

Documentazione di progetto per il corso di Basi di Dati

Aleandro Prudeniano - Giacomo Sansone

1.	Considerazioni di carattere generale	5
2.	Glossario	5
1.	Area Produzione	5
2.	Area Vendita	7
3.	Area Assistenza	9
4.	Area Smontaggio.....	10
3.	Diagramma ER: spiegazione dettagliata.....	12
1.	Area produzione	12
2.	Area vendita	18
3.	Area assistenza	24
4.	Area smontaggio	29
4.	Ristrutturazione del diagramma ER	32
1.	Rimozione delle generalizzazioni	32
2.	Conversione degli attributi composti e multi-valore	35
5.	Individuazione delle operazioni e analisi delle prestazioni	36
1.	Tavola dei volumi.....	37
2.	Operazioni scelte.....	43
3.	Analisi costi delle operazioni e introduzione di ridondanze	44
6.	Progettazione Logica	74
1.	Schema logico	74
2.	Vincoli di integrità referenziale.....	76
3.	Vincoli di integrità generici	78
7.	Analisi delle dipendenze funzionali con normalizzazione.....	80
1.	Dipendenze funzionali.....	80
8.	Implementazione su DBMS Oracle MySQL	86
1.	Indice del file Operazioni.sql.....	87
9.	Data analytics.....	90
1.	Diagnosi intelligente dei guasti: CBR	90
2.	Efficienza del processo.....	96
10.	API.....	105

1. Considerazioni di carattere generale

Il seguente lavoro costituisce la documentazione al progetto del sistema informativo per l'azienda eDevice. Per un corretto uso del prodotto, è sicuramente necessario avere chiara la sua struttura, così come le scelte che hanno portato a preferire una certa opzione piuttosto che un'altra. Per tale motivo, nella documentazione si riportano tutte le fasi del progetto, così come un attento commento a quelle che sono state le situazioni che hanno condotto ad un dibattito durante la realizzazione del database.

Poiché le macro-aree dell'azienda risultano ben definite e poco sovrapposte, abbiamo scelto di individuare ciascuna di queste con un colore. In tal modo sarà possibile capire immediatamente a quale delle quattro zone ci si riferisce, specialmente all'interno del diagramma ER (che non sarà riportato in maniera integrale nel testo). Per un fattore di gusto personale, abbiamo scelto il **verde** per l'area produzione, l'**arancione** per l'area vendita, il **blu** per l'area assistenza e il **grigio** per l'area smontaggio.

2. Glossario

Nello scrivere il glossario, aggiornato e riveduto durante la realizzazione del progetto per evitare possibili fraintendimenti, abbiamo notato che spesso non era naturale trovare sinonimi per i termini, anche sulla base del testo del progetto. Ciononostante, siamo convinti che questo non influenzi la comprensione, soprattutto a causa dei significati 'standard' dei termini che entrano in gioco. Detto questo, ulteriori dettagli sulle varie espressioni (riguardanti più l'interpretazione che non il significato), saranno introdotti dove necessario.

1. Area Produzione

Termine	Descrizione	Sinonimi	Collegamenti
Prodotto Elettronico	Prodotto realizzato dall'azienda. Può avere varianti.	Prodotto, unità	Parte, Lotto, Ordine, Garanzia, Formula di estensione della Garanzia, Faccia
Faccia	Lato su cui si può appoggiare l'oggetto in fase di assemblaggio.		Operazione, Prodotto

Parte	Singola componente di un prodotto.	Pezzo	Prodotto Elettronico, Materiale, Elemento di giunzione, Operazione
Materiale	Compongono ciascuna parte.		Parte
Elemento di giunzione	Permette di tenere assieme le parti, in alternativa alla saldatura. Ciascuno ha un tipo.		Parte, Operazione
Operazione	Permette di aggiungere una parte ad un prodotto durante l'assemblaggio.		Prodotto Elettronico, Parte, Elemento di Giunzione, Utensile, Sequenza di operazioni, Stazione
Vincolo di precedenza tecnologica	Vincolo che determina quale operazione ha priorità rispetto ad altre.		Operazione, Sequenza di operazione
Utensile	Strumento con cui si esegue un'operazione.		Operazione, Operatore
Sequenza di operazioni	Operazioni consecutive attraverso le quali si ottiene il prodotto terminato. Non è unica.	Linea di Assemblaggio, Sequenza	Prodotto Elettronico, Operazione, Indicatore di performance
Lotto	Partita di prodotto. A ciascun lotto è associata una sequenza di operazioni con le quali sono prodotti tutti i componenti.		Prodotto Elettronico, Sequenza di operazioni, Magazzino, Ubicazione, Ordine

Stazione	Fase della linea produttiva in cui sono eseguite una o più operazioni, tutte senza rotazioni.	Operazioni sequenziali	Operazione, Operatore, Utensile
Operatore	Impiegato addetto ad una stazione.		Stazione, Utensile
Unità persa	Prodotto per cui in una stazione non si sono realizzate tutte le operazioni stabilite.		Stazione, Prodotto
Indicatore di performance	Permette di avere una misura oggettiva dell'efficienza della produzione.		Sequenza di operazioni
Magazzino	Luogo fisico dove sono stoccati i lotti realizzati. Ciascuno è predisposto ad una certa classe di prodotti.		Ubicazione
Ubicazione	Suddivisione del magazzino.		Magazzino, Lotto

2. Area Vendita

Termine	Descrizione	Sinonimi	Collegamenti
Utente	Persona fisica che si iscrive al sito.		Account, Ordine, Spedizione
Account	Profilo associato ad ogni utente del sito.		Utente, Spedizione

Ordine	È effettuato da un utente per l'acquisto dal sito.		Utente, Prodotto Elettronico, Recensione, Lotto, Difetto
Stato dell'ordine	Indica l'attuale stato dell'ordine, tra: <i>in processazione, in preparazione, spedito, evaso e pendente.</i>		Ordine
Spedizione	Procedura che permette l'arrivo di un acquisto al destinatario.		Ordine, Utente, Hub
Hub	Luogo fisico dove sosta il prodotto spedito prima della consegna.	Centro di smistamento	Spedizione
Recensione	Punteggio facoltativo che il cliente assegna all'esperienza di acquisto.		Ordine
Garanzia	Copertura a guasti.		Prodotto Elettronico, Guasto
Formula di estensione di garanzia	Prodotto che permette di estendere la garanzia.		Guasto, Prodotto
Difetto	Imperfezione fisica del prodotto.		Ordine, Motivazione
Richiesta di reso	Richiesta da parte del cliente di restituire un prodotto ricevendo l'importo speso.		Ordine, Account, Motivazione, Unità Rese
Motivazione	Causa del reso.		Richiesta di reso, Difetto,

