



**UNIVERSITÀ  
DEGLI STUDI  
DI UDINE**

**hic sunt futura**

---

**BASI DI DATI – PROGETTO DI  
LABORATORIO  
A.A. 2024/25**

---

**Componenti del gruppo:**

- Lovepreet Singh 153012 – [singh.lovepreet@spes.uniud.it](mailto:singh.lovepreet@spes.uniud.it)
- Willy Kelvin Junior Djangang 165844 – [djangang.willykelvinjunior@spes.uniud.it](mailto:djangang.willykelvinjunior@spes.uniud.it)
- Alessandro Bolteri 164884 – [bolteri.alessandro@spes.uniud.it](mailto:bolteri.alessandro@spes.uniud.it)

REPOSITORY CONTENENTE TUTTO SULLA CATENA DI OFFICINE [GITHUB](#)

# Sommario

1.Introduzione .....	4
FASE 1: ANALISI DEI REQUISITI.....	5
1. Analisi dei Requisiti .....	5
1.1 Sintesi del testo progetto .....	5
1.2 Strutturazione dei requisiti in gruppi di frasi omogenee.....	6
▀ 1.3 Ambiguità e Assunzioni.....	7
📌 1.3.1 Assunzioni Preliminari .....	7
🔍 1.3.2 Ambiguità e Chiarimenti Risolti.....	8
✳ 1.3.3 Considerazioni sulla Richiesta_Fornitura .....	9
📘 1.4 Glossario – Entità Principali del Sistema Officina .....	9
Glossario dei termini .....	9
✓ Note Aggiuntive: .....	11
FASE 2 : Progettazione concettuale .....	11
2.1Diagramma E-R.....	11
🔧 2.2) Regole di Derivazione.....	12
2.3) Vincoli di integrità .....	13
FASE 3.Progettazione logica.....	14
✓ 3.1 – Tabella dei Volumi dei Dati.....	14
✓ 3.2 – Tabella delle Operazioni (frequenza).....	15
🔍 3.3 – Casi di Navigazione (Accesso ai Dati) .....	15
▶ Operazione 1: Creazione Intervento .....	15

	Operazione 2: Generazione Fattura.....	16
	Operazione 3: Contare gli interventi per officina .....	17
	3.3.1. Senza attributo derivato (calcolo on-demand).....	17
	3.3.2 Con attributo derivato (Numero_Interventi in Officina).....	18
	Confronto prestazionale.....	18
	Formula per valutare l'opportunità.....	19
	Soluzione ibrida (consigliata) .....	19
	3.4 Diagramma E-R ristrutturato.....	20
	3.5 Schema logico.....	22
4	Progettazione fisica E IMPLEMENTAZIONE .....	26
4.1	PROGETTAZIONE FISICA.....	26
	Creazione del database e delle tabelle .....	26
	Definizione di domini e tipi personalizzati .....	26
	Automobile .....	28
	Gestione delle Trigger( Vincoli ) .....	29
	Aggiornamenti e Cancellazioni.....	32
	Aggiornamento del chilometraggio di un'automobile .....	32
	Aggiornamento dello stato di un intervento .....	32
	Cancellazione di un cliente.....	33
	Cancellazione di un intervento.....	33
	6. Analisi dei Dati in R .....	33
	Analisi Statistica – Catena di Officine .....	34
	Connessione al database e caricamento librerie per far l analisi e grafici .....	34
	1. Distribuzione Clienti per Continente (con Nazione dominante) .....	34
	Analisi 2 – Top Officine per Fatturato Totale .....	36
	2. Top 5 Marche Auto Più Presenti .....	37
	4. Top 10 Officine per  Fatturato Totale .....	38

## **1.Introduzione**

Scopo del Progetto 29

Il progetto consiste nel progettare e implementare una base di dati seguendo la seguente consegna:

*“Si progetti uno schema entità/relazioni (ER) che descriva i dati di una applicazione relativa ad una catena di officine, motivando adeguatamente ogni scelta fatta. Lo schema dovrà memorizzare le seguenti informazioni.*

- *Le officine, con nome (identificante), indirizzo e telefono*
- *Le automobili con targa (identificante), modello (una stringa di caratteri senza ulteriore struttura) e proprietario.*
- *I clienti (proprietari di automobili), con codice fiscale, cognome, nome e telefono. Ogni cliente può essere proprietario di più automobili.*
- *Gli interventi di manutenzione, ognuno effettuato presso un'officina e con un numero progressivo (unico nell'ambito della singola officina), date di inizio e di fine, pezzi di ricambio utilizzati (con le rispettive quantità) e numero di ore di manodopera.*
- *I pezzi di ricambio, con codice, nome e costo unitario.*

*Successivamente, si trasformi lo schema ER ottenuto al passo precedente nel corrispondente schema relazionale, indicando le eventuali chiavi esterne degli schemi di relazione. Nel vostro caso, i requisiti saranno il testo del progetto • risolve eventuali ambiguità nei requisiti • produce un glossario*

*In questa relazione, documentiamo il processo di progettazione di una base di dati per la gestione di una catena di officine, analizzando l'approccio seguito dall'analisi dei requisiti e la risoluzione delle ambiguità passando per la progettazione concettuale con la definizione di entità, relazioni, attributi "ragionevoli" e chiavi, fino alla trasformazione logica nello schema relazionale con l'indicazione delle chiavi esterne.*

*L'obiettivo è presentare i passi compiuti e i risultati ottenuti durante la produzione di questa specifica base di dati*

## FASE 1: ANALISI DEI REQUISITI

### 1. Analisi dei Requisiti

#### 1.1 Sintesi del testo progetto

*Il compito proposto consiste nello sviluppo di un modello Entità-Relazione (E-R) volto a rappresentare un sistema informativo per la gestione di una catena di centri autorizzati alla riparazione di autoveicoli. Le entità rilevanti da trattare sono le seguenti:*

- **Officine:**
  - Identificate da un nome univoco.
  - Hanno un indirizzo e un telefono.
- **Automobili:**
  - Identificate da una targa univoca.
  - Hanno un modello (stringa di caratteri senza struttura) e un proprietario.
- **Clienti (proprietari delle automobili):**
  - Identificati da un codice fiscale.
  - Hanno cognome, nome e telefono.
  - Ogni cliente può essere proprietario di più automobili.
- **Interventi :**
  - Eseguiti presso un'officina.
  - Identificati da un numero progressivo, univoco nell'ambito della singola officina.
  - Hanno una data di inizio, una data di fine, i pezzi di ricambio utilizzati (con quantità) e il numero di ore di manodopera.
- **Pezzi di ricambio:**
  - Identificati da un codice univoco.
  - Hanno un nome e un costo unitario.

## 1.2 Strutturazione dei requisiti in gruppi di frasi omogenee

Organizziamo i requisiti in gruppi di frasi omogenee, ciascuno relativo a una specifica entità o relazione, includendo le nuove specifiche.

### Frasi di carattere generale

Si intende progettare una base di dati per gestire una catena di officine automobilistiche.

### Frasi relative alle officine

Ogni officina ha un nome (identificante)

Ogni officina ha un indirizzo

Ogni officina ha un numero di telefono

### Frasi relative alle automobili

- Ogni automobile ha una targa (identificante)
- Ogni automobile ha un modello (una stringa di caratteri senza ulteriore struttura)
- Ogni automobile ha un proprietario

### Frasi relative ai clienti

- Ogni cliente è un proprietario di almeno un'automobile
- Ogni cliente ha un codice fiscale (identificante)
- Ogni cliente ha un cognome
- Ogni cliente ha un nome
- Ogni cliente ha un numero di telefono
- Ogni cliente può essere proprietario di più automobili.

### Frasi relative agli interventi

- Ogni intervento di manutenzione è effettuato presso un'officina
- Ogni intervento di manutenzione ha un numero progressivo (identificante) unico nell'ambito della singola officina
- Ogni intervento di manutenzione ha una data di inizio
- Ogni intervento di manutenzione ha una data di fine
- Ogni intervento di manutenzione utilizza un numero di ricambi
- Ogni intervento di manutenzione ha un numero di ore di manodopera

### Frasi relative ai pezzi di ricambio

- Ogni pezzo di ricambio ha un codice (identificativo)
- Ogni pezzo di ricambio ha un nome
- Ogni pezzo di ricambio ha un costo unitario

## 1.3 Ambiguità e Assunzioni

Durante l'analisi dei requisiti del sistema informativo per la gestione delle officine, sono emerse alcune ambiguità concettuali che abbiamo chiarito attraverso assunzioni progettuali e scelte modellistiche coerenti. Questo paragrafo documenta in modo sistematico tali scelte, funzionali alla successiva **progettazione concettuale, logica e fisica** del database.

