



# RELAZIONE PROGETTO BDSI

A.A. 2020-21

Autori:

Arturo Batistoni (7024683)  
Alberto Biliotti (7026939)

- Realizzato e testato in ambiente Windows 10 e MacOS 11
- Versione MySQL Workbench: 8.022

# Indice

<b>RICHIESTA .....</b>	<b>2</b>
<b>PROGETTAZIONE CONCETTUALE.....</b>	<b>3</b>
ANALISI DELLA RICHIESTA .....	3
<i>Analisi dei singoli termini e costruzione del loro glossario.....</i>	<i>3</i>
COSTRUZIONE DELLO SCHEMA CONCETTUALE .....	5
<b>PROGETTAZIONE LOGICA .....</b>	<b>8</b>
ANALISI DELLE PRESTAZIONI DELLO SCHEMA CONCETTUALE .....	8
<i>Tabella dei volumi.....</i>	<i>8</i>
<i>Tabella delle operazioni.....</i>	<i>8</i>
<i>Tabella degli accessi.....</i>	<i>9</i>
RISTRUTTURAZIONE DELLO SCHEMA E-R.....	10
<i>Analisi delle ridondanze.....</i>	<i>10</i>
<i>Eliminazione delle generalizzazioni.....</i>	<i>10</i>
<i>Partizionamento/accorpamento di concetti.....</i>	<i>11</i>
<i>Eliminazione degli attributi multi-valore.....</i>	<i>12</i>
<i>Scelta degli identificatori principali.....</i>	<i>12</i>
<i>Risultato finale .....</i>	<i>12</i>
TRADUZIONE VERSO IL MODELLO RELAZIONALE.....	13
<b>IMPLEMENTAZIONE IN MYSQL.....</b>	<b>14</b>
CREAZIONE E POPOLAMENTO TABELLE .....	14
INTERROGAZIONI.....	14
PROCEDURE E FUNZIONI .....	14
VISTE.....	15
TRIGGER.....	15

## **Richiesta**

Una piccola università richiede una base di dati per gestire le risorse del proprio sistema bibliotecario. Queste possono essere libri, tesi o articoli su rivista. Per ogni risorsa vogliamo sapere la collocazione precisa (sala) in ogni biblioteca, gli autori, la data di pubblicazione, il titolo, il genere e la sua disponibilità; ogni risorsa sarà inoltre caratterizzata da un codice numerico identificativo e le risorse fisiche avranno un barcode univoco. Per ogni biblioteca si richiedono il nome ed il dipartimento di appartenenza. Vogliamo inoltre memorizzare i dati delle persone di interesse, che possono essere autori o dipendenti, e in particolare vogliamo conoscerne nome e cognome; ogni persona sarà inoltre identificata da un ID univoco e per i dipendenti vogliamo conoscere anche recapito e codice fiscale. Ogni biblioteca avrà inoltre un responsabile specificato tra i dipendenti. Per ogni dipendente vogliamo tenere traccia delle attività, che potrebbero coinvolgere risorse, a cui partecipa o a cui ha partecipato e che sono terminate (in questo caso vogliamo sapere anche la data di fine dell'attività) e per ognuna di queste attività vogliamo conoscere il nome, la data di inizio e la scadenza prevista. La base di dati dovrà essere utilizzata sia dalle biblioteche che dagli utenti.

# Progettazione concettuale

## Analisi della richiesta

Procediamo ad evidenziare i termini “chiave” e i termini ad essi relativi nella specifica:

Una piccola università richiede una base di dati per gestire le risorse del proprio sistema bibliotecario. Queste possono essere **libri, tesi o articoli su rivista**. Per ogni **risorsa** vogliamo sapere la **collocazione** precisa (sala) in ogni **biblioteca**, gli **autori**, la **data di pubblicazione**, il **titolo**, il **genere** e la sua **disponibilità**; ogni risorsa sarà inoltre caratterizzata da un **codice numerico identificativo** e le risorse fisiche avranno un **barcode** univoco. Per ogni **biblioteca** si richiedono il **nome** ed il **dipartimento** di appartenenza. Vogliamo inoltre memorizzare i dati delle **persone** di interesse, che possono essere **autori** o **dipendenti**, e in particolare vogliamo conoscerne **nome** e **cognome**; ogni persona sarà inoltre identificata da un **ID** univoco e per i dipendenti vogliamo conoscere anche **recapito** e **codice fiscale**. Ogni **biblioteca** avrà inoltre un **responsabile** specificato tra i dipendenti. Per ogni **dipendente** vogliamo tenere traccia delle **attività**, che potrebbero coinvolgere **risorse**, a cui **partecipa** o a cui **ha partecipato** e che sono terminate (in questo caso vogliamo sapere anche la **data di fine** dell'attività) e per ognuna di queste **attività** vogliamo conoscere il **nome**, la **data di inizio** e la **scadenza prevista**. La base di dati dovrà essere utilizzata sia dalle biblioteche che dagli utenti.

## Analisi dei singoli termini e costruzione del loro glossario

Dopo aver evidenziato i termini chiave e quelli ad essi relativi, procediamo a costruire un glossario dei primi:

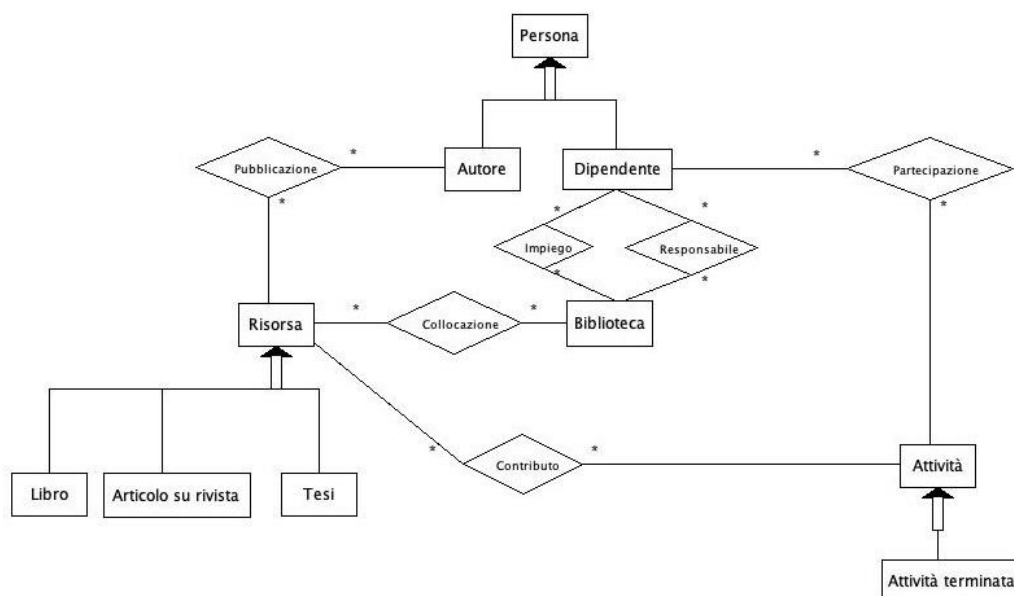
Nome	Descrizione	Termini relativi	Collegamenti
<i>Risorsa</i>	Pubblicazione custodita in una biblioteca. Può essere un libro, una tesi o un articolo su rivista	Collocazione (sala), data di pubblicazione, titolo, genere, ID, barcode, libro, tesi, articolo su rivista	Biblioteca, Attività, Autori
<i>Biblioteca</i>	Custodisce un certo numero di risorse	Nome, dipartimento	Risorsa, Dipendente, Responsabile
<i>Persona</i>	Persona di interesse nel contesto di questa base di dati. Può trattarsi di un dipendente o di un autore.	Nome, cognome, ID	Autore, Dipendente
<i>Autore</i>	Persona che ha contribuito alla pubblicazione di almeno una risorsa	Gli stessi di Persona	Risorsa, Persona
<i>Dipendente</i>	Persona che lavora in una biblioteca.	Gli stessi di Persona, recapito, codice fiscale	Biblioteca, Persona, Attività