Unità rese	Unità tornate in azienda a causa di una richiesta di reso.	Richiesta di reso
-------------------	--	-------------------

3. Area Assistenza

Termine	Descrizione	Sinonimi	Collegamenti
Guasto	Anomalia nel normale funzionamento di un prodotto.		Cliente, Prodotto elettronico, Codice di errore
Codice di errore	Segnale inviato dal prodotto che specifica il tipo di guasto.		Prodotto elettronico, Guasto
Rimedio	Sequenza di operazioni da eseguire per sopperire ad un guasto.		Prodotto elettronico, Guasto, Codice di errore
Quesito di diagnosi	Domanda da porre al cliente che richiede assistenza virtuale, a cui si può rispondere solo "sì" o "no".	Domanda	Prodotto elettronico, Rimedio
Intervento	Intervento di assistenza che può essere eseguito in un centro di assistenza o presso il domicilio del cliente. Può altresì essere un sopralluogo con o senza prelievo, una riparazione o una consegna.		Centro di assistenza, Prodotto elettronico, Guasto, Rimedio, Tecnico
Centro di assistenza	Luogo fisico nel quale può		Tecnico

	avvenire un intervento di assistenza.		
Tecnico	Addetto alla risoluzione di guasti in assistenza fisica.	Riparatore	Centro di assistenza
Preventivo	Prima stima da parte del tecnico del costo per la riparazione.		Intervento, Tecnico
Ordine parti di ricambio	Ordine di parti di ricambio effettuato da un tecnico per eseguire una assistenza presso domicilio.		Tecnico, intervento, parte di ricambio
Parte di ricambio	Componente usata per sostituirla una danneggiata durante un intervento di riparazione.	Pezzo di ricambio	Ordine parti di ricambio, prodotto
Ricevuta fiscale	Ricevuta emessa alla fine dell'intervento di riparazione.		Tecnico, intervento
Metodo di pagamento	Metodo di pagamento usato per pagare una ricevuta fiscale.		Ricevuta fiscale, Intervento

4. Area Smontaggio

Termine	Descrizione	Sinonimi	Collegamenti
Prodotto ricondizionato	Prodotto danneggiato che è stato riparato ed è stato reimpresso	Prodotto refurbished	Prodotto elettronico, Caratteristiche

	nel mercato con un prezzo minorato.		
Lotto ricondizionato	Insieme di prodotti ricondizionati.		Prodotto elettronico
Controllo generale	Serie di test eseguiti su un prodotto da ricondizionare.		Test, Prodotto elettronico
Test	Test eseguito su un prodotto elettronico per accertarne il funzionamento in una data caratteristica.		Controllo generale, prodotto elettronico, Parte
Test tree	Gerarchia dei test da eseguire su un prodotto e le sue componenti per dichiararne l'effettiva funzionalità.	Albero dei test	Test, Controllo generale, Prodotto
Politica di sostituzione	Indice percentuale di test falliti che porta alla totale sostituzione dei un pezzo del prodotto.		Test, Parte, Prodotto

3. Diagramma ER: spiegazione dettagliata

In questa fase della documentazione, introduciamo prima la tabella dove si elenca, per ciascuna area, i nomi delle entità, gli attributi e gli identificatori per come appaiono nel diagramma ER completo, poi le relazioni, con nome, eventuali attributi e le entità che vengono collegate; dopo, procediamo con la spiegazione della semantica dello schema ER associato, poiché da solo non basta per comprendere le numerose sfumature adottate.

1. Area produzione

Entità	Attributi	Identificatore
Prodotto	IDProdotto, Marca, Modello, Nome, Soglia ricondizionamento (SdR), Prezzo, DataProduzione, Predisposizione	IDProdotto
Variante	IDVariante, Prezzo, Descrizione	IDVariante
Variante di prodotto	Codice seriale, Ricondizionata	Codice seriale
Parte	Codice, Nome, Prezzo, Peso, Coefficiente di svalutazione (CdS), intermedio	Codice
Materiale	IDMateriale, Nome, Valore al kg(V/KG)	IDMateriale
Giunzione	IDGiunzione, Nome, Caratteristica (Descrizione, Unità di Misura (UdM), Valore)	IDGiunzione
Operazione	IDOperazione, Nome, Priorità	IDOperazione
Utensile	IDUtensile, Nome	IDUtensile
Faccia	IDFaccia	IDFaccia
Sequenza di operazioni	IDSequenza, Indicatore, Tempo stazione	IDSequenza
Lotto	Codice, Data di produzione prevista, Data di produzione effettiva, Sede di produzione, Quantità (Qta.)	Codice

Lotto nuovo

Stazione	IDStazione, Precedenza	IDStazione
Operatore	Badge, Nome, Cognome	Badge
Operazioni Campione	Nome	Nome
Ubicazione	IDUbicazione, Scaffale,Piano, Stanza	IDUbicazione
Magazzino	Codice, Capienza, Predisposizione Peso (Pred.)	Codice

Relazione	Attributi	Entità coinvolte
Aggiunta		Prodotto, Variante
Variazione		Prodotto, Variante di Prodotto
Modello		Variante, Variante di Prodotto
Componente		Prodotto, Parte
Composizione	Quantità	Parte, Materiale
Assemblaggio		Operazione, Parte, Parte
Unione		Operazione, Giunzione
Uso		Operazione, Utensile
Appoggia		Operazione, Faccia
Posizione		Prodotto, Faccia
Fase		Operazione, Stazione
Catena		Stazione, Sequenza Di Operazioni
Unità perse	Quantità	Stazione, lotto nuovo
Lavora		Stazione, Operatore
Capacità	Tempo medio, Varianza	Operazione campione, operatore
Assemblaggio		Sequenza di operazioni, Lotto nuovo
Sequenze possibili		Sequenza di operazioni, Prodotto
Stoccaggio storico	Data inizio, Data fine	Lotto, Ubicazione
Stoccaggio attuale	Data inizio	Lotto, Ubicazione
Compone		Ubicazione, Magazzino
Produzione		Lotto, Variante di Prodotto
Modello		Variante, variante di prodotto

Prodotto

Come è ragionevole fin dall'incipit del testo del progetto, il *prodotto* è l'elemento attorno a cui ruota l'intera base di dati. Proprio per questo motivo, abbiamo a lungo dibattuto su cosa dovesse contenere tale entità. In un primo momento, avevamo pensato di inserire al suo interno ogni singola variante del prodotto. Pensando alle lavatrici, avremmo avuto quindi 'lavatrice standard', 'lavatrice con cestello maggiorato'... Tuttavia, questa soluzione portava molte problematiche all'interno della progettazione del Database. Ad esempio, risultava complicato associare elementi della stessa classe di prodotto, cosa che rendeva molto scomoda la relazione con 'sequenza di operazioni'. Per tale motivo, abbiamo optato affinché in *Prodotto* vi siano le classi di prodotto (iPhone, Lavatrice Samsung X2F...) senza contemplare alcuna variante. Con data di produzione si intende l'immissione sul mercato di questo prodotto.

