

PROGETTO BASI DI DATI

Corso di Laurea di Informatica

Anno 2025/2026



Ardi Ndreu - 7161575

Rayan Hakam Taleb Moh'd - 7157792

Swaran Singh - 7159864

Indice

PROGETTO BASI DI DATI	1
Indice	2
1. Progettazione concettuale.....	3
1.1 Il dominio dell'applicazione.....	3
1.2 La richiesta	3
1.3 Ristrutturazione della richiesta	4
1.4 Schema ER.....	5
1.4.1 Gestione dei Fascicoli e Classificazione dei Reati	5
1.4.2 Gerarchia delle Persone e Ruoli Investigativi	6
1.4.3 Reperti, Analisi e Catena di Custodia	7
2. Progettazione Logica.....	8
2.1 Analisi dei Volumi e delle Operazioni.....	8
2.1.1 Tabella dei Volumi	8
2.1.2 Tabelle delle Operazioni	8
2.1.3 Ridondanze	8
2.2 Diagramma logico.....	9
3. Implementazione in SQL	10
3.1 Ambiente di Sviluppo	10
3.2 Creazione Tabelle	10
3.3 Procedure	10
3.3.1 Navigazione nella struttura ad albero	10
3.3.2 Gestione Operativa	11
3.4 Trigger.....	11
3.4.1 Gestione delle automazioni e coerenza dei processi	12
3.4.2 Gestione delle modifiche dei dati	12
3.5 Viste	13

1. Progettazione concettuale

1.1 Il dominio dell'applicazione

Il Dipartimento di Scienze Forensi necessita di una base di dati per la gestione digitalizzata dei fascicoli investigativi con l'obiettivo primario di garantire la tracciabilità dei reperti attraverso la catena di custodia e la gestione delle analisi di laboratorio. Il sistema deve assicurare l'integrità e la sicurezza delle prove, gestendo rigorosamente i ruoli del personale e le incompatibilità con i soggetti coinvolti nei crimini.

1.2 La richiesta

Il sistema deve permettere la memorizzazione dei **luoghi** di interesse (Magazzini, Laboratori, Scene del Crimine, Archivi o Uffici), registrandone nome, indirizzo, tipologia e una descrizione.

Le **persone** sono descritte tramite codice fiscale, nome, cognome e data di nascita. E per i soggetti coinvolti in un caso è necessaria una descrizione.

Agenti sono delle persone, per le quali è necessario registrare anche matricola, grado (Agente, Ispettore, Commissario, Questore, Sovrintendente) e dipartimento di appartenenza (Polizia Postale, Carabinieri, Guardia di Finanza, Polizia Scientifica, Servizi Antidroga).

Le attività investigative sono organizzate in **fascicoli**, identificati univocamente da un codice. Ogni fascicolo possiede uno stato (Aperto, Chiuso, Archiviato), una priorità, una data di prima apertura ed è affidato ad un unico **agente responsabile**. Per fascicolo aperto si intende che i processi sono in corso, per fascicolo archiviato si intende che i processi sono stati interrotti, invece per fascicolo chiuso si intende che i processi sono conclusi. Un fascicolo può riguardare diverse categorie di reato. Vigono regole di sicurezza stringenti:

- Non è possibile modificare o inserire nuovi dati (reperti, analisi o soggetti) se il fascicolo è in stato "Chiuso" o "Archiviato".
- Per evitare conflitti di interesse, un agente o un analista non può operare su un caso in cui figura personalmente come soggetto coinvolto (Sospettato, Vittima o Testimone).

Le prove materiali sono denominate **reperti**. Di ogni reperto si registrano codice univoco, descrizione, categoria (Biologico, Chimico, Traccia, Balistico, Informatico,

Documentale, Generico). Il sistema deve gestire la gerarchia delle prove, permettendo a un reperto di essere generato da un altro (es. un campione di DNA estratto da un indumento). **Catena di custodia** è fondamentale: ogni spostamento del reperto deve essere tracciato registrando data, nuovo luogo, agente responsabile del trasferimento e descrizione del motivo.