	Può esserne il responsabile.		
<i>Responsabile</i>	Dipendente responsabile di una determinata biblioteca.	Gli stessi di Dipendente	Gli stessi di Dipendente
<i>Attività</i>	Attività nell'ambito del sistema bibliotecario a cui possono prendere parte i dipendenti e che può coinvolgere delle risorse. Può essere in corso oppure terminata	Nome, data di inizio, scadenza prevista	Attività terminata, data di fine
<i>Attività terminata</i>	Attività che è stata portata a termine	Gli stessi di Attività, data di fine	Gli stessi di Attività

Possiamo dunque procedere alla costruzione di uno schema concettuale E-R.

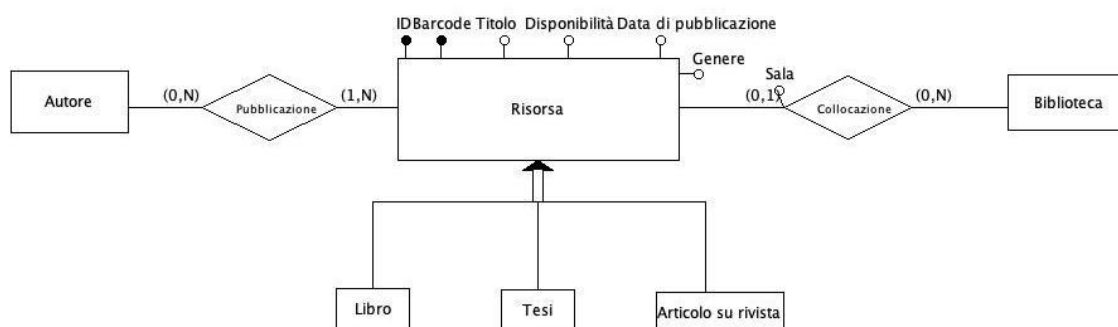
## Costruzione dello schema concettuale

Per la costruzione dello schema E-R adottiamo una strategia mista, partendo dallo schema scheletro sotto riportato che collega i tre macro-concetti della richiesta, ossia Risorsa, Biblioteca e Attività; si è deciso in particolare di tradurre tutte le parole chiave sopra riportate in entità, fatta eccezione di Responsabile, che è stato tradotto in una associazione tra Biblioteca e Dipendente, poiché un Responsabile è un Dipendente. Tutte le altre relazioni seguono abbastanza banalmente dalla specifica. Non ci siamo preoccupati in questa prima fase di progettazione a livello molto astratto degli attributi delle entità e delle cardinalità delle relazioni.



Si sono considerati in seguito i tre macro-concetti della richiesta singolarmente, procedendo alla creazione di tre schemi concettuali indipendenti, sulla base delle informazioni fornite dalla specifica e analizzate e riorganizzate nel glossario precedentemente.

Il seguente è il frammento di schema per Risorsa:



Di seguito è riportato quello per Biblioteca:



Si segnala che anche qui la cardinalità di Recapito è  $(0, N)$ . Per compattezza, inoltre, alcuni nomi di attributi sono stati abbreviati, senza cambiarne comunque il senso. Più avanti, nella fase di traduzione e implementazione, sarà fatto riferimento ai nomi più lunghi indicati precedentemente.



# Progettazione logica

## Analisi delle prestazioni dello schema concettuale

### Tabella dei volumi

Si riporta di seguito la tabella dei volumi fornita dalla specifica:

Concetto	Tipo	Volume
Risorsa	E	100.000
Libro	E	40.000
Tesi	E	5.000
Articolo su rivista	E	55.000
Collocazione	R	90.000
Biblioteca	E	5
Persona	E	200.000
Dipendente	E	100
Autore	E	199.900
Responsabile	R	5
Impiego	R	100
Attività	E	500
Attività terminata	E	450
Partecipazione	R	100
Contributo	R	90.000

Per quanto riguarda il volume di Contributo, si stima che la metà delle attività coinvolge risorse. Di queste, circa un terzo ne coinvolge migliaia (ad esempio, trasferimenti da altre biblioteche o donazioni di vaste collezioni), un terzo ne coinvolge centinaia (p.e. il controllo delle consistenze delle collocazioni in una sala) e un terzo decine (p.e. la pulizia di libri antichi). Otteniamo dunque:

$$\left(\frac{1}{3} * 1000 + \frac{1}{3} * 100 + \frac{1}{3} * 10\right) * \frac{500}{2} \approx 90000$$

### Tabella delle operazioni

Si riportano di seguito le operazioni più frequenti e onerose per il sistema che, secondo la cosiddetta regola “ottanta-venti”, avranno il maggiore impatto sulle sue prestazioni.