### 1.3.1 Assunzioni Preliminari

#### Tema

#### Assunzione

<b>Telefon</b>	
<b>o e indirizz</b>	Obbligatorio per <b>Clienti e Officine</b> ; l'indirizzo delle officine comprende <i>via, città, CAP</i> (CAP opzionale).
<b>o Magazzi</b>	Ogni Officina ha <b>un solo Magazzino</b> (relazione 1:1). I pezzi sono gestiti tramite la relazione Stoccato.
<b>Fornitor i</b>	Ogni Pezzo di Ricambio è fornito da <b>almeno un Fornitore</b> (relazione N:1 tra Pezzo_Ricambio e Fornitore).
<b>Fattura</b>	Ogni Intervento può generare al massimo <b>una Fattura</b> (relazione 1:1). Un Cliente può avere più Fatture (1:N).
<b>Costo intervento</b>	Calcolato come: Ore_Manodopera * Costo_Orario + Σ(Quantità * Costo_Unitario_Pezzo).

### ⌚ 1.3.2 Ambiguità e Chiarimenti Risolti

Dubbio	Domanda	Assunzione / Soluzione Progettuale
<b>Distinzion e Cliente</b>	“Come distinguere privati e aziende?”	Introduzione di <b>generalizzazione</b> : Cliente entità padre, con sottotipi Privato (Nome, Cognome, Telefono) e Azienda (Ragione_Sociale, PIVA, Telefono). Entrambi possono possedere auto.
<b>Gestione stato</b>	“Si tiene traccia dello storico degli stati?”	No storico. Stato è un campo ENUM con valori: 'Inizio', 'In Corso', 'Sospeso', 'Annullato', 'Concluso'.
<b>Intervent o</b>	“Serve entità Manodopera separata per manodopera?”	No. Intervento contiene direttamente Ore_Manodopera e Costo_Orario. Nessuna entità Manodopera.
<b>Data_Fine intervent o</b>	“È obbligatoria sempre?”	No. Può essere NULL. Se presente, Data_Fine >= Data_Inizio.
<b>Disponibilità pezzi</b>	“Come si gestisce l'assenza scorte?”	Se i pezzi non sono disponibili, viene generata una <b>Richiesta_Fornitura</b> e l'intervento è sospeso (trigger). I pezzi restano nel sistema anche se disponibili = FALSE.
<b>Intervent o annullato</b>	“Si cancella o si conserva?”	Conservato. Stato impostato a 'Annullato'. Serve per tracciabilità e audit.

<b>Vendita automobili</b>	“Il cliente resta nel sistema?” “Cancellare”	Sì. Il cliente rimane, ma marcato come <b>Inattivo</b> . L’auto viene eventualmente disassociata.
<b>Chiusura officina</b>	o storicizzare? ”	Nessuna cancellazione. L’officina viene <b>marcata come inattiva</b> . I dati restano accessibili.
<b>Obsolescenza pezzi</b>	“Vanno eliminati?”	No. I pezzi obsoleti rimangono con <b>disponibile = FALSE</b> . Utili per riferimento storico.
<b>Cambio tipologia cliente</b>	“Un cliente può cambiare da privato a azienda?”	Sì. Si aggiornano gli attributi del record, senza duplicazione.

### 1.3.3 Considerazioni sulla Richiesta\_Fornitura

- La Richiesta\_Fornitura viene **attivata da trigger** al momento dell’uso di pezzi (**Utilizza**) **quando le scorte risultano insufficienti**.
- La tabella Richiesta\_Fornitura include:
  - Codice\_Pezzo, Quantità\_Richiesta, Data\_Richiesta
  - ID\_Magazzino (per l’officina)
  - Numero\_Intervento (associato)
- L’intervento che attiva la richiesta passa automaticamente allo **stato "Sospeso"**.
- Le **scorte vengono aggiornate** solo al momento dell’effettiva consegna da parte del fornitore (**Fornisce**).

### 1.4 Glossario – Entità Principali del Sistema Officina

#### Glossario dei termini

<b>Terminologie</b>	<b>Descrizione</b>	<b>Sinonimi</b>	<b>Collegamenti</b>
<b>Cliente</b>	Soggetto che possiede una o più automobili e può richiedere interventi o ricevere fatture. Può essere un privato o un'azienda.	Utente, Proprietario	Automobile, Fattura
<b>Automobile</b>	Veicolo registrato nel sistema, identificato da una <b>Targa</b> . È di proprietà di un cliente.	Auto, Veicolo	Cliente, Intervento
<b>Officina</b>	Struttura fisica in cui si effettuano interventi su automobili e dove è presente un magazzino associato.	Sede, Centro Riparazione, Ricevutario	Intervento, Magazzino
<b>Fattura</b>	Documento che riassume i costi relativi ad un intervento eseguito su un'automobile.	Ricevuta, Documento	Intervento, Cliente
<b>Intervento</b>	Operazione tecnica eseguita su un'automobile. Include manodopera, uso pezzi e stato operativo.	Riparazione, Servizio	Officina, Automobile, Pezzi, Fattura
<b>Pezzo</b> <b>Ricambio</b>	Componente installabile o sostituibile su un'automobile durante un intervento.	Ricambi, Componente	Intervento, Magazzino, Fornitore
<b>Fornitore</b>	Azienda che fornisce pezzi di ricambio alle officine.	Distributore, Partner	Pezzo Ricambio, Magazzino
<b>Magazzini</b>	Spazio collegato ad un'officina in cui vengono stoccati i pezzi di ricambio e da cui si prelevano per gli interventi.	Deposit, Storage	Officina, Pezzo Ricambio

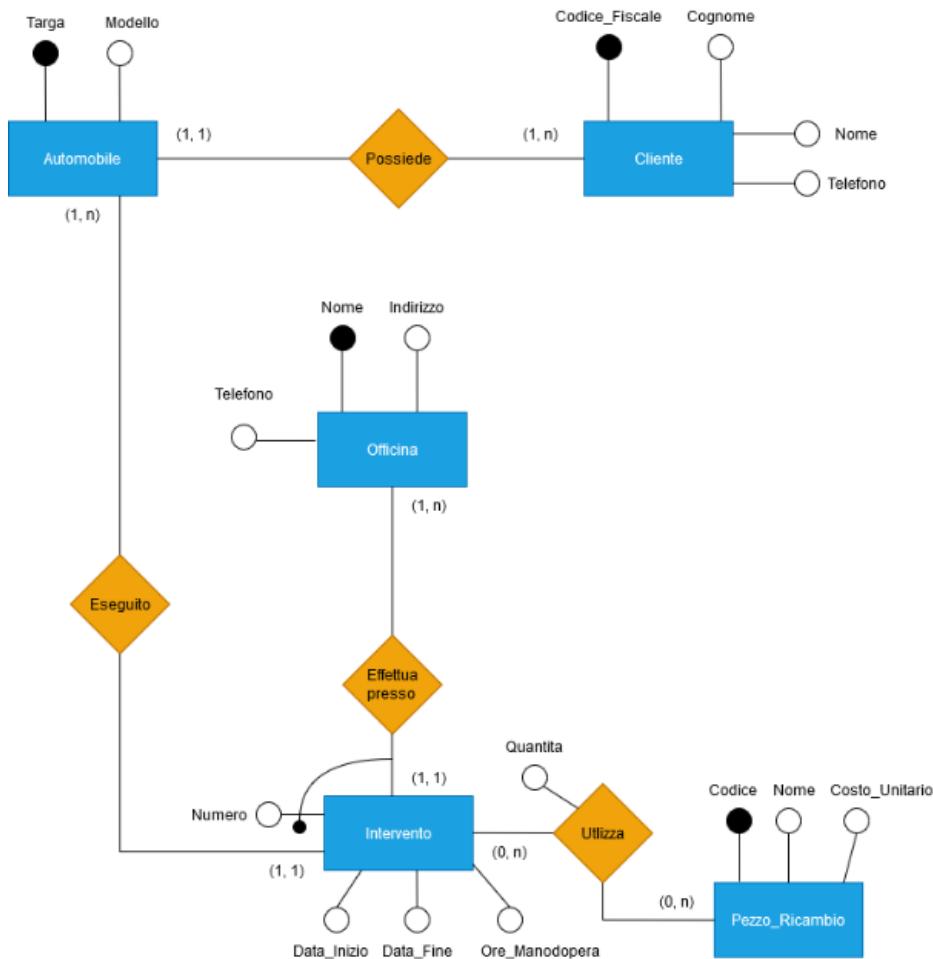
## Note Aggiuntive:

- Il **cliente** non può più essere specializzato in *Privato* e *Azienda* (come da modello ER esteso).
- L'**intervento** è l'elemento centrale che collega cliente, automobile, officina e costi.
- I **pezzi di ricambio** vengono tracciati nel **magazzino** e movimentati tramite interventi e forniture.