Variante

In questa entità troviamo, descritte con un campo testuale, tutte le possibili varianti ammissibili. Ogni *variante* può appartenere a più prodotti, così come un prodotto può avere, tra i possibili upgrade, più varianti.

Variante di prodotto

Questa entità ci permette di rappresentare un singolo modello di un prodotto contemplando anche le possibili varianti. Se in prodotto troviamo 'iPhone', qui avremo 'iPhone giallo 36GB', 'iPhone MAX 128GB' e così via. È scontato che esista un vincolo logico su tale entità: si devono avere delle combinazioni ammissibili di prodotto e variante. Il rischio è infatti quello di ottenere un 'iPhone con cestello maggiorato', che, per quanto invitante come regalo natalizio, risulta privo di logica. All'interno di questa entità si inseriscono anche i prodotti ricondizionati, che sono istanza di un prodotto e possono avere una serie di varianti (eventualmente aggiunte durante la fase di ricondizionamento). Dalle relazioni che legano la variante di prodotto si può risalire al costo di ogni unità, nel seguente modo:

$$PrezzoUnità = f * \left(PrezzoProdotto + \sum_{i=0}^n PrezzoVariante_i \right) \quad (1)$$

Dove f è un coefficiente che vale 1 nel caso in cui il prodotto sia nuovo, 0.8 nel caso in cui il prodotto sia ricondizionato.

Parte - Materiale

Parte è l'elemento costitutivo di ciascun prodotto. All'interno di *parte*, nella nostra progettazione dell'assemblaggio di un prodotto, rientrano anche le fasi intermedie del processo. Ad esempio, dovendo costruire un tablet, una volta messi insieme case esterno, processore e memoria RAM, avrò un elemento che mi permetterà, una volta unito con altri, di arrivare al prodotto definitivo. Anche questo elemento sarà considerato come una 'parte' di prodotto. Ogni *parte* fondamentale, ossia quelle che non si ottengono grazie a delle operazioni (contrassegnate tramite l'attributo intermedio), è composta da una serie di *materiali*. Il quantitativo di ogni materiale per ogni parte è espresso nella relazione '*composizione*' con l'attributo quantitativo.

Operazione – Giunzione – Utensile - Faccia

Operazione è il concetto attorno al quale ruota l'intera area. Ad essa sono necessariamente associate le parti che deve andare ad unire, così come la giunzione e gli utensili utilizzati. Rispetto alla giunzione, abbiamo scelto di usare un attributo multi-valore per esprimere tutte le possibili caratteristiche. In esso vi sono, oltre al nome, anche un valore numerico ed una sua unità di misura. Per esempio, volendo modellare una vite, potremmo avere tra le caratteristiche '*passo*', con 1.2 come valore e *mm* come unità di misura. Ogni operazione può essere eseguita su una singola faccia. Per tale motivo esiste un'entità faccia, che, oltre a permetterci di determinare quali siano le facce su cui può essere appoggiato un dispositivo (non ha importanza che abbiano una descrizione, basta un identificativo), fa sì che si possa associare ad ogni operazione l'unica faccia sulla quale essa può essere effettuata. In nome di ciò, il numero di facce di un prodotto risulta un attributo derivabile. In particolare, si può ottenere contando il numero di istanze della relazione '*posizione*' in cui compare un certo prodotto.

Sicuramente, un punto delicato all'interno del progetto della base di dati era come gestire i vincoli di precedenza tecnologica all'interno della produzione di un prodotto, vincoli che mi permettono anche di stabilire quali sono i possibili modi con il quale un prodotto può essere assemblato. Dopo un'attenta analisi, abbiamo scelto la seguente soluzione.

Ogni operazione ha una priorità, indicata con un numero interno da 1 a N. Affinché siano rispettati i vincoli di precedenza tecnologica, devono essere effettuate prima tutte le operazioni con priorità 1, poi tutte quelle con priorità 2 e così via. Sequenze di produzioni diverse si ottengono dalla possibilità di ordinare le operazioni all'interno delle singole priorità sulla base dell'indicatore di performance scelto.

Lotto

Se prodotto e variante di prodotto rappresentano, rispettivamente, le classi di prodotto e i vari modelli che sono realizzati dall'azienda, un lotto è una partita

di prodotti omogenei. Riferendoci ai lotti di prodotti nuovi, al loro interno troviamo prodotti che hanno tutti la stessa variante. In questo modo infatti, è possibile risalire alle unità disponibili per la vendita senza necessariamente avere il concetto di 'unità di prodotto'. Nel momento in cui un cliente vuole effettuare un acquisto di una merce, si cerca, tra i lotti disponibili di quel dato prodotto, se vi siano ancora elementi disponibili. Nonostante la costituzione del lotto sia omogenea sulla base della variante, la sequenza di operazione che si va ad associare prima della produzione dello stesso si riferisce al prodotto, in modo da non dover modificare la linea di assemblaggio per prodotti che, di fatto, sono identici, a meno di pochi dettagli. Una descrizione di 'Lotto Ricondizionato' si trova nella trattazione dell'area Smontaggio. Risulta evidente che, nell'ordinaria gestione dell'azienda, è necessario conoscere, per ciascuna variante di prodotto, quante unità sono attualmente disponibili per la vendita. Per rendere possibile questo, in prima istanza non è necessario un contatore su lotto che segni, per ciascuno, le unità disponibili. Infatti, basta contare le unità vendute per ogni variante di prodotto e confrontarle con il numero di unità complessivamente prodotte, valore ottenibile sommando i valori di 'Quantità' dei lotti realizzati.