- Operazione 1: aggiornare l’elenco delle risorse custodite da una biblioteca, circa 2 volte al giorno.
- Operazione 2: ricercare la disponibilità e la collocazione di una risorsa, circa 500 volte al giorno.
- Operazione 3: aggiornare l’elenco delle attività dei dipendenti e delle relative risorse coinvolte, circa 10 volte a settimana.

Inseriamo dunque le informazioni rilevanti nella seguente tabella delle operazioni.

<b>Operazione</b>	<b>Tipo</b>	<b>Frequenza</b>
Operazione 1	B	2 volte al giorno
Operazione 2	I	500 volte al giorno
Operazione 3	B	10 volte a settimana

### **Tabella degli accessi**

Si riportano di seguito le tabelle degli accessi, che sono state costruite cercando di tenere sempre in considerazione il caso peggiore.

#### *Operazione 1*

Si suppone che l'operazione coinvolga circa 10 nuove risorse alla volta.

<b>Concetto</b>	<b>Costrutto</b>	<b>Accessi</b>	<b>Tipo</b>
Risorsa	E	10	S
Collocazione	R	10	S
Pubblicazione	R	20	S
Autore	E	20	S

#### *Operazione 2*

<b>Concetto</b>	<b>Costrutto</b>	<b>Accessi</b>	<b>Tipo</b>
Risorsa	E	1	L
Collocazione	R	1	L

#### *Operazione 3*

Si tiene conto che il rapporto tra i volumi di Contributo e Attività è 90.000 a 500, ossia 180 a 1.

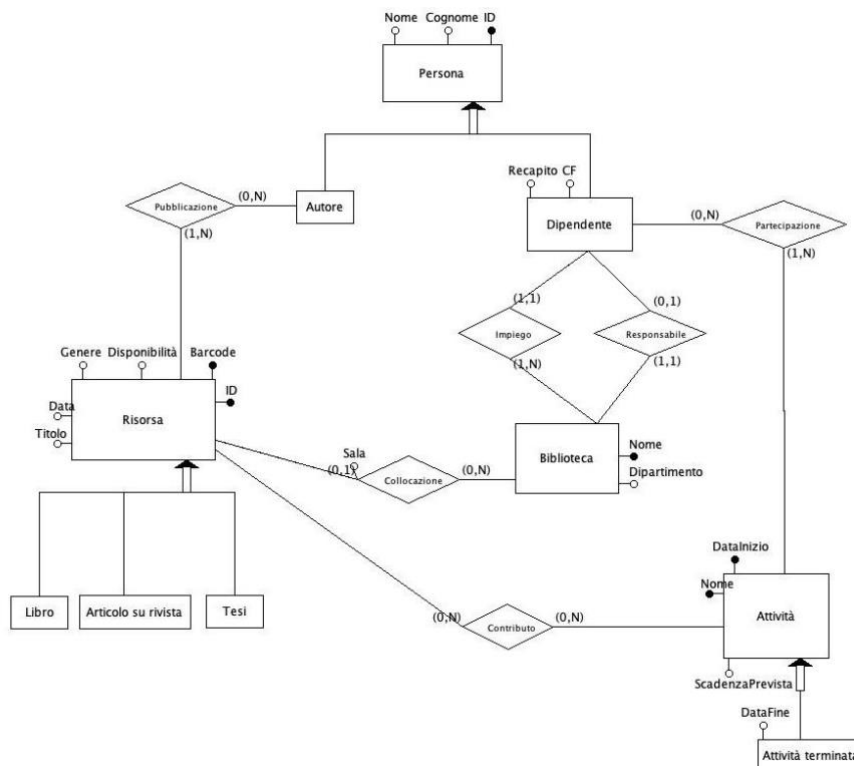
<b>Concetto</b>	<b>Costrutto</b>	<b>Accessi</b>	<b>Tipo</b>
Attività	E	3	S
Partecipazione	R	6	S
Contributo	R	540	S

Si ritiene opportuno dare la priorità alla velocità di esecuzione della seconda operazione, in quanto interattiva, anche a costo di rallentare leggermente l'esecuzione delle altre operazioni.