Infine, sui reperti possono essere effettuate **analisi di laboratorio**. Per ogni analisi si devono memorizzare: descrizione, data di inizio, data ed esito finale, reperto analizzato e la persona (analista) che ha eseguito il test.

1.3 Ristrutturazione della richiesta

Fascicolo. Per ogni fascicolo, identificato da un codice univoco, rappresentiamo lo stato corrente, la priorità, la data di apertura e l'agente responsabile delle indagini. Un fascicolo può essere associato a più categorie di reato. Non è possibile modificare o aggiungere elementi a un fascicolo se il suo stato è "Chiuso" o "Archiviato".

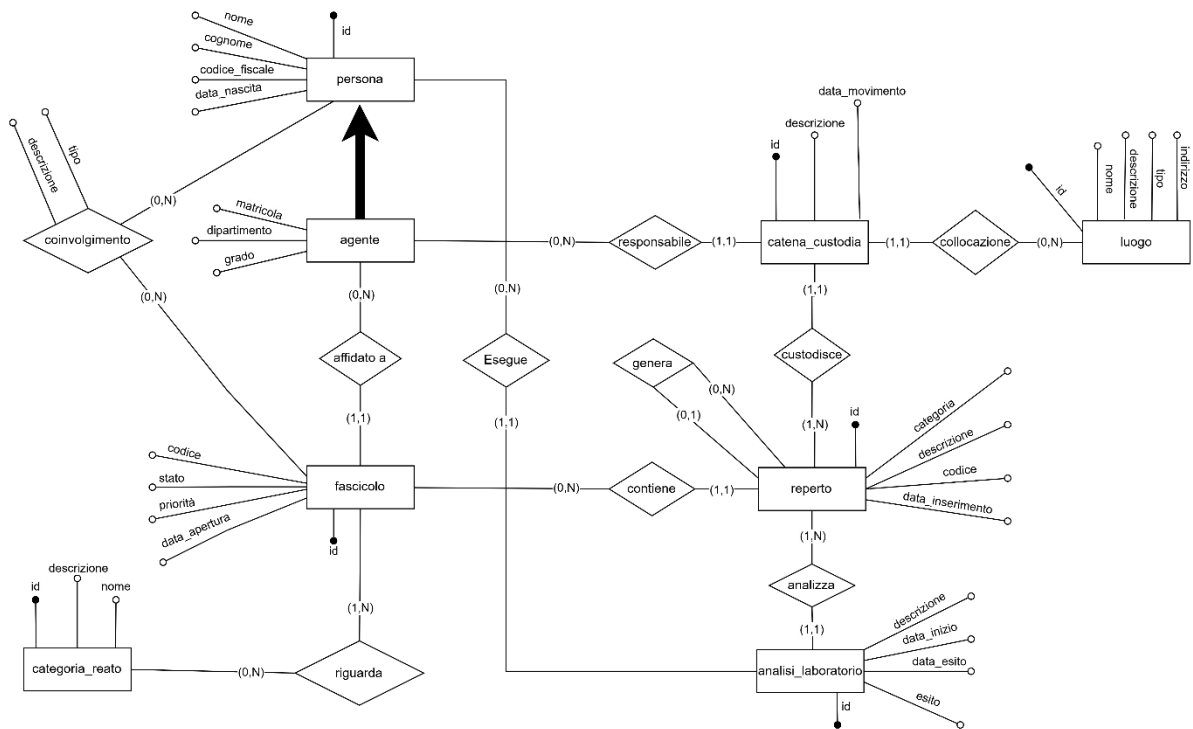
Persone e Agenti. Per ogni persona rappresentiamo codice fiscale, nome, cognome e data di nascita, e se è coinvolta in un caso, una descrizione. Un sottoinsieme delle persone è costituito dagli agenti, per i quali memorizziamo anche matricola, grado e dipartimento. Una persona può essere coinvolta in un fascicolo con ruoli diversi.