## FASE 2 : Progettazione concettuale

### 2.1 Diagramma E-R

Il diagramma E-R descritto include quattro entità principali (Officina, Intervento, Pezzo\_Ricambio, Automobile e Cliente) collegate da relazioni specifiche. Ogni entità ha attributi univoci e cardinalità definite, come la relazione 1-n tra Officina e Intervento o 1-1 tra Automobile e Cliente, con Intervento come entità debole dipendente dall'Officina.



## 2.2) Regole di Derivazione

#	Regola	Descrizione
1	<b>Durata Intervento</b>	<p>Se l'intervento è <i>Concluso</i>, la durata è la differenza tra <i>data_fine</i> e <i>data_inizio</i>. Se è ancora in corso o sospeso, si calcola la differenza tra <i>CURRENT_DATE</i> e <i>data_inizio</i>.</p> <p>Regola:</p> <p>Se <i>Data_Fine</i> è impostata → Stato = 'Concluso', altrimenti Stato IN ('Inizio', 'In Corso', 'Sospeso').</p>
2	<b>Disponibilità</b>	Un pezzo è considerato disponibile se nel magazzino

	<b>Pezzo Ricambio</b>	dell'officina esiste almeno una quantità positiva.
<b>3</b>	<b>Quantità Totale Stoccati per Pezzo</b>	Totale quantità di un pezzo specifico disponibile in tutti i magazzini.
<b>4</b>	<b>Costo Manodopera Intervento</b>	Derivato da Ore_Manodopera * Costo_Orario. Non memorizzato, ma calcolabile al volo.
<b>5</b>	<b>Aggiornamento Numero_Interventi Officina</b>	Numero di interventi INSERITI o ELIMINATI attivi per una determinata OFFICINA.

### 2.3) Vincoli di integrità

- Ogni Automobile può essere associata a uno o più Interventi ma abbiamo definito purché sia solo 1, ma ogni Intervento è associato a una sola Automobile e a una sola Officina. Un Intervento non può essere eseguito su un'automobile non registrata nel sistema.
- Un Cliente può possedere più Automobili, ma ogni Automobile può essere associata a un solo Cliente. Non esistono automobili senza proprietario.
- Ogni Officina può gestire più Interventi, ma ogni Intervento è eseguito da una sola Officina. Un Intervento non può essere associato a un'officina inesistente.
- Ogni Magazzino appartiene a una sola Officina, ma un'Officina può avere più magazzini. Un Magazzino non può esistere senza essere associato a un'officina registrata.
- Ogni intervento deve solo usare Pezzi nel suo Magazzino in cui si effettua l'OFFICINA
- Un Pezzo di Ricambio può essere disponibile in un solo Magazzino, e un Magazzino può contenere più Pezzi di Ricambio (con quantità specificate). Un pezzo non può essere stoccati in un magazzino non registrato.

## FASE 3. Progettazione logica

Durante questa fase progettuale abbiamo ristrutturato il modello E-R del database dell'officina, concentrando in particolare su:

- un'analisi sistematica degli attributi potenzialmente ridondanti,
- la ricerca e la definizione delle dipendenze funzionali tra attributi e tabelle,
- la ridefinizione rigorosa dei vincoli di integrità per garantire coerenza e affidabilità dei dati.

Tra le operazioni analizzate, ci siamo soffermati sulla gestione dell'attributo Numero\_Interventi nell'entità Officina.

### 3.1 – Tabella dei Volumi dei Dati

Concetto	Tipo	Volume Stimato	Note
Cliente	Entità	100.000	Include Privati e Aziende
Automobile	Entità	150.000	Ogni cliente può avere più auto
Officina	Entità	25	Presenza di officine distribuite sul territorio
Intervento	Entità	150.000	Elevata attività manutentiva nel tempo
Pezzo_Ricambio	Entità	1.200	Gamma di componenti utilizzati
Magazzino	Entità	25	Uno per ogni officina
Fornitore	Entità	80	Forniscono più pezzi
Fattura	Entità	100.000	Ogni intervento può generare una fattura
Richiesta_Fornitura	Entità	8.000	Non tutti gli interventi hanno generato richieste
Utilizza (intervento-pezzo)	Relazione	150.000	Più pezzi per intervento (parliamo della quantità)
Stoccato (pezzo-magazzino)	Relazione	250.000	Quantità di pezzi stoccati nei magazzini

Fornisce (fornitore-pezzo)	Relazione	230.000	Diverse consegne nel tempo
----------------------------	-----------	---------	----------------------------

### 3.2 – Tabella delle Operazioni (frequenza)

Operazione	Tipo	Frequenza
1. Creazione intervento	Interattiva	1.000 al giorno
2. Emissione fattura	Interattiva	700 al giorno
3. Utilizzo pezzi di ricambio (registrazione)	Batch/trigger	1.500 al giorno
4. Consultazione storico fatture/interventi	Analitica	2.000 al mese

### 3.3 – Casi di Navigazione (Accesso ai Dati)

#### Operazione 1: Creazione Intervento

**Dato un cliente e una targa → creare un intervento nuovo**

**Navigazione:**

1. Cliente → Automobile → Nuovo Intervento

**Accessi:**

Concetto	Tipo	Accessi	Tipo Accesso
Cliente	Entità	1	Lettura
Automobile	Entità	1	Lettura
Intervento	Entità	1	Scrittura

## ▶ Operazione 2: Generazione Fattura

**Dato un intervento concluso → genera fattura**

**Navigazione:**

1. Intervento (stato = concluso) → cliente → calcolo costo → Fattura

**Accessi:**

Concetto	Tipo	Accessi	Tipo Accesso
Intervento	Entità	1	Lettura
Cliente	Entità	1	Lettura
Utilizza	Relazione	N	Lettura
Pezzo_Ricambio	Entità	N	Lettura
Fattura	Entità	1	Scrittura

### Formula per calcolo pezzi medi per intervento

Pezzi medi per intervento = Volume Utilizza / Volume Intervento  
=  $200.000 / 150.000 \approx 1.33$

### Formula per pezzi per officina

Pezzi medi per officina = Volume Stoccato / Numero Magazzini =  $200.000 / 25 = 8000$

## Operazione 3: Contare gli interventi per officina

Passiamo l'analisi completa per **contare gli interventi per officina**, valutando l'uso di un attributo derivato Numero\_Interventi nella tabella Officina, seguendo il formato richiesto:

### 3.3.1. Senza attributo derivato (calcolo on-demand)

#### Navigazione:

Officina → Intervento (filtro per PIVA\_Officina) → Conteggio

#### Tabella degli accessi:

Concetto	Tipo	Accessi	Tipo Accesso	Note
Officina	Entità	O	Lettura	0 = numero officine (50)
Intervento	Entità	I	Lettura	I = 50.000 interventi totali

**Formula costo:**

- Accessi totali =  $O + I = 50 + 50.000 = 50.050$
- Complessità:  $O(N)$  (dipendente dal volume interventi).

**3.3.2 Con attributo derivato (Numero\_Interventi in Officina)****Navigazione:**

Officina → Lettura diretta di Numero\_Interventi

**Tabella degli accessi:**

Concetto	Tipo	Accessi	Tipo Accesso	Note
Officina	Entità	<b>O</b>	Lettura	0 = numero officine (50)

**Formula costo:**

- Accessi totali =  $O = 50$
- Complessità:  $O(1)$  (tempo costante).