Possiamo a questo punto procedere alla ristrutturazione dello schema E-R, per prepararlo alla traduzione.

## Ristrutturazione dello schema E-R

Si riporta di nuovo per comodità lo schema ottenuto dalla fase di progettazione concettuale:



### Analisi delle ridondanze

Non si rilevano nello schema informazioni che possono essere derivate da altre.

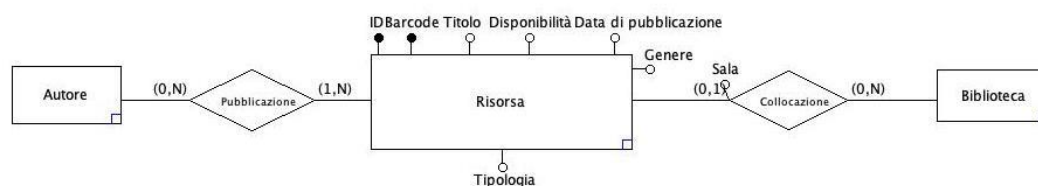
### Eliminazione delle generalizzazioni

Nello schema E-R sono presenti tre generalizzazioni, che andiamo a esaminare una ad una di seguito.

#### Risorsa

Si è deciso di accorpare le entità figlie nell'entità genitore, e di aggiungere un attributo Tipologia a Risorsa. Infatti, ai fini delle operazioni che saranno principalmente eseguite, la tipologia della risorsa non è utile a sufficienza da giustificare l'overhead che sarebbe introdotto dalla sostituzione della generalizzazione con delle associazioni, ma può comunque essere utile per gli utenti essere a conoscenza della tipologia di una risorsa.

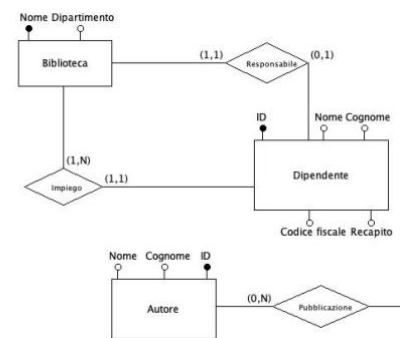
Il risultato è riportato nel seguente frammento dello schema E-R:



## Persona

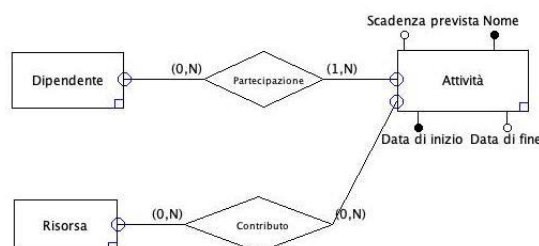
In questo caso si ritiene opportuno accorpare l'entità genitore nelle entità figlie, poiché queste sono prevalentemente scorrelate e si prevede che l'eventualità che una Persona sia Dipendente e Autore sia molto rara.

Riportiamo a destra il risultato:



## Attività

Si procede con l'accorpamento dell'entità figlia nell'entità genitore, dato che le operazioni non fanno distinzioni tra le due. All'entità Attività viene inoltre aggiunto un attributo opzionale "Data di fine" con cardinalità (0,1). Il risultato è riportato di seguito: si segnala che anche in questo caso dallo schema la cardinalità dell'attributo non è visibile.