Reperti. Per ogni reperto, identificato da un codice, rappresentiamo la descrizione, la categoria e il fascicolo di riferimento. Un reperto può derivare da un altro reperto "padre".

Catena di custodia. Per garantire la tracciabilità, ogni movimento di un reperto viene registrato nella catena di custodia. Per ogni movimento rappresentiamo l'agente responsabile, il luogo di destinazione, la descrizione dello spostamento (motivo, mezzi, conservazione, etc.) e la data dell'operazione.

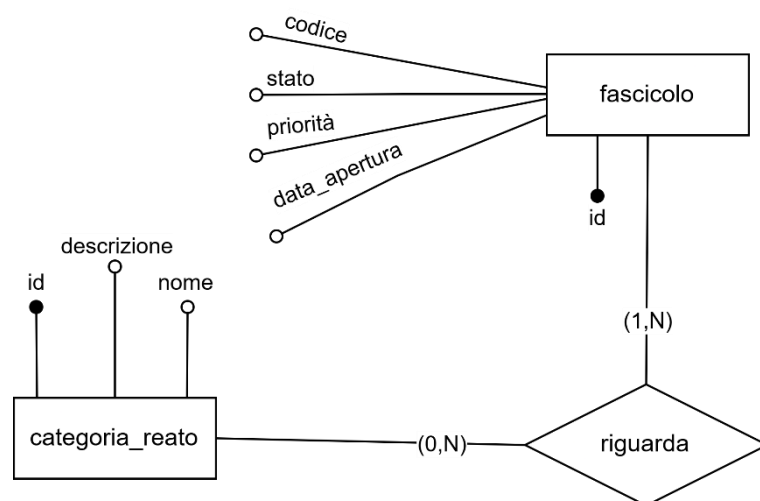
Analisi. I reperti possono essere sottoposti ad un'analisi di laboratorio. Per ogni analisi rappresentiamo la descrizione, l'esito, le date di inizio e fine, e l'analista (**persona**) che l'ha eseguita. Un analista non può lavorare su reperti di un caso in cui è coinvolto personalmente.

1.4 Schema ER



L'immagine presente qui sopra descrive il diagramma Entity-Relationship. Per una lettura facile, analizziamo il diagramma scomponendolo in tre macroaree tematiche

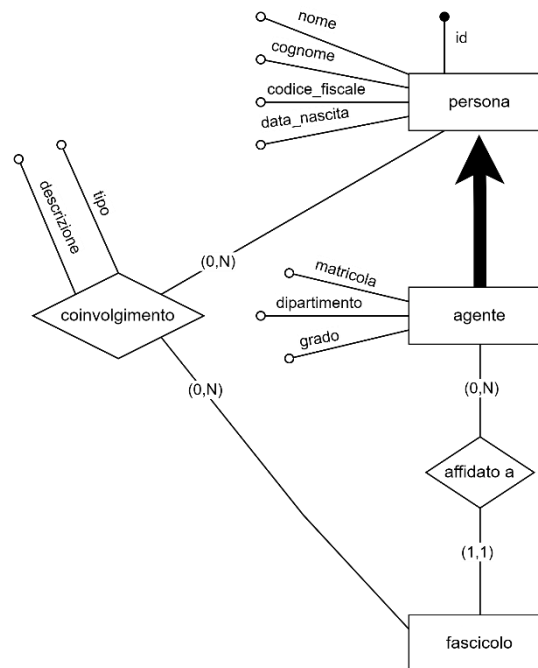
1.4.1 Gestione dei Fascicoli e Classificazione dei Reati



Il nucleo centrale del sistema è rappresentato dall'entità **fascicolo**, che aggrega tutte le informazioni investigative. Tra l'entità **fascicolo** e l'entità **categoria_reato** sussiste

una relazione molti-a-molti: un singolo fascicolo può riguardare una o più tipologie di reato, viceversa, una categoria di reato può essere associata a 0 o numerosi fascicoli distinti.