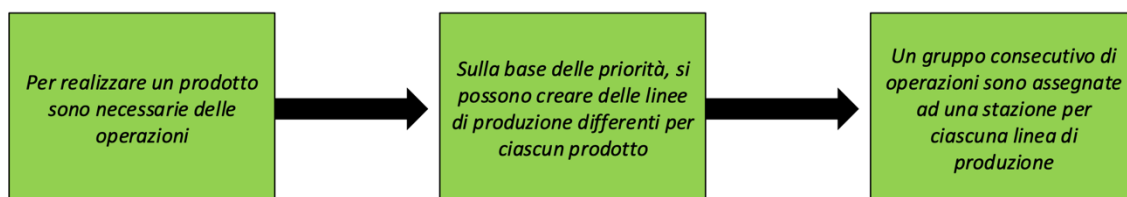
Sequenza di operazione – Sequenze possibili – Stazione

Con sequenza di operazione riusciamo a salvare nel Database le linee di produzione generate sulla base di un certo indice di performance per un certo prodotto. Si suppone che le sequenze di operazione possano essere generate indipendentemente dalla produzione di un lotto con tale sequenza. In particolare, risulta uno snodo cruciale la relazione *Fase*, che lega, per ciascuna sequenza, alcune delle sue operazioni alla stazione nella quale saranno effettuate. È banale il seguente vincolo: se O_1, \dots, O_m sono le m operazioni assegnate ad una stazione, queste devono essere effettuate sulla stessa faccia. Sapendo che ogni linea di produzione ha un tempo T entro il quale ciascuna stazione deve aver portato a termine tutte le operazioni, risulterebbe problematico associare tale valore alla singola stazione. Oltre ad un'evidente ridondanza (tutte le stazioni di una certa linea di produzione devono avere tale attributo), si va incontro ad anomalie in fase di modifica del valore in questione. Perciò, T è associato alla sequenza di operazioni. Quando non si riesce a rispettare questo tempo, l'unità viene persa e reimmessa nel processo produttivo alla fine (è il motivo per cui il termine della produzione di un lotto potrebbe essere diverso da quello previsto). Per tenere traccia di queste unità, si usa la relazione 'unità persa' tra stazione e lotto. Essa ha come attributo 'quante', in modo che, per ogni lotto di prodotti nuovi, si possa sapere in quali stazioni ho unità perse e quante (ovviamente, sapendo la stazione, si risale anche a tutte le operazioni effettuate prima della perdita). La relazione sequenza di operazione associa a ciascun prodotto tutte le sequenze relative.

In una revisione successiva dalla progettazione concettuale, ci siamo resi conto che, per quanto questa relazione risulti comoda per l'accesso immediato a tutte le sequenze di un certo prodotto (specialmente per quanto riguarda l'aggiunta di un lotto al database), essa è di fatto una ridondanza. Infatti, alle sequenze possibili è possibile accedere, a partire da ciascun prodotto, con il cammino *prodotto -> componente -> parti -> assemblaggio -> operazione -> fase -> stazione -> catena -> sequenze di operazioni*. A ragione di ciò, si noti che l'esistenza di una ridondanza ha come conseguenza la presenza di un ciclo dentro il diagramma. Piuttosto che discutere la relazione nell'ambito delle ridondanze, abbiamo preferito tenerla segnalando però il problema. In ogni caso, nel momento in cui andremo ad implementare i vincoli di integrità, terremo conto di questo aspetto. Raggiungere le sequenze di operazioni nel modo sopra esposto, oltre ad essere dispendioso, è anche innaturale nell'ambito dell'area di produzione: non ha senso accedere a tutte le parti di un prodotto, tutte le operazioni, tutte le fasi dell'assemblaggio per ricavare le sequenze (non è possibile passare da lotto perché, come detto sopra, possiamo generare una sequenza di operazioni prima che si generi un lotto con questa).

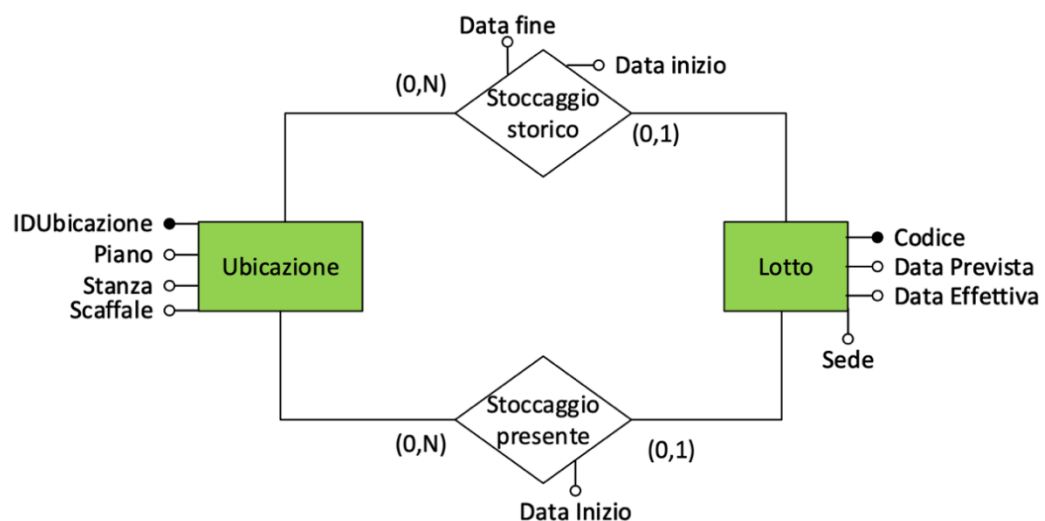
Operatore – Operazione Campione

All'interno della documentazione la figura dell'operatore risulta poco discussa; per questo ci siamo limitati a caratterizzarla nei termini più umani possibili, consapevoli che tale entità possa essere modificata nel tempo senza apportare alcuna conseguenza alla base di dati. Un operatore può essere assegnato a più stazioni, inteso come più postazioni di linee di produzioni di prodotti diversi. Per tenere traccia della 'capacità' lavorativa di ciascun operatore, abbiamo introdotto l'entità 'Operazione campione'. Questa comprenderà ad esempio 'saldare', 'svitare una vite'... Ogni operatore simulerà tali operazioni che si potrebbe trovare a fare durante il lavoro quotidiano nell'azienda, mentre quest'ultima ne misurerà tempo medio e varianza (quanto ci si discosta dal tempo medio). In questo modo, potremo avere una misura oggettiva delle abilità di ciascuno, per assegnare in maniera corretta ogni operatore a ciascuna stazione, per rendere più efficiente il processo produttivo.