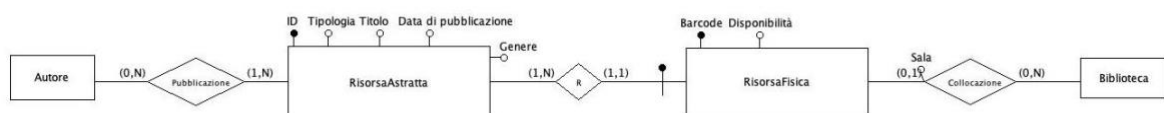
## Partizionamento/accorpamento di concetti

### Risorsa → RisorsaAstratta, RisorsaFisica

Si è ritenuto opportuno partizionare l'entità Risorsa in due entità RisorsaAstratta e RisorsaFisica, effettuando un partizionamento verticale. Si separa dunque il concetto di una pubblicazione (caratterizzata da ID, Titolo, Autore, Genere, Data di pubblicazione e Tipologia) da quello della sua copia fisica (caratterizzata dall'ID della pubblicazione, da un barcode e da una collocazione).

Questo partizionamento permette di facilitare e velocizzare l'accesso alle risorse fisiche da parte degli utenti, rendendo più veloce l'esecuzione dell'operazione 2. Inoltre, vengono eliminate le ridondanze e le potenziali anomalie degli attributi che avremmo altrimenti avuto nelle istanze di Risorsa, con la ripetizione dei campi di RisorsaAstratta per tutte le copie fisiche della pubblicazione.

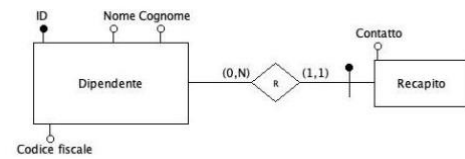
Si riporta di seguito il risultato:



## Eliminazione degli attributi multi-valore

### Recapito

Si è deciso di sostituire l'attributo multi-valore Recapito con una associazione uno-a-molti tra Dipendente e una nuova entità Recapito con un attributo Contatto, che avrà come identificatore esterno la sua partecipazione a questa associazione.