1.4.2 Gerarchia delle Persone e Ruoli Investigativi

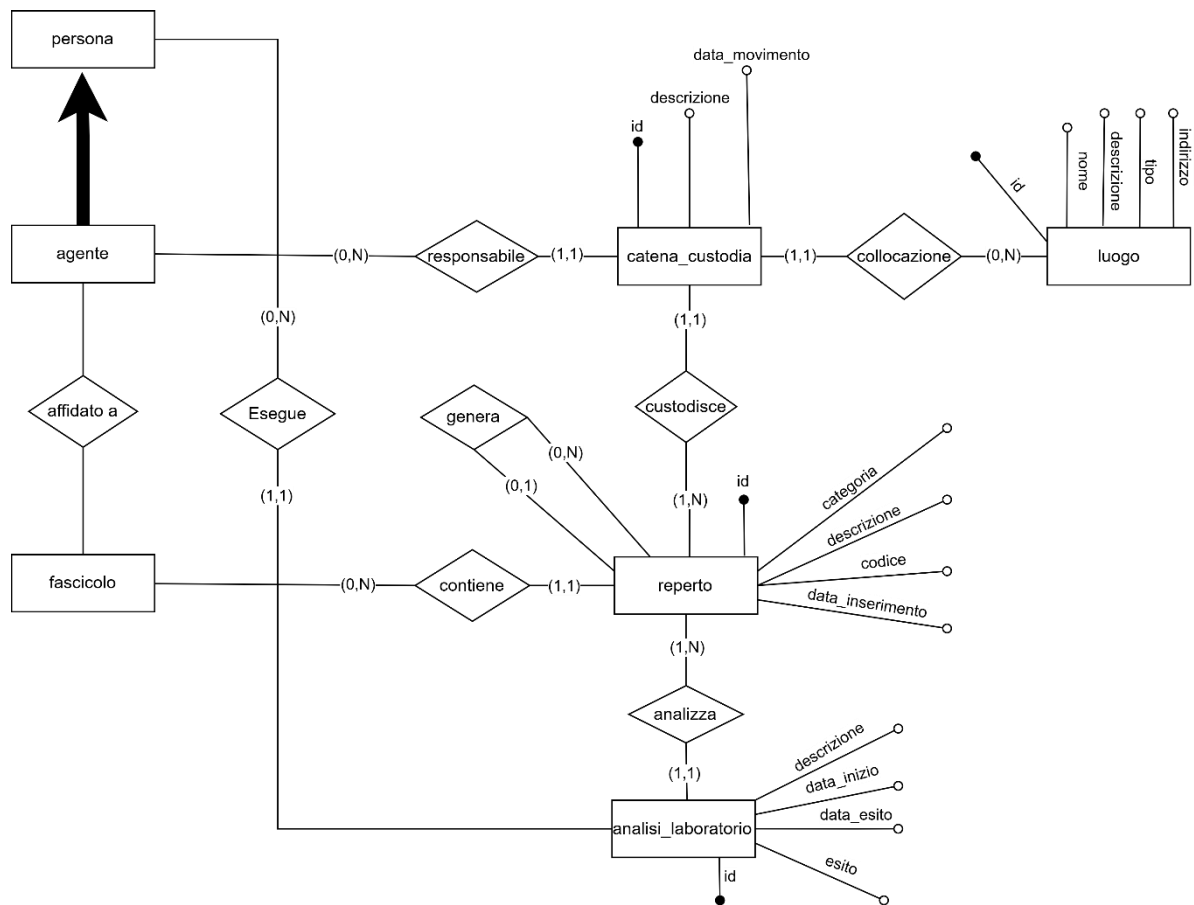


L'entità *persona* rappresenta genericamente ogni individuo con i propri dati anagrafici. Da questa deriva l'entità figlia **agente**, che eredita tutti gli attributi della *persona* aggiungendo caratteristiche specifiche del ruolo professionale (matricola, grado, dipartimento).