Ubicazione – Magazzino

Ogni lotto si trova in un magazzino dell'azienda. In particolare, la relazione che permette di identificare un lotto in un certo magazzino è duplice: c'è 'Stoccaggio presente' e 'Stoccaggio storico'. La prima relazione è tale da identificare immediatamente i lotti disponibili in un dato magazzino, dai quali è ancora possibili prendere delle unità di prodotto da vendere. Una volta esaurito, si elimina il lotto da tale relazione e si inserisce nella seconda, che, come suggerisce il nome, permette di risalire a quali lotti sono stati presenti in un certo magazzino nel corso del tempo. Come si osserva dal diagramma ER, lotto non è però direttamente collegato con magazzino, ma con ubicazione. Immaginandoci un magazzino come un vastissimo capannone a più piani, potrebbe essere difficile dal punto di vista gestionale, risalire alla posizione di un singolo lotto. Per tale motivo, si ipotizza ogni magazzino come diviso in più piani, ciascuno dei quali è diviso in più stanze nelle quali vi sono diversi scaffali. In questo modo, si identifica univocamente e immediatamente la posizione di un lotto all'interno del magazzino (l'ispirazione è evidentemente i magazzini di **B**ea che si incontrano al termine del percorso). La predisposizione del magazzino indica la classe di prodotti che possono starvi, scelta tra 'leggeri' (smartphone), 'medi' (televisori) e 'pesanti' (lavatrici).



2. Area vendita

Entità	Attributi	Identificatore
Utente	Codice fiscale, Nome, Cognome, Indirizzo, Numero di telefono, Data iscrizione, Documento di identità (Tipo, Data scadenza, Ente, Numero),	Codice fiscale

	Indirizzo e-mail, Username, Hash, Salt, Credito	
Domanda di sicurezza	Codice, Testo	codice
Ordine	Codice ordine, Timestamp ordine, Indirizzo di consegna, DataIncasso	Codice ordine
Ordine in processazione		
Ordine in preparazione		
Ordine spedito		
Ordine evaso		
Ordine pendente		
Spedizione	Codice tracking, Data consegna prevista, Data consegna effettiva, Data Spedizione	Codice tracking, Data Spedizione
Spedizione spedito		
Spedizione in transito		
Spedizione in consegna		
Spedizione consegnata		
Hub	IDHub, Città	IDHub
Recensione	Codice, Campo testuale	Codice
Giudizio	IDGiudizio, Descrizione	IDGiudizio
Formula di Garanzia	Codice, Costo, Periodo	Codice
Formula di Garanzia temporale		
Formula di Garanzia a classe di guasto		
Unità acquistata	IDUnità	IDUnità
Richiesta di reso	IDRichiesta, Accettato	IDRichiesta
Reso	Codice, Ricondizionato	Codice
Motivazione	Codice, nome, descrizione	Codice
Diritto di reso		
Difetto		

Relazione	Attributi	Entità coinvolte
Sicurezza	Risposta	Utente, Sicurezza
Passaggio	Data, Ora	Spedizione, Hub
Caratterizzata	Voto	Recensione, Giudizio

Valutazione	Unità acquistata, Recensione
Catalogo	Utente, Ordine
Acquisto	Ordine, Unità acquistata
Consegna	Ordine, Spedizione
Motivo	Motivazione, Richiesta di reso
Restituzione	Unità acquistata, Richiesta di reso
Reso accettato	Reso, Richiesta di reso
Locazione	Reso, Magazzino
Estensione	Unità acquistata, Formula di garanzia
Applicabile	Formula di garanzia, Prodotto
Specifica	A classe di guasto, Guasto

Utente

Utente ha la principale finalità di memorizzare i dati relativi ad un utente della piattaforma online usata per la vendita dei prodotti elettronici; è dunque composta sia dagli attributi che modellano una persona, quali informazioni anagrafiche, numero di telefono ed indirizzo di abitazione, così come dai tipici dati di utente digitale, come username, indirizzo e-mail, nome utente e credito residuo. Abbiamo deciso, non essendoci indicazioni in tal senso, che gli acquisti possano essere fatti sul sito dell'azienda tramite un credito che ogni utente può gestire tramite carta di credito. Questo ci permetterà anche di fare controlli di consistenza in una fase successiva, come ad esempio stabilire se un utente ha abbastanza credito per effettuare un certo acquisto. La password dell'utente è memorizzata tramite coppia *hash + salt* per venire incontro alle normative vigenti in fatto di sicurezza dei dati (Paragrafo 83 GDPR).

Domanda di sicurezza

L'entità domanda di sicurezza definisce una domanda usata per verificare l'identità di un utente qualora si richieda un secondo livello di sicurezza o il rinnovo della password. Essa memorizza quindi il testo di alcune domande, decise dalla compagnia, tra le quali l'utente potrà scegliere al momento dell'iscrizione. Per memorizzare la risposta scelta dall'utente per la propria domanda di sicurezza si usa la relazione tra le due: il campo testuale al suo interno è volto proprio a tale scopo.

Ordine

Ordine si occupa di mantenere in memoria le informazioni circa un ordine eseguito da un cliente. Lo stato dell'ordine prevede 5 fasi, descritte all'interno del diagramma ER dalla relativa generalizzazione:

1. Mentre l'utente aggiunge elementi al carrello l'ordine è **in processazione**. Si ipotizza proprio che l'ordine sia inserito all'interno della base di dati dal momento in cui si inizia ad inserire elementi dentro il carrello digitale (ma, logicamente, eliminato se l'acquisto non va in porto. D'altronde, siti come Amazon permettono di mantenere un carrello 'attivo' anche alla fine di una sessione, in modo che eventualmente l'utente possa decidere successivamente di procedere con l'acquisto). Non si può procedere alle fasi successive se l'utente non ha abbastanza credito.
2. Quando l'utente ultima l'acquisto (il tasto 'procedi all' acquisto è presente in quasi tutti i siti di e-commerce) si può avere il passaggio allo stato **pendente** nel caso in cui non siano disponibili abbastanza elementi tra i lotti del magazzino. Sarà una procedura implementata nel database a sapere se un ordine debba passare in questo stato e, periodicamente, a verificare la disponibilità di ciascun prodotto per far proseguire l'iter dell'ordine.
3. Quando tutte le unità sono disponibili e il credito dell'utente è sufficiente, l'ordine passa allo stato **in preparazione**. Tale istante è memorizzato all'interno dell'attributo data incasso: in questo modo è possibile sapere quando è stato incassato l'ordine da parte dell'azienda (infatti tale momento non coincide con la creazione dell'ordine).
4. Una volta preparato il pacco contenente i prodotti dell'ordine e segnalato al corriere l'ordine passa in **evaso**.
5. Appena il corriere ritira il pacco in consegna l'ordine passa in **spedito**.

Spedizione

È l'entità che memorizza le informazioni sulla spedizione di un ordine come la data prevista di arrivo, la data effettiva di arrivo ed un codice di tracking. Una spedizione attraversa varie fasi durante il suo ciclo di vita:

- **Spedito** quando il *vettore* ritira il pacco dall'azienda;
- **In transito** quando il pacco viaggia tra i vari Hub;
- **In consegna** quando il vettore propone al destinatario un giorno di consegna ed esso accetta;
- **Consegnato** quando il destinatario riceve effettivamente il pacco e firma la ricevuta.