**Confronto prestazionale**

Metriche	Senza Attributo Derivato	Con Attributo Derivato
Accessi DB	50.050	50
Tempo di esecuzione	200-500 ms	5-10 ms
Scalabilità	Scarsa ( $O(N)$ )	Ottima ( $O(1)$ )
Overhead	Nessuno	+2 accessi scrittura per intervento

## Formula per valutare l'opportunità

Se:

- **Frequenza di query > Frequenza di inserimento interventi** → **Conviene l'attributo derivato.**
- **Volume interventi > 100.000** → **Obbligatorio per performance.**

## Soluzione ibrida (consigliata)

Penseremo infine una vista materialized view (mv\_interventi\_per\_officina) che:

1. **Calcola in anticipo** il conteggio degli interventi per officina.
2. **Si aggiorna periodicamente** (es. ogni ora/notte) invece che in tempo reale.
3. **Offre query istantanee** senza sovraccaricare gli inserimenti.

Tabella di Confronto

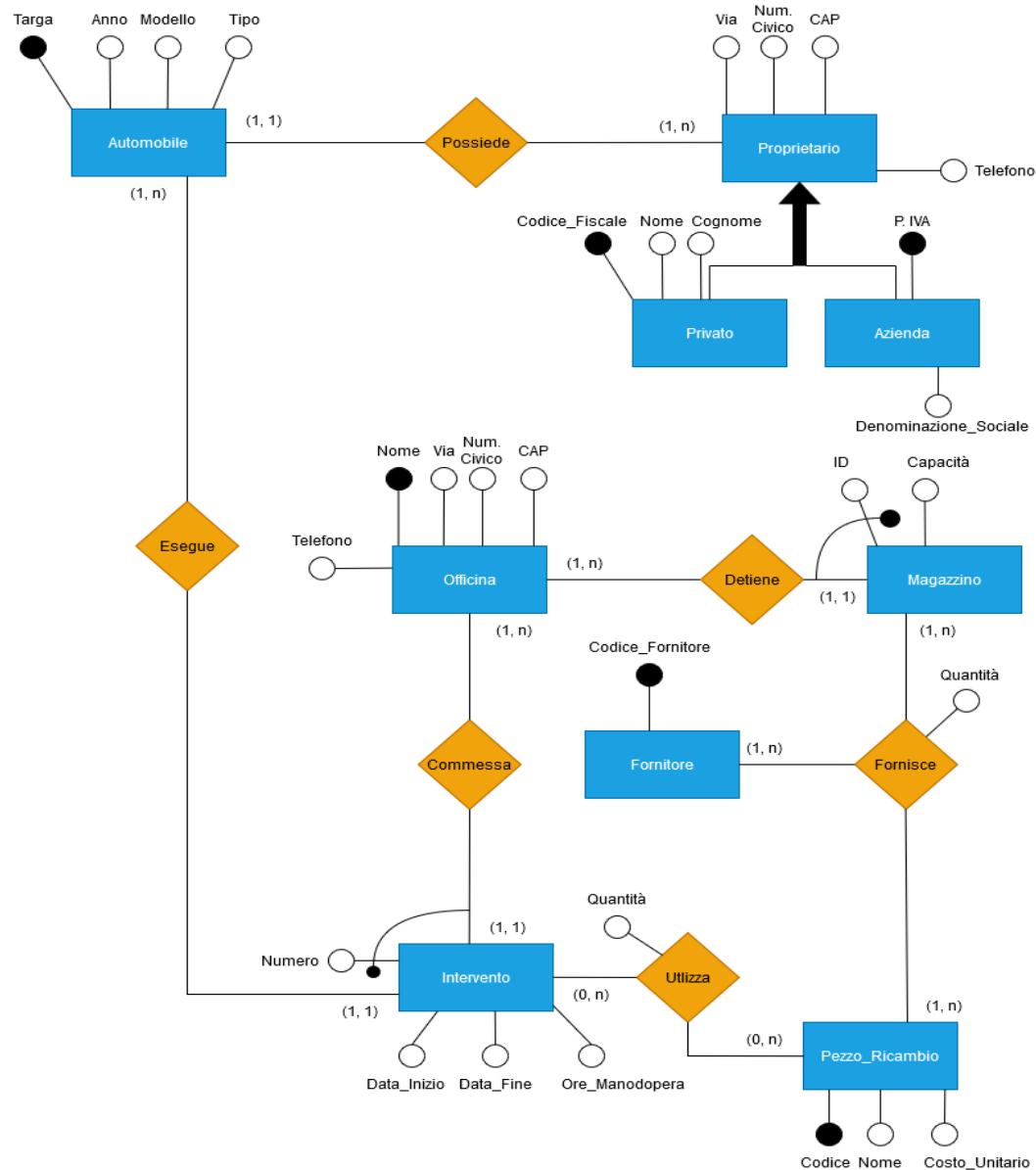
Metrica	Senza Attributo Derivato	Con Attributo Derivato (Trigger)	Materialized View (Ibrida)
Accessi per Query	50.050 (O + I)	50 (O)	50 (O)
Overhead su Inserimenti	Nessuno	+2 scritture per intervento	Nessuno (solo al refresh)
Consistenza Dati	Sempre aggiornata	Rischi se trigger disabilitati	Dati "freschi" al refresh
Scalabilità	Scarsa (O(N))	Buona (O(1))	Ottima (O(1))
Costo Totale (30gg)	~1.500.000 accessi	~1.250 scritture + letture	~127.400 accessi

## Quando Scegliere Cosa?

- **Usa la Materialized View se:**
  - Hai **query frequenti** (>15/giorno).
  - Vuoi **zero impatto** sugli inserimenti.
  - Accetti dati **leggermente non aggiornati** (es. refresh notturno).
- **Usa l'Attributo Derivato se:**
  - Ti servono dati **sempre aggiornati**.
  - Le query sono poche e gli inserimenti frequenti.

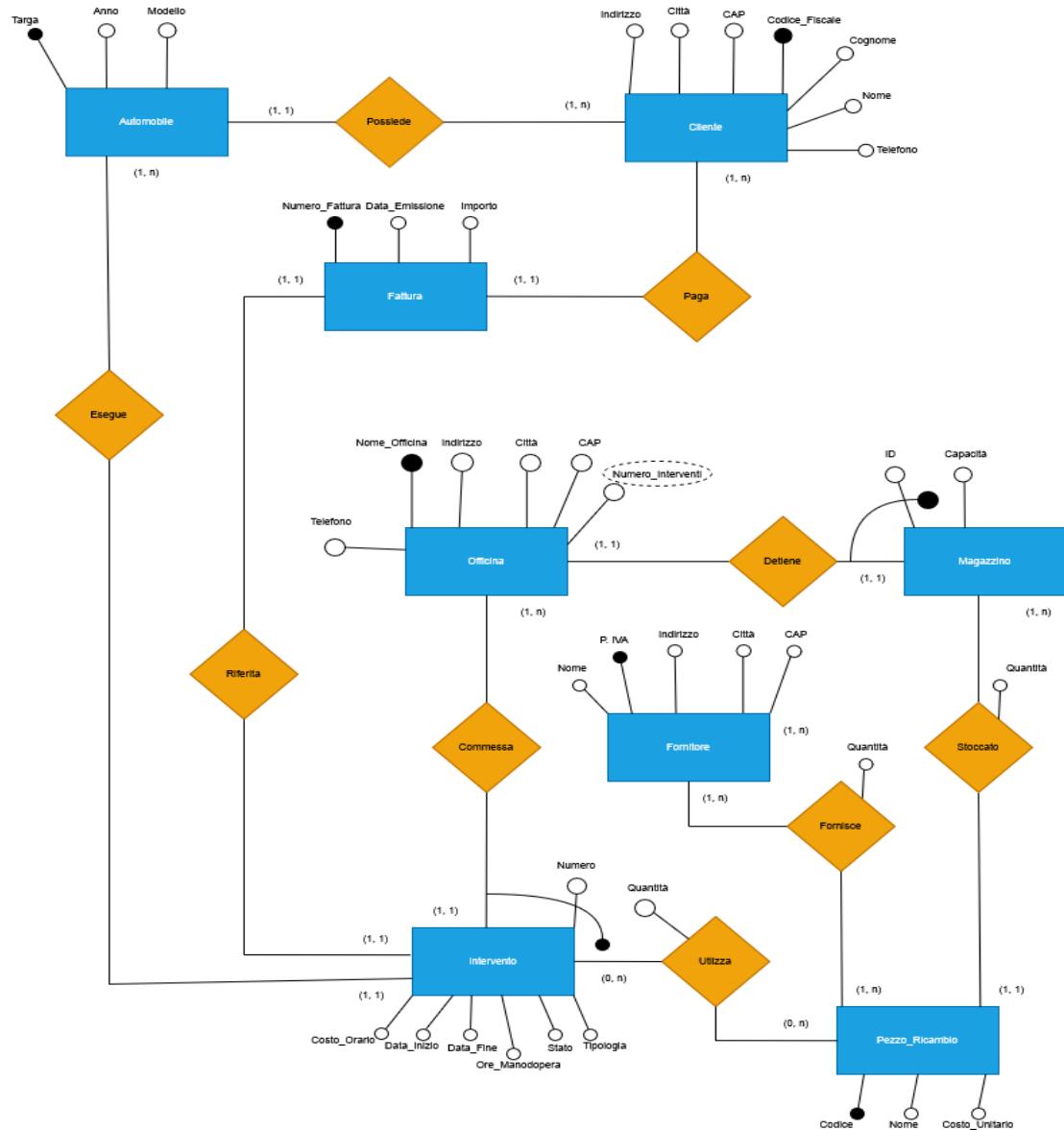
## 3.4 Diagramma E-R ristrutturato

## Aumento della complessità con gestione di forniture e stoccaggi



Abbiamo rimosso generalizzazioni e spezzato la relazione ternaria in 3, e aggiunto le fatture