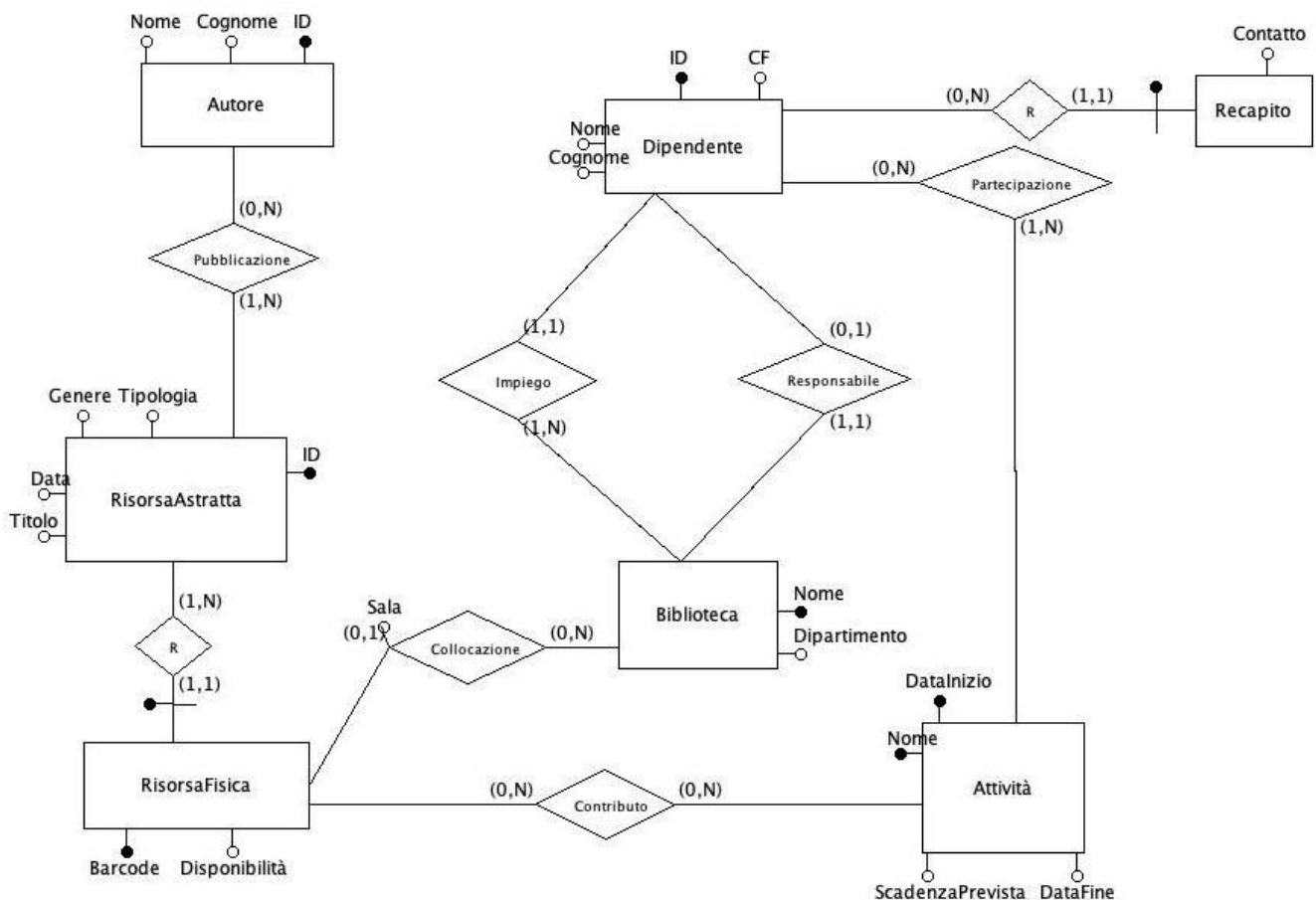


### Scelta degli identificatori principali

Risulta naturale scegliere come identificatori principali quelli che già erano stati scelti in fase di progettazione concettuale e successivamente aggiornati nelle precedenti fasi di progettazione logica, dato che si è verificato che non presentano criticità e, come da specifica, identificano univocamente le entità a cui sono associati.

### Risultato finale

Si riporta di seguito lo schema E-R ristrutturato per intero:



## Traduzione verso il modello relazionale

Il risultato finale è il seguente, ottenuto tramite l'applicazione di regole banali di traduzione:

<p><b>RisorsaAstratta</b>(<u>ID</u>, Titolo, Data di pubblicazione, Genere, Tipologia) <b>Pubblicazione</b>(<u>Risorsa</u>, <u>Autore</u>) Pubblicazione.Risorsa → RisorsaAstratta.ID <b>Autore</b>(<u>ID</u>, Nome, Cognome) Pubblicazione.Autore → Autore.ID <b>RisorsaFisica</b>(<u>RisorsaAstratta</u>, <u>Barcode</u>, Disponibilità, Biblioteca, Sala) RisorsaFisica.RisorsaAstratta → RisorsaAstratta.ID <b>Biblioteca</b>(<u>Nome</u>, Dipartimento, Responsabile) RisorsaFisica.Biblioteca → Biblioteca.Nome <b>Dipendente</b>(<u>ID</u>, Nome, Cognome, CF, Impiego) Dipendente.Impiego → Biblioteca.Nome Biblioteca.Responsabile → Dipendente.ID <b>Recapito</b>(<u>Dipendente</u>, <u>Contatto</u>) Recapito.Dipendente → Dipendente.ID <b>Attività</b>(<u>Nome</u>, <u>Data_inizio</u>, Scadenza prevista, Data di fine*) <b>Partecipazione</b>(<u>Dipendente</u>, <u>Attività</u>, <u>Data_inizio</u>) Partecipazione.Dipendente → Dipendente.ID Partecipazione.Attività → Attività.Nome Partecipazione.Data_inizio → Attività.Data_inizio <b>Contributo</b>(<u>ID_Risorsa</u>, <u>Barcode_Risorsa</u>, <u>Attività</u>, <u>Data_inizio</u>) Contributo.ID_Risorsa → RisorsaFisica.RisorsaAstratta Contributo.Barcode_Risorsa → RisorsaFisica.Barcode Contributo.Attività → Attività.Nome</p>
---

Le informazioni dell'associazione Collocazione, uno-a-molti con partecipazione opzionale di entrambe le entità, sono state inserite nella relazione RisorsaFisica, in quanto si ritiene che questo sia più utile ai fini di questa base di dati.