In questa sede, il codice di tracking, oltre ad identificare univocamente la spedizione, permette all'utente di risalire tramite il sito al percorso seguito dal

pacco, in modo da sapere lo stato, gli Hub dove è passato e il giorno previsto per la consegna. Tale funzionalità è equivalente all'opzione 'traccia il mio pacco' di molti siti di e-commerce.

Hub

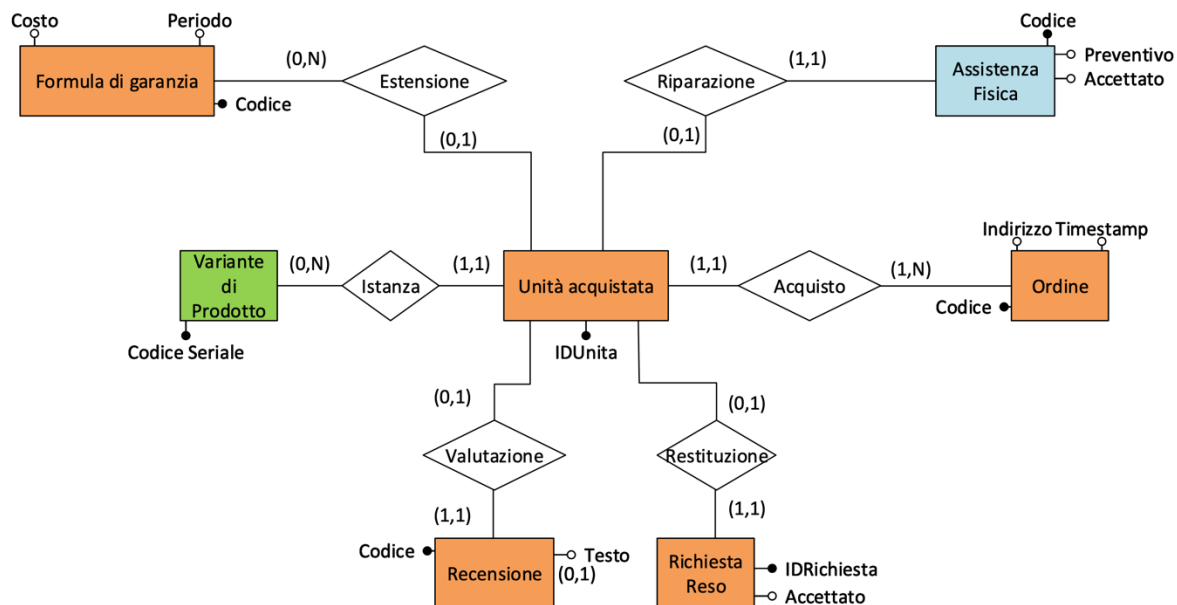
Un Hub è un centro di smistamento del vettore di spedizione. Il pacco mentre è in transito ne attraversa diversi, affinché il trasporto di un pacco possa essere più efficiente, in linea con le richieste sempre più stringenti di clienti in termini di tempo di attesa per l'arrivo dei pacchi. La relazione che lega Hub con Spedizione permette, ai fini del tracking sopra descritto, di risalire al momento del passaggio attraverso ciascun Hub per ogni spedizione.

Recensione - Giudizio

Nel momento in cui la spedizione relativa ad un certo ordine passa allo stato 'consegnata', il cliente ha la possibilità di inserire una recensione sul sito, una per ciascun'unità di prodotto acquistato (infatti, all'interno di un ordine, il cliente potrebbe essere soddisfatto di un certo prodotto, ma meno di un altro). Essa comprende un campo testuale e un voto su una serie di parametri stabiliti dall'azienda stessa, eventualmente facoltativi. Tali parametri sono contenuti nell'entità 'Giudizio'. Il voto, espresso tramite un intero da 1 a 5, è memorizzato come attributo della relazione 'caratterizzata'. Nel momento in cui si inserisce una recensione, si devono compilare tutti i parametri di giudizio.

Unità acquistata

Unità acquistata è il fulcro dell'area vendita (e, insieme a 'prodotto', una delle due entità attorno a cui orbita l'intera base di dati). Essa si utilizza per modellare la singola componente acquistata da un utente, ma, tramite le diverse relazioni ad essa collegate, ci permette di risalire all'ordine nel quale è stato aggiunto (*Acquisto*), alla specifica variante acquistata (*Istanza*), permette di collegare una recensione (*Valutazione*), una richiesta di reso (*Restituzione*), e così via.



La modellazione in questo modo è stato il frutto di un lungo lavoro di affinamento del diagramma. In un primo momento, avevamo realizzato un intero schema che permettesse di modellare, una per una, ogni unità realizzata dall'azienda. In questo modo, alla creazione di un lotto era associato la formazione di altrettante unità di prodotto, ciascuna di esse istanza della variante di prodotto legata al singolo lotto. In tale diagramma, l'unità di prodotto poteva essere acquistata o meno, e, di conseguenza, vi si legavano tutte le relazioni che sono adesso unite con 'Unità Acquistata'. Dopo una discussione, abbiamo trovato una serie di problematiche legate a questa implementazione: innanzitutto, vi era uno spreco di memoria all'interno del database per dei record che non avevano la necessità di esistere prima dell'acquisto dell'unità relativa (non si lavora mai con le unità in sé, ma solo con quelle che sono state acquistate); in secondo luogo, sarebbe stata necessaria l'implementazione di numerosi controlli per evitare anomalie: un'unità poteva avere un'estensione alla garanzia solo dopo essere stata acquistata, e allo stesso modo un voto, una riparazione, una restituzione... In ultimo, tutte le operazioni che avremmo potuto fare con quell'assetto del diagramma ER sono replicabili in questo: ad esempio, per sapere se è disponibile una unità di prodotto, basta andare a vedere se il numero di unità acquistate della variante di prodotto A è minore o alla somma di elementi prodotti con i lotti di A.

Formula di garanzia

È l'entità generale che permette di modellare una garanzia. All'acquisto, la garanzia è fissata per legge a 24 mesi. Essa può essere di due tipi: *temporale* o *a classe di guasto*, benché entrambe abbiano in comune un periodo di aumento della garanzia del prodotto acquistato. Un utente può decidere di

aggiungere una o più garanzie al singolo prodotto acquistato a patto che questa sia applicabile al prodotto in questione.

- Temporale: è la formula di base, aumenta il periodo di garanzia generale del prodotto di 36 o 48 mesi
- A classe di guasto: è una formula specifica per determinati guasti che coinvolgono il dispositivo.