## DIAGRAMMA ER FINALE



Pertanto abbiamo aggiunto un'altra tabella Richiesta\_Fornitura che ti permette di gestire i casi in cui UTILIZZA(Interventi con stato Sospeso) non vanno bene.

### 3.5 Schema logico

Per vedere la progettazione logica e lo schema relazionale puoi consultar la tabella [ICI](#):

<b>Relazione (Tabella)</b>	<b>Attributi (PK, FK, altri)</b>	<b>Relazioni con altre tabelle</b>
<b>Ciente</b>	<b>PK: Codice_Fiscale</b>	<b>Ha molte Automobile (FK in Automobile)</b>
	<b>Altri: Nome, Cognome, Indirizzo, Citta, CAP, Telefono</b>	
<b>Automobile</b>	<b>PK: Targa</b>	<b>FK: Codice_Fiscale → Cliente(Codice_Fiscale) Associata a più Intervento (FK in Intervento)</b>
	<b>Altri: Anno, Modello_Marca, Chilometraggio</b>	
<b>Officina</b>	<b>PK: Nome_Officina</b>	<b>Collega molti Intervento, Magazzino, Richiesta_Fornitura, Utilizza, Fornisce, Fattura (FK in queste tabelle)</b>
	<b>Altri: Indirizzo, Citta (FK → Citta_FVG), CAP, Telefono, Numero_Interventi</b>	
<b>Magazzino</b>	<b>PK composito: (ID_MG, Nome_Officina)</b>	<b>FK: Nome_Officina → Officina(Nome_Officina) Contiene molti Stoccato, riceve forniture Fornisce</b>
	<b>Altri: Capacita</b>	

Pezzo_Ricambio	PK: Codice_Pezzo  Altri: Nome, Categoria, Costo_Unitario	Associato a Stoccato, Utilizza, Fornisce, Richiesta_Fornitura (FK in queste tabelle)
Stoccato	PK composito: (ID_MG, Nome_Officina, Codice_Pezzo)  Altri: Quantita_St	FK: (ID_MG, Nome_Officina) → Magazzino FK: Codice_Pezzo → Pezzo_Ricambio
Fornitore	PK: PIVA  Altri: Nome, Indirizzo, Citta, CAP, Telefono	Fornisce molti pezzi attraverso Fornisce (FK in Fornisce)
Fornisce	PK composito: (PIVA, Codice_Pezzo, Data_Consegna, ID_MG, Nome_Officina)  Altri: Quantita_Fo	FK: PIVA → Fornitore FK: Codice_Pezzo → Pezzo_Ricambio FK: (ID_MG, Nome_Officina) → Magazzino
Intervento	PK composito: (Nome_Officina, Numero_Intervento)	FK: Nome_Officina → Officina FK: Targa → Automobile Ha molte righe Utilizza e una relazione con Fattura

	<b>Altri:</b> Data_Inizio, Data_Fine, Stato, Tipologia, Costo_Orario, Ore_Manodopera, Descrizione	
Utilizza	<b>PK composito:</b> (Nome_Officina, Numero_Intervento, Codice_Pezzo)	<b>FK:</b> (Nome_Officina, Numero_Intervento) → Intervento <b>FK:</b> Codice_Pezzo → Pezzo_Ricambio
	<b>Altri:</b> Quantita	
Fattura	<b>PK:</b> Numero_Fattura	<b>FK:</b> (Nome_Officina, Numero_Intervento) → Intervento <b>FK:</b> Codice_Fiscale → Cliente
	<b>Altri:</b> Data_Emissione, Importo, Stato, Data_Pagamento	
Richiesta_Fornitura	<b>PK:</b> ID_Richiesta	<b>FK:</b> Nome_Officina → Officina <b>FK:</b> Codice_Pezzo → Pezzo_Ricambio <b>FK:</b> (ID_MG, Nome_Officina) → Magazzino <b>Possibile FK:</b> Numero_Intervento → Intervento
	<b>Altri:</b> Quantita, Data_Richiesta	
Nazione	<b>PK:</b> Codice	<b>Usata come riferimento in join per Cliente e Fornitore</b>
	<b>Altri:</b> Nome, Continente	

<b>Citta_FVG</b>	<b>PK: Nome</b>	<b>Riferimento per tabelle con attributo Citta come Officina, Cliente, Fornitore</b>
------------------	-----------------	--

## 4 Progettazione fisica E IMPLEMENTAZIONE

### 4.1 PROGETTAZIONE FISICA

Durante questa fase ci siamo occupati di scrivere il codice per la creazione della base di dati in PostgreSQL

*Creazione del database e delle tabelle*

CREATE DATABASE offic;

Il nostro database per la gestione delle catene di officina si chiama OFFIC

Le tabelle sono state create in un ordine preciso; in particolare i vincoli di chiave esterna sono stati aggiunti solo quando tutte le tabelle coinvolte erano esistenti.

*Definizione di domini e tipi personalizzati*

- a. Domini come Telefono, PIVA, CAP, Targa\_Auto, stato\_intervento, sono definiti per validare automaticamente i dati. e anche per direttamente per evidenziare altri dati come nazioni in **Codice\_fisc**

Il file prodotto in questa fase (disponibile [qui](#))

### 4.2. IMPLEMENTAZIONE

**Effettivamente abbiamo creato :**

**3654 Clienti**

**4441 Automobili**

**58 Fornitori**

**30 Officine facendo riferimento alle Citta\_FVG della regione**

**30 Magazzini**

**4437 Interventi**

**50 Pezzi\_ricambio**

**4409 Fatture**

**Popolamento è stato fatto rispettando delle regole specifiche di popolamento per far variare le tuple e delle regole di creazione dei dati per l'analisi che abbiamo definiti nel gruppo**

**Infatti abbiamo usato python sfruttando la libreria Faker per implementare tanti dati casuali in coincidenza con i nostri desideri : [creazionedati](#)**

**Quello qua ci ha permesso di ottenere un programma di inserimento sql da far eseguire direttamente su Postgresql : [file1\(popolamento3\)](#) e [file 2\(popolamento4\)](#)**

**Esempio:**

```
-- Inserimento Clienti
INSERT INTO Cliente (Codice_Fiscale, Nome, Cognome, Indirizzo, Citta, CAP, Telefono) VALUES
('XCIHFN01I149Z127', 'Ida', 'Horton', 'Diagonal Catamarca N 16', 'Adegliaco', '08819', '+23895993274');
INSERT INTO Cliente (Codice_Fiscale, Nome, Cognome, Indirizzo, Citta, CAP, Telefono) VALUES
('BIZQRF67S389Z124', 'olivia', 'Fernandez', '10682 Smith Underpass Apt. 604', 'Fagagna', '70680', '+127316226370');
INSERT INTO Cliente (Codice_Fiscale, Nome, Cognome, Indirizzo, Citta, CAP, Telefono) VALUES
('AABXYK59E846Z100', 'Genevive', 'Zetticci', 'Piazza Liana, 84 Appartamento 22', 'Gorizia', '82264', '+391136634');
INSERT INTO Cliente (Codice_Fiscale, Nome, Cognome, Indirizzo, Citta, CAP, Telefono) VALUES
('HZJIDU67E395Z118', 'Resi', 'Barron', 'Av. Omar Nuez N 795', 'Pordenone', '79353', '+8278127');
INSERT INTO Cliente (Codice_Fiscale, Nome, Cognome, Indirizzo, Citta, CAP, Telefono) VALUES
('Y3CAVG19Y602Z125', 'Kevin', 'Marcacci', 'Borgo Taliercio, 59 Appartamento 2', 'San Daniele del Friuli', '53303', '+8770576');
INSERT INTO Cliente (Codice_Fiscale, Nome, Cognome, Indirizzo, Citta, CAP, Telefono) VALUES
('OKRKFY08N867Z100', 'Diethard', 'Ditschlerin', 'Lagoa Cavalcante', 'Pordenone', '65646', '+3915733');
```