Le relazioni tra i soggetti e i fascicoli sono di due tipi distinti:

- **Affidato a:** È una relazione tra *agente* e *fascicolo*. Un fascicolo deve essere affidato alla responsabilità di un solo agente, mentre un agente può essere responsabile di zero o più casi (cardinalità 0:N).
- **Coinvolgimento:** È una relazione tra la generica *persona* e il *fascicolo*. Poiché una persona può essere implicata in più casi e un caso coinvolge più persone, si tratta di una relazione molti-a-molti. Questa relazione possiede attributi propri (*tipo* e *descrizione*) per specificare il ruolo che il soggetto ricopre in quel preciso contesto (Sospettato, Vittima o Testimone).

1.4.3 Reperti, Analisi e Catena di Custodia



Questa macroarea modella la gestione delle prove materiali. L'entità **reperto** è associata all'entità fascicolo (un reperto appartiene a un solo fascicolo, mentre un fascicolo può contenere zero o più reperti). Un aspetto cruciale è la relazione ricorsiva sull'entità reperto: un reperto (figlio) può essere generato o estratto da un altro reperto (padre), un reperto padre può generare n sotto reperti, come nel caso di un campione biologico prelevato da un oggetto sequestrato.

La gestione dei reperti si articola in due ulteriori rami:

- **Analisi in Laboratorio:** L'entità **analisi_laboratorio** è legata ad un reperto ed è associata ad una Persona (specificatamente un analista) che esegue il test.
- **Catena di Custodia:** Per garantire la tracciabilità legale, è stata modellata un'entità complessa che lega tre altre entità: reperto, agente e luogo. Questa struttura registra lo storico dei movimenti, specificando per ogni spostamento chi (agente) ha spostato cosa (reperto) e verso dove (luogo), creando così una catena di custodia affidabile.

2. Progettazione Logica

2.1 Analisi dei Volumi e delle Operazioni

Per procedere alla progettazione logica prevediamo e analizziamo le seguenti tabelle dei volumi e delle operazioni.

2.1.1 Tabella dei Volumi

Concetto	Tipo	Volume
persona	E	5.000
agente	E	100
categoria_reato	E	50
fascicolo	E	500
reperto	E	2.500
luogo	E	3.500
analisi_laboratorio	E	2.000
catena_custodia	E	7.500
coinvolgimento	R	1.500
categoria_fascicolo	R	600