Richiesta di reso

Se un utente acquista un prodotto di cui non è soddisfatto o che ha dei difetti evidenti riconducibili ad errori nel processo di produzione, ha la possibilità di richiedere una restituzione. Nel momento in cui invia la richiesta, essa deve essere accompagnata da una motivazione scelta tra quelle fornite dall'azienda. Se la richiesta viene accettata si procede al reso effettivo.

Reso

In base a come l'azienda risponde alla richiesta si può configurare o meno il reso. Esso è il primo passo verso il ricondizionamento di prodotti, fase che vedremo successivamente nella porzione di progettazione ad essa dedicata. Il reso è comunque stoccato in un magazzino in apposte aree adibite allo stoccaggio di unità rese, fintanto che non viene ricondizionato.

Motivazione

Le motivazioni per le quali si può richiedere una restituzione sono scelte dall'azienda, e le possiamo distinguere in due gruppi dal valore profondamente distinto:

- Diritto di recesso: dopo l'acquisto per legge si dispone di un periodo di 30 giorni entro il quale si può richiedere il reso del prodotto ed il completo rimborso dell'importo speso, senza altre motivazioni. L'azienda in tal caso è obbligata ad accogliere la richiesta, benché si debba verificare il diritto da parte dell'utente di richiedere tale procedura.
- Malfunzionamento: se dopo l'acquisto il prodotto smette di funzionare o assume comportamenti inconsueti può essere configurato un reso. In questo caso, nella richiesta va esplicitata la motivazione scegliendola tra quelle proposte (difetto estetico, difetto di funzionamento...). In tal sede, le motivazioni ritenute poco valide dall'azienda portano (in modo non automatico, ma sulla base di una scelta umana che distingua caso per caso) ad un rifiuto della richiesta.

3. Area assistenza

Entità	Attributi	Identificatore
--------	-----------	----------------

Guasto	Codice, Nome, Descrizione,	Codice
Codice di errore	Codice di errore	Codice di errore, Codice prodotto
Rimedio	Codice, Descrizione	Codice
Domanda	IDDomanda, Testo della domanda	IDDomanda
Sì domanda		
No domanda		
Sì rimedio		
No rimedio		
Assistenza fisica	Codice assistenza, Preventivo, Accettato	Codice assistenza
Intervento	Ticket, Data, Descrizione, FasciaOraria, Descrizione	Ticket
Intervento a casa	Latitudine, Longitudine	
Intervento presso CdA		
Tecnico	Nome, Cognome, Badge, Costo orario	Badge
Ordine parti	Codice, Data ordine, Data prevista consegna, Data effettiva consegna	Codice
Ricevuta fiscale	Codice, Metodo pagamento, Costo totale	Codice
Relazione	Attributi	Entità coinvolte
Riferisce		Codice d'errore, guasto
Fornisce		Codice d'errore, prodotto
Soluzione		Codice d'errore, rimedio
Diagnosi		Guasto, intervento
Riparazione		Assistenza fisica, unità acquistata
Prenotazione		Intervento, assistenza fisica
Fattura		Ricevuta fiscale, assistenza fisica
Ordinazione		Assistenza fisica, ordine parti
Effettua		Tecnico, intervento
Richiede		Tecnico, ordine parti
Ricambio		Ordine parti, parte

Assistenza virtuale	Prodotto, Guasto, Domanda
Next yes	Domanda, sì domanda
Next no	Domanda, no domanda
End yes	Rimedio, sì rimedio
End no	Rimedio, no rimedio

Guasto

Entità che raccoglie i possibili guasti che possono verificarsi nei vari prodotti. Ciascun di essi ha un nome ed una descrizione.

Codice di errore

Se, quando si verifica un guasto in un prodotto, esso viene riconosciuto, il display mostra un codice numerico grazie al quale è possibile risalire al guasto e ai possibili rimedi. Dato che uno stesso codice può essere condiviso da più prodotti per diversi guasti è necessario, per discernarli, che la chiave sia la composizione del codice e dello specifico prodotto.

Rimedio

Un rimedio è una serie di operazioni che possono essere eseguite per riparare il prodotto che presenta un guasto. Si compone di una descrizione testuale che elenca, appunto, le operazioni da compiere.

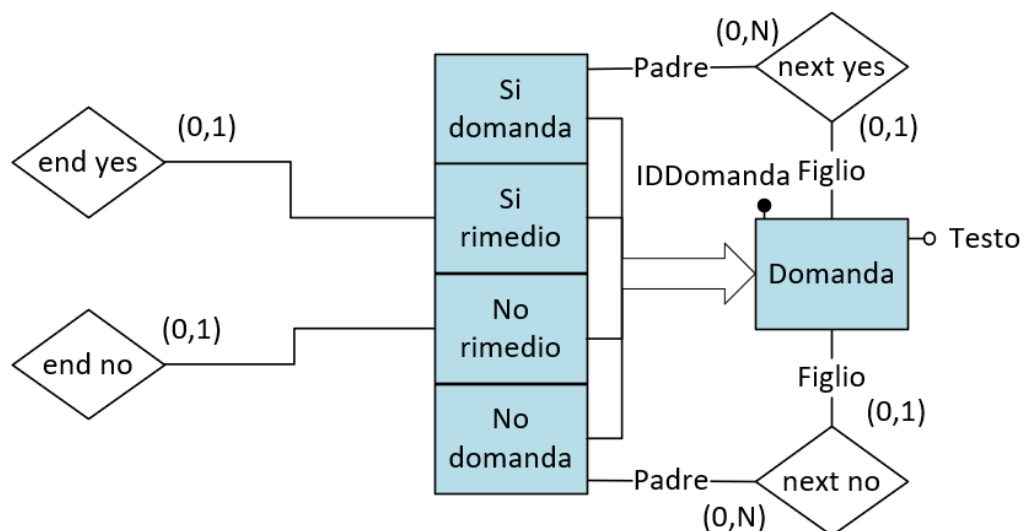
Domanda

L'entità domanda è il concetto attorno al quale ruota l'assistenza virtuale. Essa si compone di un campo testuale in cui si memorizza la domanda che sarà posta al cliente, il quale può rispondere solamente con un *sì* o con un *no*. Sulla base della risposta, si procede all'interno dell'albero. La nostra idea infatti consiste nel presupporre che, rispondendo *sì* (ma lo stesso vale per il *no*) ad una domanda, si possano avere tre alternative: c'è un intervento che potrebbe risolvere il problema; c'è un'ulteriore domanda; non vi sono altri collegamenti. Nel terzo caso, ragionevolmente, siamo arrivati alla fine del percorso, e sarà necessaria l'assistenza fisica. Per permettere tale memorizzazione, abbiamo optato per una generalizzazione parziale ma sovrapposta dal momento che ogni entità ha corrispondenza sia in uno degli insiemi relativi al "*sì*" che nell'insieme relativo al "*no*" e non sono compresi i casi di fine percorso (e, come abbiamo detto poc'anzi, ciascuna domanda non può avere due collegamenti con *sì* o con *no*, in quanto alla risposta il comportamento deve essere univoco):

- *Sì domanda*: se l'utente risponde affermativamente al testo ma la sequenza non porta direttamente ad un rimedio si procede alla prossima domanda.