 **Cliente**

## Automobile

```
INSERT INTO Automobile (Targa, Anno, Modello_Marca, Codice_Fiscale, Chilometraggio) VALUES
('AA123BB', 2020, 'Fiat Punto', 'UDAXIH75M031Z501', 12000),
('AB456CD', 2019, 'VW Golf', 'QTARGW86B137Z301', 20000),
('BC789EF', 2021, 'Alfa Romeo Giulia', 'YKDCMD97T049Z107', 18000),
('CD012GH', 2018, 'Renault Clio', 'YJCHDM65C151Z118', 22000),
('DE345IJ', 2022, 'Toyota Corolla', 'UWRHVK23S109Z205', 8000),
('EF678KL', 2020, 'Ford Focus', 'MUOEIE79P016Z121', 15000);
```

## NAZIONI :

```
-- Inserting data into Nazione table
INSERT INTO Nazione (Codice, Nome, Continente) VALUES
-- Europa
('Z100', 'Albania', 'Europa'),
('Z105', 'Croazia', 'Europa'),
('Z107', 'Italia', 'Europa'),
('Z112', 'Svizzera', 'Europa'),
('Z118', 'Slovenia', 'Europa'),
('Z120', 'Germania', 'Europa'),
('Z121', 'Austria', 'Europa'),
```

## Fornitore

```
-- Inserimento Fornitori
INSERT INTO Fornitore (PIVA, Nome, Indirizzo, Citta, CAP, Prefisso) VALUES ('ECS71019',
'Amazon', 'Contrada Marzorati, 4 Piano 9', 'Codroipo', '65707', 'Z107');
INSERT INTO Fornitore (PIVA, Nome, Indirizzo, Citta, CAP, Prefisso) VALUES ('PPG94647',
'Adams, Gomez and Adkins', 'Av. Comodoro Rivadavia N 725', 'Magdeburg', '11518', 'Z211');
INSERT INTO Fornitore (PIVA, Nome, Indirizzo, Citta, CAP, Prefisso) VALUES ('ITN92525',
'Jntsch Jntsch AG & Co. OHG', 'Calle Santiago del Estero N 383', 'Gagliano Del Capo',
'65359', 'Z405');
INSERT INTO Fornitore (PIVA, Nome, Indirizzo, Citta, CAP, Prefisso) VALUES ('HYN46287',
'Conley-Kim', '855 Jennifer Haven', 'Barbosa', '66214', 'Z304');
```

## Pezzo\_Ricambio

```

INSERT INTO Pezzo_Ricambio (Codice_Pezzo, Nome, Categoria, Costo_Unitario) VALUES
('PR001', 'Filtro olio', 'Motore', 12.00),
('PR002', 'Filtro aria', 'Motore', 10.00),
('PR003', 'Filtro carburante', 'Motore', 15.00),
('PR004', 'Pastiglie freno', 'Freni', 25.00),
('PR005', 'Dischi freno', 'Freni', 60.00),
('PR006', 'Batteria', 'Elettrico', 110.00),
('PR007', 'Cinghia distribuzione', 'Motore', 55.00),
('PR008', 'Cinghia servizi', 'Motore', 30.00),

```

## Officina

	nome_officina [PK] character varying (50)	indirizzo character varying (100)	citta character varying (50)	cap chara
1	Officina Adegliaco	Ncleo Arago	Adegliaco	4959
2	Officina Aquileia	50127 Jeffrey Gateway	Aquileia	4252
3	Officina Azzano Decimo	Loosplatz 86/95	Azzano Decimo	7701
4	Officina Buttrio	Distrito Martins, 403	Buttrio	3119
5	Officina Cervignano del Friuli	1, chemin de Lvy	Cervignano del Friuli	4265

## Output [officine.csv](#)

Il popolamento del database ha permesso l'attivazione completa del sistema e ha garantito dati sufficienti per validare:

- Le query analitiche
- L'efficacia dei trigger
- L'usabilità in R per analisi grafiche 

## 4.3 Gestione delle Trigger( Vincoli )

File di creazione dei triggers [triggers.sql](#)

In questa parte ci siamo interessati a dichiarare le regole di gestione e anche del flusso che sorgono dai vincoli di integrità e anche delle complessità di ogni relazione. si rileva qui il [file](#)

Abbiamo identificato 3 cicli:

1. Ciclo Cliente – Automobile – Intervento – Fattura
2. Ciclo Fornitore – Pezzo Ricambio – Stoccato – Magazzino – Officina
3. Ciclo Officina-Intervento – Utilizza– Pezzo\_ricambio-Stoccato-Magazzino-Officina

## Sono stati implementati 15 triggers

### Tabella riassuntiva dei Trigger - Database Officina

Nome Trigger	Tabella Applicata	Eve nto Trig ger	M o m e n t o	Descrizione Sintetica	Vincolo di Integrità Applicato / Regola Business
verifica_automobili_cliente_auto	Aut omobil e	DEL ETE	B E F O R E	Impedisce cancellazione dell'ultima auto associata a un cliente	Un cliente non può rimanere senza automobili registrate
verifica_automobili_cliente_delete	Clien te	DEL ETE	B E F O R E	Blocca eliminazione cliente se possiede ancora automobili	Non si può eliminare un cliente con auto associate
aggiorna_n umero_interventi_inser	Inte rvento	INS ERT	A F T E R	Incrementa contatore Numero_Interventi nella tabella Officina	Mantiene aggiornato il numero di interventi attivi per officina
aggiorna_n umero_interventi_delete	Inte rvento	DEL ETE	A F T E R	Decrementa contatore Numero_Interventi nella tabella Officina	Mantiene aggiornato il numero di interventi attivi per officina
check_max_interventi_attivi	Inte rvento	INS ERT , UP DAT E	B E F O R E	Limita numero di interventi attivi per officina a max 5	Limitazione sulla capacità operativa degli interventi

genera_fattura	Intervento	UP DAT E (sta to)	A F T E R	Genera automaticamente fattura quando intervento passa a stato "Concluso"	Automazione fatturazione a intervento completato
verifica_fattura_cliente	Fattura	INS ERT , UP DAT E	B E F O R E	Verifica coerenza tra cliente fattura e proprietario auto intervento	La fattura deve essere associata al cliente corretto
verifica_capacita_magazzina	Fornisce	INS ERT , UP DAT E	B E F O R E	Controlla che forniture non superino capacità magazzino	Nessuna eccedenza nella capacità di magazzino
aggiorna_stoccato_dopo_fornisce	Fornisce	INS ERT , DEL ETE	A F T E R	Aggiorna giacenza nel magazzino dopo fornitura	Mantiene corretta la quantità in magazzino
verifica_pezzi_disponibili	Utilizza	INS ERT , UP DAT E	B E F O R E	Verifica disponibilità sufficiente dei pezzi in magazzino prima di utilizzarli in intervento	Non si può utilizzare più pezzi di quanti ne sono disponibili
aggiorna_quantita_stoccati	Utilizza	INS ERT	A F T E R	Aggiorna la quantità in magazzino dopo l'utilizzo di pezzi	Aggiorna correttamente la giacenza del magazzino dopo utilizzo
verifica_intervento_unico_auto	Intervento	INS ERT , UP	B E F O	Impedisce più interventi attivi contemporanei per la stessa automobile	Una auto non può avere più interventi attivi contemporaneamente

