |
|---|--------------------------------|-----------------------------------|---|
| 1 | Sophia | Williams | Nonesuch Records |

6. Il numero di degli articoli venduti per genere musicale ordinati in modo decrescente.

```
select gen.tipo as genere, count(*) as quantita
from scontrino as sco, registro as reg, catalogo as cat, articolomusicale as am, genere as gen
where sco.data = reg.datascontrino and sco.ora = reg.orascontrino and reg.catalogo = cat.articolomusicale and
      cat.articolomusicale = am.codicearticolo and am.genere = gen.tipo
group by gen.tipo
order by quantita desc
```

| | genere character varying (12) | quantita bigint |
|---|----------------------------------|--------------------|
| 1 | Rock | 412 |
| 2 | Punk | 135 |
| 3 | Pop | 82 |
| 4 | Heavy Metal | 75 |
| 5 | Blues | 49 |
| 6 | Rap | 46 |
| 7 | Jazz | 41 |
| 8 | Country | 18 |
| 9 | Raggae | 4 |

5.1 Indici

Sono stati realizzati i seguenti indici per velocizzare le query 1 e 4: il primo è in forma di BTree per la ricerca della data compresa tra due valori nella query 1, gli altri sono invece in forma di Hash per i vari join tra le tabelle "Scontrino", "Registro" e "Catalogo" che contengono numerose tuple.

```
drop index if exists data_scontrino_idx_btree;
drop index if exists ora_scontrino_idx_hash;
drop index if exists data_registro_idx_hash;
drop index if exists ora_registro_idx_hash;
drop index if exists catalogo_registro_idx_hash;
drop index if exists articolomusicale_catalogo_idx_hash;

create index data_scontrino_idx_btree on scontrino(data);
create index ora_scontrino_idx_hash on scontrino using hash(ora);
create index data_registro_idx_hash on registro using hash(datascontrino);
create index ora_registro_idx_hash on registro using hash(orascontrino);
create index catalogo_registro_idx_hash on registro using hash(catalogo);
create index articolomusicale_catalogo_idx_hash on catalogo using hash(articolomusicale);
```

6. Software

Per utilizzare il software per l'esecuzione delle query è necessario seguire i seguenti passaggi:

- Scaricare la cartella *software*
- Eseguire il comando *make* per compilare tutti i file .c oppure eseguire manualmente il seguente comando

```
gcc -o file.out file.c dependencies/dbconfig.c dependencies/dbutils.c -I dependencies/include -lpq
```

- Per successivamente eseguire la una determinata query eseguire il comando

```
./query<numero_query>.out <host> <username> <DB> <port> <password>
```

un esempio potrebbe essere:

```
./query1.out localhost postgres buddymusic 5432 music
```