# Implementazione in MySQL

## Creazione e popolamento tabelle

Per quanto riguarda la creazione delle tabelle, l'unica criticità riscontrata è stata quella di inserire i vincoli di integrità individuati nella fase di traduzione dello schema, in particolare quello tra Biblioteca.ID\_Responsabile e Dipendente.ID, dato sono presenti vincoli in entrambe le direzioni (risultanti dalla traduzione delle due associazioni Responsabile e Impiego tra Biblioteca e Dipendente). Si sono perciò resi necessari l'utilizzo di un comando ALTER TABLE per aggiungere tale vincolo nella fase di creazione e di un comando UPDATE TABLE per inserire gli ID dei responsabili nella fase di popolamento.

Il popolamento è stato effettuato seguendo varie modalità: inserimento diretto di record nel codice o popolamento da file (di tipo txt e csv).

## Interrogazioni

Sono state realizzate 6 interrogazioni utilizzando diversi costrutti del linguaggio, commentate all'interno dello script. In particolare:

- trovare nome e cognome dei dipendenti con un particolare codice fiscale (in questo caso, che inizia per M e termina per R);
- trovare gli autori le cui risorse fisiche pubblicate sono presenti in una data biblioteca (Architettura in questo caso);
- trovare, per ogni attività, i dipendenti che vi hanno partecipato;
- trovare i dipendenti che non hanno mai partecipato a un'attività. Si è deciso, per semplificare la query, di creare una vista DipendentiAttività, che sarà anche riutilizzata in seguito;
- trovare gli autori che hanno pubblicato più di una risorsa
- trovare il dipendente che ha partecipato al maggior numero di attività. Per semplificare questa query si è utilizzata la stessa vista DipendentiAttività di sopra.

Si noti che durante la fase di test su MacOS si è reso necessario l'utilizzo del seguente comando, inserito commentato nello script, per far funzionare la terza query:

```
SET sql_mode=(SELECT REPLACE (@@sql_mode, 'ONLY_FULL_GROUP_BY', ''));
```

## Procedure e funzioni

Sono state implementate due procedure e una funzione, in particolare:

- una procedura che realizza l'Operazione 2 descritta precedentemente, ossia la ricerca di una risorsa fisica e della sua collocazione per titolo;
- una procedura che, dato un dipendente, restituisce il nome del responsabile della biblioteca in cui lavora;
- una funzione che restituisce il numero di volte in cui una data risorsa astratta è stata utilizzata in un'attività iniziata in un dato anno, e che restituisce NULL se si inserisce un anno successivo al corrente.

Si noti che per permettere la creazione della funzione si è reso necessario su MacOS l'utilizzo del seguente comando, inserito commentato nello script:

```
SET GLOBAL log_bin_trust_function_creators=TRUE;
```

Su Windows si è reso invece necessario esplicitare la keyword DETERMINISTIC.

## Viste

Sono state create due viste correlate:

- DipendentiMatematica contiene le informazioni relative ai dipendenti che lavorano nella Biblioteca di Matematica "DIMAI-Ulisse DINI". È stato inoltre aggiunto un vincolo di tipo WITH LOCAL CHECK OPTION per controllare che i dipendenti inseriti lavorino effettivamente in questa biblioteca. Sono presenti degli esempi nello script;
- AttivitaDipendentiMatematica è definita anche in termini di DipendentiMatematica e contiene le informazioni relative alle attività a cui questi hanno partecipato (anche con dipendenti di altre biblioteche)

## Trigger

Si è deciso di implementare un trigger sulla tabella Risorsa Fisica: quando si prova ad inserire una Risorsa Fisica la cui Risorsa Astratta correlata non esiste, viene eseguito anche l'inserimento di una nuova tupla di Risorsa\_Astratta contenente solamente l'ID di tale risorsa. L'inserimento in Risorsa\_Fisica viene eseguito in ogni caso alla fine.