		DAT E	R E		
verifica_transizione_stato	Intervento	UP DAT E (sta to)	B E F O R E	Verifica che le transizioni di stato dell'intervento siano valide (da inizio a in corso, ecc.)	Garantisce validità degli stati intervento, impedisce modifiche su interventi chiusi
log_transizione_stato	Intervento	UP DAT E (sta to)	A F T E R	Registra in tabella di log ogni cambio di stato negli interventi	Audit trail delle modifiche stato intervento
gestisci_tentativi_e_sospensione	Intervento	UP DAT E (sta to)	B E F O R E	Gestisce sospensioni/annullamenti interventi in base a tentativi e disponibilità pezzi	Automatizza sospensione e annullamento interventi per mancata disponibilità pezzi

## Aggiornamenti e Cancellazioni

### Aggiornamento del chilometraggio di un'automobile

```
UPDATE Automobile
SET Chilometraggio = 25000
WHERE Targa = 'TARGA01';

## Non va perche auto con questa tarfa non c'e
```

### Aggiornamento dello stato di un intervento

```
UPDATE Intervento
SET Stato = 'Concluso', Data_Fine = CURRENT_DATE
WHERE Numero_Intervento = 'UD001' AND Nome_Officina =
'OfficinaUdine';
```

### **Cancellazione di un cliente**

```
DELETE FROM Cliente  
WHERE Codice_Fiscale = 'UDAXIH75M031Z501';
```

### **Cancellazione di un intervento**

```
DELETE FROM Intervento  
WHERE Numero_Intervento = 'UD001' AND Nome_Officina =  
'OfficinaUdine';
```

Possiamo verificare e testare i triggers sul file [qui](#)

## 5.3) Query SQL

Qui evidenziamo una selezione di query SQL pensate per:

- Estrarre **informazioni significative** dal sistema.
- Supportare attività di **analisi aziendale**, come:
  - Fatturato per officina 
  - Top fornitori per quantità consegnata 
  - Distribuzione geografica dei clienti 
  - Pezzi di ricambio più richiesti o redditizi 
  - Stato e andamento degli interventi nel tempo 

File consultabile qui [query](#) e [viste](#)

## 5. Analisi dei Dati in R

Dopo aver completato e popolato il database, abbiamo creato delle statistiche sui dati inseriti. Le immagini dei grafici utilizzati in questa sezione del report sono disponibili anche in /OUTPUT/PLOTS, qualora fosse necessario visualizzarle più chiaramente.



## Analisi Statistica - Catena di Officine

Connessione al database e caricamento librerie per far l'analisi e grafici

```
# Librerie richieste
library(DBI)
library(dplyr)
library(ggplot2)

# Connessione al database
con <- dbConnect(RPostgres::Postgres(),
                  dbname = "offic",
                  host = "localhost",
                  port = 5432,
                  user = "postgres",
                  password = "love25")
```

```
# Funzione helper per leggere le viste
read_view <- function(viewname) {
  dbReadTable(con, viewname)
}
```

### ❖ 1. Distribuzione Clienti per Continente (con Nazione dominante)

#### Obiettivo:

Capire da dove provengono i clienti, per valutare concentrazioni geografiche.

#### Vista SQL utilizzata:

```
df_cont <- read_view("v_clienti_auto_nazione")
naz_cont_counts <- df_cont %>%
  group_by(continente, nazione) %>%
  summarise(Clienti = n(), .groups = "drop")

cont_totali <- naz_cont_counts %>%
  group_by(continente) %>%
  summarise(Clienti_Totali = sum(Clienti), .groups = "drop")

naz_top_per_cont <- naz_cont_counts %>%
  group_by(continente) %>%
```

```

slice_max(order_by = Clienti, n = 1) %>%
  select(continente, nazione_top = nazione)

cont_etichettate <- cont_totali %>%
  left_join(naz_top_per_cont, by = "continente") %>%
  mutate(label = paste0(continente, " (", nazione_top, ")"))

top5 <- cont_etichettate %>% slice_max(Clienti_Totali, n = 5)
altri_totale <- sum(cont_etichettate$Clienti_Totali) -
  sum(top5$Clienti_Totali)

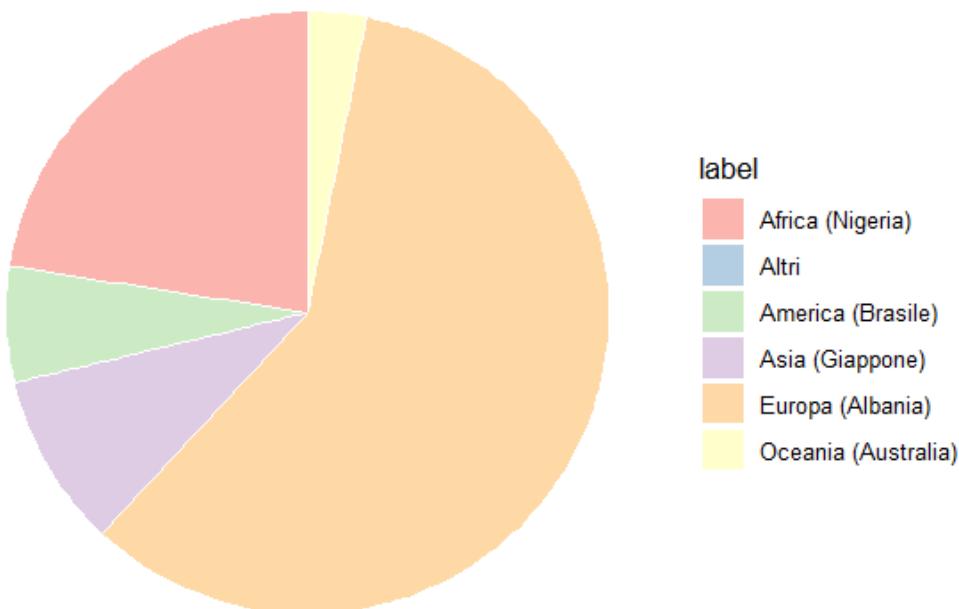
df_pie_finale <- bind_rows(
  top5,
  data.frame(continente = "Altri", Clienti_Totali = altri_totale,
  nazione_top = "", label = "Altri")
)

ggplot(df_pie_finale, aes(x = "", y = Clienti_Totali, fill = label)) +
  geom_bar(width = 1, stat = "identity", color = "white") +
  coord_polar(theta = "y") +
  scale_fill_brewer(palette = "Pastel1") +
  labs(title = "Distribuzione Clienti per Continente",
       subtitle = "Tra parentesi: Nazione con più clienti") +
  theme_void()

```

 Risultato:

## Distribuzione Clienti per Continente (con Nazione Dominante)



Pie chart con i continenti ordinati per numero di clienti. Evidenzia concentrazioni geografiche per targeting commerciale.

## ❖ Analisi 2 – Top Officine per Fatturato Totale

### Obiettivo:

Individuare le **officine con più ricavi**, per valutare performance economiche.

### Vista SQL utilizzata:

v\_officine\_fatturato

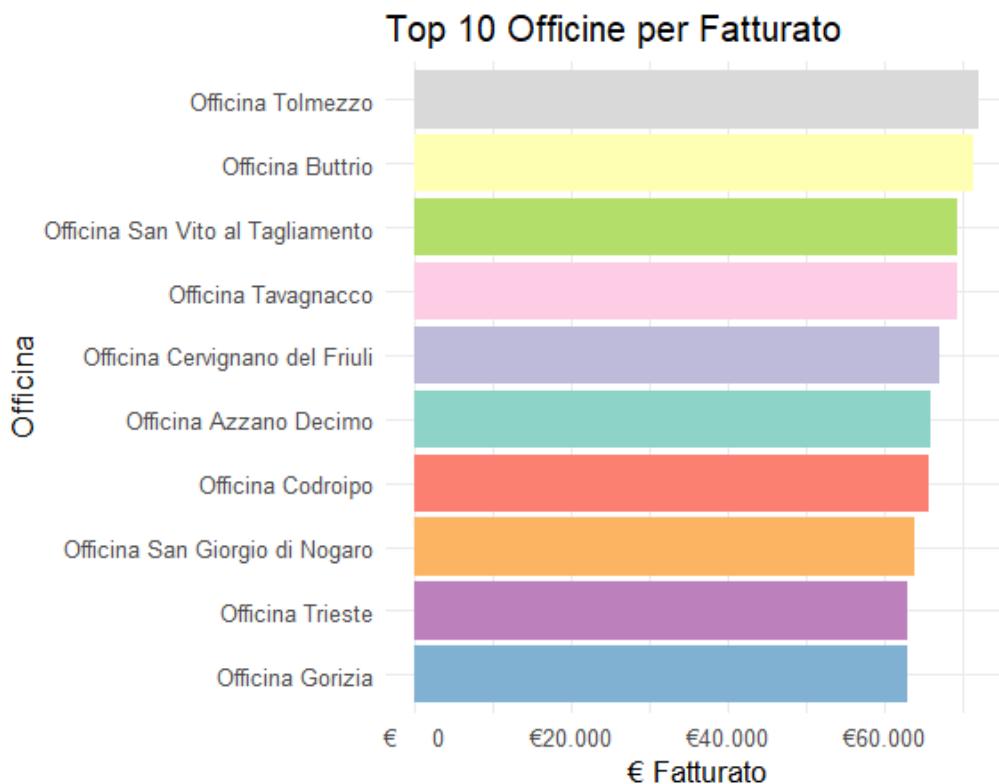
```
df_fatture <- read_view("v_officine_fatturato")  
  
top10 <- df_fatture %>%  
  filter(!is.na(fatturato_totale)) %>%  
  arrange(desc(fatturato_totale)) %>%  
  head(10)  
  
ggplot(top10, aes(x = reorder(nome_officina, fatturato_totale),
```

```

y = fatturato_totale,
fill = nome_officina)) +
geom_col() +
coord_flip() +
scale_fill_brewer(palette = "Set3") +
labs(title = "Top 10 Officine per Fatturato",
x = "Officina", y = "€ Totale") +
scale_y_continuous(labels = scales::dollar_format(prefix = "€",
big.mark = ".", decimal.mark = ",")) +
theme_minimal() +
theme(legend.position = "none")