2.1.2 Tabelle delle Operazioni

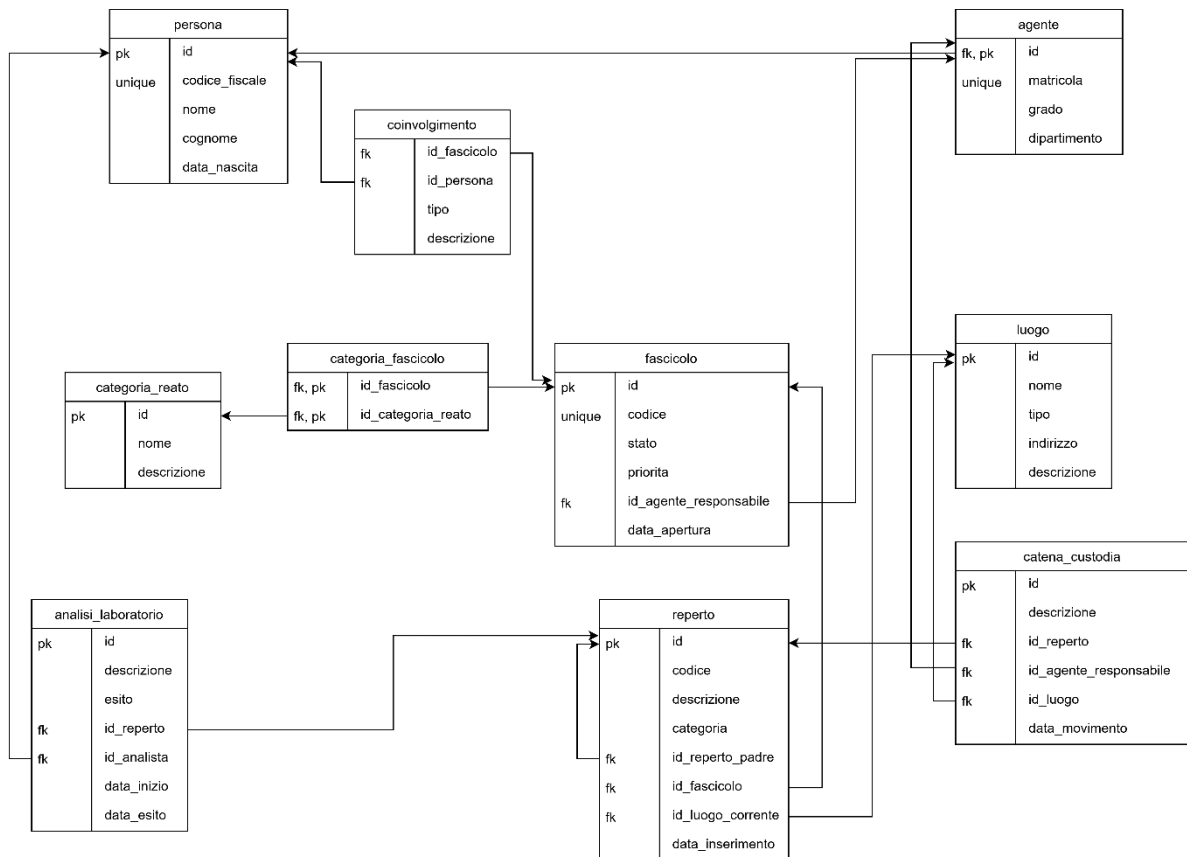
Operazione	Descrizione	Tipo	Frequenza
Op. 1	Inserimento di un reperto in un fascicolo	Scrittura	7 al giorno
Op. 2	Aggiornamento della catena di custodia	Scrittura	30 al giorno
Op. 3	Ricerca di tutti i reperti di un fascicolo	Lettura	60 al giorno

2.1.3 Ridondanze

Dalla tabella dei volumi e delle operazioni, senza neanche ricorrere a formule matematiche, emerge una chiara necessità di ottimizzare l'operazione di lettura della posizione dei reperti (Op. 3), che avviene con alta frequenza. Teoricamente, la posizione attuale di un reperto è derivabile analizzando lo storico dei movimenti nell'entità *catena_custodia* e selezionando l'ultimo inserimento in ordine temporale. Tuttavia, considerando che la tabella *catena_custodia* tende a crescere molto rapidamente, calcolare la posizione attuale tramite query complesse su tale tabella

ad ogni accesso risulterebbe oneroso. A tale scopo si è deciso di introdurre l'attributo **id_luogo_corrente** nell'entità reperto.

2.2 Diagramma logico



La conversione dello schema E-R in schema logico relazionale ha seguito le regole di derivazione standard. Si evidenziano le seguenti scelte implementative specifiche.

- **persona-agente:** l'entità **agente**, ha come padre l'entità **persona**. Quindi id nella tabella agente è sia da chiave primaria, sia chiave esterna verso la tabella padre persona.
- **Relazione ricorsiva:** l'entità **reperto** presenta un'associazione riflessiva tradotta con id_reperto_padre. Trattandosi di un legame gerarchico, è stato necessario prevedere vincoli di integrità (implementati successivamente tramite trigger) per evitare la creazione di cicli infiniti (loop) che comprometterebbero la struttura dell'albero.