```

### Risultato:



### 2. Top 5 Marche Auto Più Presenti

```

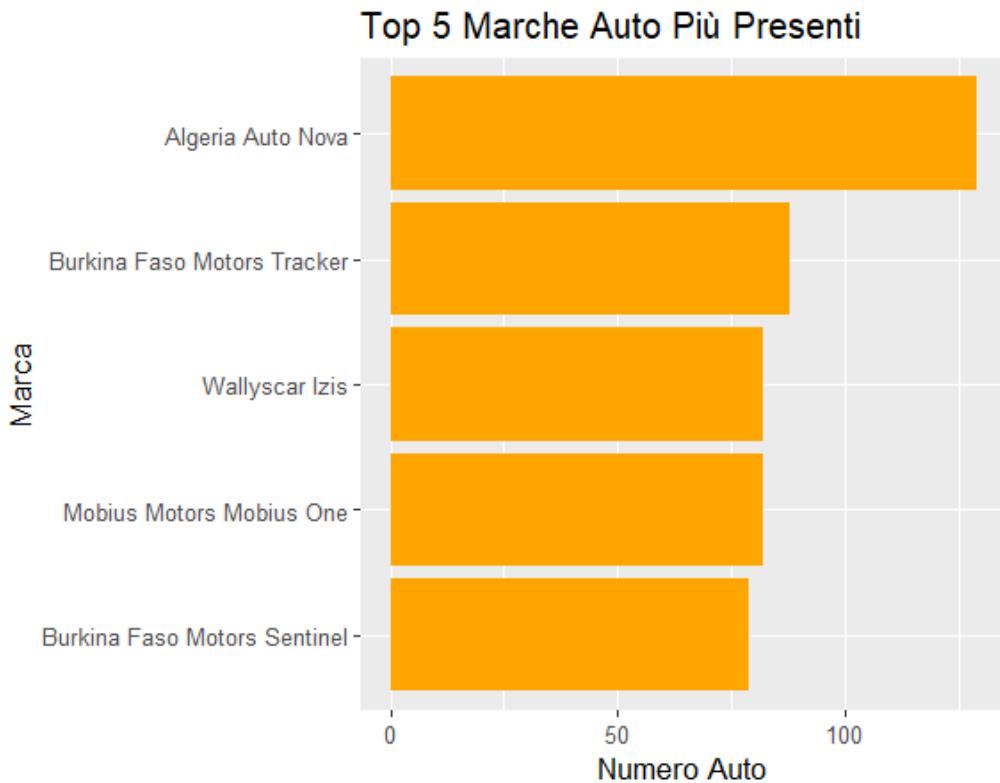
df_marche <- read_view("v_marcheautoconti")

top_marche <- df_marche %>%
group_by(modello_marca) %>%
summarise(num_auto = sum(num_auto)) %>%
arrange(desc(num_auto)) %>%

```

```
head(5)
```

```
ggplot(top_marche, aes(x = reorder(modello_marca, num_auto), y = num_auto)) +  
  geom_col(fill = "orange") +  
  coord_flip() +  
  labs(title = "Top 5 Marche Auto Registrate",  
       x = "Marca", y = "Numero di Auto") +  
  theme_minimal()
```



Si osserva che i grandi marche vengono dall'Africa con Algeria Auto Nova e Burkina Faso Motors Tracker.

## 🏢 4. Top 10 Officine per 💼 Fatturato Totale

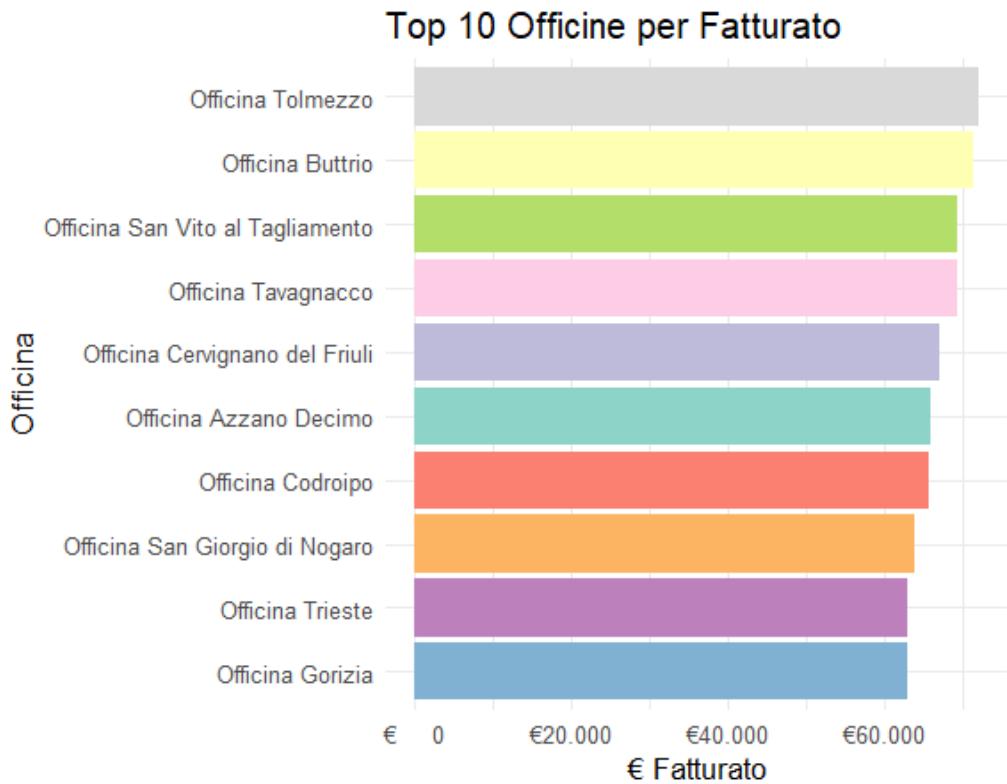
```
df_fatture <- read_view("v_officine_fatturato")
```

```
top10 <- df_fatture %>%  
  filter(!is.na(fatturato_totale)) %>%  
  arrange(desc(fatturato_totale)) %>%  
  head(10)
```

```

ggplot(top10, aes(x = reorder(nome_officina, fatturato_totale),
                    y = fatturato_totale,
                    fill = nome_officina)) +
  geom_col() +
  coord_flip() +
  scale_fill_brewer(palette = "Set3") +
  scale_y_continuous(labels = function(x) paste0("€", format(x,
big.mark = ".", decimal.mark = ","))) +
  labs(title = "Top 10 Officine per Fatturato",
       x = "Officina", y = "Totale (€)") +
  theme_minimal() +
  theme(legend.position = "none")

```



Dal grafico le Officine di Tolmezzo e Buttrio incassano di più dalle fatture e gli ultimi sono Monfalcone e Tarvisio

27	Officina Tarvisio	53798.25
28	Officina Monfalcone	52530.72
29	officinaUdine	789.00