3.Implementazione in SQL

3.1 Ambiente di Sviluppo

Il progetto è implementato e testato nel seguente ambiente di sviluppo:

- Sistema operativo: **Windows 11**
- DBMS: **MySQL Server 8.0**
- IDE: **MySQL Workbench 8.0.43**

3.2 Creazione Tabelle

L'implementazione in SQL dello schema logico non richiede particolari attenzioni. Tuttavia, bisogna attenersi a un certo ordine di creazione in quanto alcune tabelle hanno vincoli referenziali. Poi da tenere in mente che per il tipo di un luogo, grado di un agente, dipartimento di un agente, stato di un fascicolo, tipo di un coinvolgimento, categoria di un reperto, si è deciso di adottare un enumeratore. La creazione delle tabelle è presente nel file setup.sql

3.3 Procedure

Per incapsulare la logica di alcune operazioni complesse, sono state implementate quattro Stored Procedures e sono presenti nel file setup.sql. Di cui se ne descrive il funzionamento e l'implementazione logica.

3.3.1 Navigazione nella struttura ad albero

L'investigazione richiede spesso di tracciare l'origine di una prova (es. un campione di DNA estratto da un indumento) o, viceversa, di individuare tutti i reperti derivati da una prova padre. Per risolvere questo problema, che implica l'attraversamento di un albero, si è deciso di implementare le seguenti procedure.

- **risali_albero_reperti (in id_reperto int):** questa procedura, implementata con Common Table Expression, permette di ricostruire la "storia". la query parte dal reperto in input (figlio) e “risale ricorsivamente” con id_reperto_padre fino a raggiungere il primo reperto (quello che non ha genitori). Il parametro livello viene incrementato ad ogni passo ricorsivo, permettendo di quantificare anche la distanza fra reperti.
- **trova_tutti_discendenti (in id_reperto int):** esegue l'operazione inversa alla precedente. Partendo da un reperto "padre", si scende la gerarchia selezionando tutti i nodi figli, nipoti e successivi.

3.3.2 Gestione Operativa

- **registra_movimento_reperto (...):** questa procedura astrae l'operazione di inserimento nella tabella catena_custodia. Invece di eseguire inserimenti diretti, si usa la procedura che richiede in input i soli dati essenziali (reperto, agente, luogo, motivo).
- **crea_reperto_figlio (...):** questa procedura automatizza il processo eseguendo tre operazioni sequenziali:
 1. Recupero Dati: interroga il database per ottenere l'ID del fascicolo e l'ID del luogo corrente del reperto padre. Questo garantisce che il figlio appartenga allo stesso caso e si trovi inizialmente nello stesso luogo fisico del genitore.
 2. Creazione Reperto: inserisce il nuovo record nella tabella reperto utilizzando i dati ereditati e quelli passati in input.
 3. Tracciamento Immediato: utilizzando la funzione last_insert_id(), recupera l'id appena generato, assicurandosi che il nuovo reperto nasca già con una storia di custodia tracciata.

3.4 Trigger

L'implementazione dei trigger, che è presente nella terza parte del file setup.sql, risponde alla duplice esigenza di automatizzare i flussi operativi e di imporre vincoli che garantiscano la coerenza logica dei dati.

3.4.1 Gestione delle automazioni e coerenza dei processi

- **update_posizione_reperto:** all'inserimento di un nuovo movimento nella tabella *catena_custodia*, il sistema aggiorna istantaneamente l'attributo *id_luogo_corrente* del reperto associato. Ciò elimina la necessità di doppi aggiornamenti manuali.
- **no_insert_fascicolo_chiuso_archiviato:** ogni nuovo fascicolo deve essere creato necessariamente nello stato "Aperto", e può essere chiuso o archiviato successivamente.
- **no_insert_reperto_figlio_fascicolo_diverso:** garantisce l'integrità della gerarchia delle prove. Il trigger verifica che, al momento della creazione di un reperto figlio, esso venga associato al medesimo fascicolo del reperto padre, prevenendo inconsistenze logiche nell'albero dei reperti.
- **Gestione del Conflitto di Interessi** è gestito dai trigger della **sezione 4**. Questo insieme di trigger implementa i vincoli richiesti. Il sistema non permette l'assegnazione di ruoli operativi (Responsabile del caso, Analista o Agente responsabile della custodia di un reperto) a persone che figurano già nel fascicolo come parti coinvolte (sospettati, testimoni, vittime). Per aggiungere un agente responsabile come coinvolto nel dato caso, bisogna prima sollevarlo dall'incarico, così da non avere incoerenze nei dati.
- **Congelamento dei Fascicoli Chiusi o Archiviati** è gestito dai trigger della **sezione 5**. Per garantire l'inalterabilità delle indagini concluse o Archivate, questi trigger bloccano qualsiasi operazione di scrittura (inserimento o modifica) su reperti, analisi e soggetti coinvolti se il fascicolo di riferimento si trova in stato "Chiuso" o "Archiviato". Qualsiasi integrazione probatoria richiede la formale riapertura del fascicolo (cambio di stato in "Aperto"), e non è consentita la riapertura dei fascicoli chiusi.

3.4.2 Gestione delle modifiche dei dati

Per garantire la sicurezza tipica dell'ambito forense, si è scelto di limitare drasticamente le operazioni di modifica. La filosofia adottata prevede che i dati sensibili, una volta scritti, siano inalterabili: eventuali errori devono essere corretti tramite la cancellazione e il reinserimento del record.

Di seguito si dettagliano le uniche operazioni di modifica consentite, mentre ogni altro tentativo di alterazione dei dati verrà respinto dal DBMS.

- **Tabella agente:** è consentito l'aggiornamento esclusivamente degli attributi grado e dipartimento per un eventuale promozione al grado oppure un trasferimento a un dipartimento diverso.
- **Tabella fascicolo:** le modifiche sono permesse solo per gestire il ciclo di vita dell'indagine (cambio dello stato) e relative ai cambiamenti di operatività (priorità e agente responsabile).
- **Tabella reperto:** al fine di garantire la coerenza con la catena di custodia, è consentita la modifica automatica dell'attributo ridondante id_luogo_corrente; ed è permessa anche la modifica del campo descrizione. La scelta di vietare modifiche al campo id_reperto_padre è di fondamentale importanza, in quanto previene la creazione di falle logiche di tipo loop.

3.5 Viste

Per facilitare l'accesso ai dati, sono state definite delle Viste MySQL (nell'ultima parte del file setup.sql) che aggregano le informazioni necessarie per le interrogazioni quotidiane.

- **dettagli_reperti:** fornisce un elenco completo dei reperti arricchito dai dati relativi. Particolarmente utile per rispondere alla richiesta: "Mostra tutti i reperti associati al fascicolo XXX".
- **storico_spostamenti:** questa vista ricostruisce in modo leggibile la catena di custodia. Risponde direttamente alla necessità operativa di tracciare cronologicamente tutti gli spostamenti di uno specifico reperto.
- **registro_coinvolgimenti:** è una vista che unisce i dati anagrafici delle persone con i dati dei fascicoli. Questa singola struttura risponde a due esigenze diverse, a seconda di come viene filtrata:
 - **Analisi del caso:** filtrando per fascicolo, restituisce l'elenco di tutti i soggetti coinvolti (sospettati, testimoni, vittime).
 - **Profilazione soggetto:** filtrando per persona, restituisce lo storico di tutti i fascicoli in cui un individuo è stato coinvolto.