- Sì rimedio: se l'utente risponde affermativamente al testo e la sequenza porta ad un rimedio diretto, allora viene proposto all'utente.
- No domanda: se l'utente risponde negativamente alla domanda ma non vi è un rimedio associato allora viene posta una nuova domanda. Tale relazione rappresenta la versione duale di "Sì domanda".
- No rimedio: può accadere che, ad una risposta negativa, si abbia comunque un rimedio. È il caso gestito da questa specializzazione.

Un'alternativa a tale scelta, che avrebbe sicuramente permesso una maggiore comprensione del meccanismo anche direttamente dall'ER, poteva essere quella di dire che, posta una domanda, un sì avrebbe portato sempre ad un rimedio, un no sempre ad una nuova domanda (o viceversa). In questo modo, avremmo avuto meno relazioni attorno all'entità domanda, così come meno complicazioni esplicative. Questa soluzione sarebbe però stata estremamente limitante. Si trattava infatti di porre delle domande per cui una risposta di un certo tipo doveva portare necessariamente ad un rimedio. Ma tale costrizione, oltre ad essere, appunto, restrittiva, non lasciava spazio ad alcuna verifica, poiché si basa sulla semantica della domanda, non su dato vero e proprio. In nome di ciò, abbiamo preferito lasciare la libertà di gestire l'assistenza virtuale e le domande come si preferisce, anche a costo di rendere (apparentemente) il tutto più criptico.



Assistenza fisica

È l'entità che raggruppa tutto ciò che riguarda il singolo caso di riparazione di un'unità acquistata. Se l'acquirente non è riuscito a riparare il prodotto tramite l'assistenza virtuale, l'iter di assistenza procede verso questa fase. Una richiesta di assistenza viene posta tramite il sito, e ciò porta alla creazione del primo intervento. All'interno di questa entità sono altresì memorizzati il preventivo e

se esso è stato accettato o meno. Dal momento in cui il preventivo viene accettato si può procedere alla riparazione del guasto.

Intervento

Un intervento può essere eseguito:

- a casa: presso il domicilio per prodotti ingombranti (elettrodomestici come lavatrici o frigoriferi)
- presso un centro di assistenza (CDA)

Un intervento a domicilio può comunque portare al trasporto del dispositivo presso uno dei centri di assistenza. Per modellizzare le varie possibilità, la scelta è stata quella di separare 'assistenza fisica' e 'intervento', in modo che, all'interno del singolo processo di riparazione di un oggetto, si potessero avere più interventi effettuati in maniera diversa. L'intervento ha un codice ticket che lo identifica univocamente e tiene conto della durata dell'intervento ai fini del costo totale della riparazione. È presente altresì una descrizione di quanto eseguito dal tecnico, in modo da mantenere traccia se eventuali altri tecnici dovessero occuparsi del caso, così come un collegamento, tramite la relazione 'Diagnosi', dei guasti riscontrati. Come specifica del testo, per l'organizzazione dei tecnici dell'azienda, sono previste delle fasce orarie giornaliere entro le quali il cliente può prenotarsi per un intervento. Con le prenotazioni, settimanalmente, a ciascun tecnico vengono assegnati gli interventi da eseguire ogni giorno per prossimità geografica. Per quanto riguarda le fasce orarie, abbiamo ipotizzato l'esistenza di quattro distinte: mattina1 (8:00-10:00), mattina2 (10:00-12:00), pomeriggio1 (15:00-17:00), pomeriggio2 (17:00-19:00). Un cliente può prenotarsi in una delle seguenti fasce orarie, nella misura in cui è disponibile almeno un tecnico libero. Per fare questo, sono memorizzate le prenotazioni per ciascuna settimana senza assegnare loro un tecnico immediatamente. Sulla base del loro numero, considerando quanti tecnici lavorano per l'azienda, si valuta se è possibile effettuare un intervento in tale fascia oraria:

Se ho due tecnici, e ho già due prenotazioni per la fascia mattina2 di Sabato prossimo, sarà impossibile che possa permettere un'ulteriore prenotazione per quel periodo. In nome di ciò, alla prenotazione del terzo cliente, Sabato – mattina2 non sarà mostrato.

In termini di efficienza dell'assegnamento dei tecnici, dopo un'attenta analisi, abbiamo optato per memorizzare le coordinate geografiche del luogo di intervento.

L'algoritmo di assegnamento degli interventi ai tecnici funziona come segue: si prendono tutti gli interventi e tutti i tecnici. Si pone inizialmente come ultima posizione di ogni tecnico latitudine 0 e longitudine 0 e si pone a 0 anche la

distanza totale percorsa. Si ordinano gli interventi di una giornata in base alla fascia oraria e si scorrono uno per uno. Per ogni intervento si calcola per tutti i tecnici che non hanno già un intervento in quella fascia oraria la distanza che coprirebbero andando ad eseguire questo intervento. Si assegna in fine al tecnico che minimizza questa distanza aggiornando ultima posizione, ultima fascia oraria e distanza totale. Contando M tecnici ed N interventi (con $N \leq 4 \cdot M$) la complessità di questo algoritmo risulta essere $O(N \cdot M)$. È possibile analizzare un PoC implementato in C++ di questo algoritmo nel file *dispatcher.cpp*

Tecnico

Un tecnico si occupa di eseguire una prima diagnosi e fornire un preventivo. Esso è inoltre l'unico che può creare un ordine per pezzi da sostituire. Per rendere più flessibile la gestione, da parte dell'azienda, degli interventi, ciascun tecnico ha un costo orario, in modo da distinguere tecnici specializzati da altri che lo sono meno. In questo modo, il costo del singolo intervento si ottiene moltiplicando il numero di ore necessarie per la manodopera del tecnico che lo ha effettuato

Ordine parti

Se per la riparazione è necessaria la sostituzione di qualche parte allora un tecnico effettua un ordine di pezzi. All'arrivo delle componenti si configura un nuovo intervento e si esegue la sostituzione. Una volta riparato si concorda con l'acquirente una data di ritiro del prodotto.

Ricevuta fiscale

Al ritiro del prodotto, o direttamente nel momento della riparazione effettuata presso il domicilio del cliente, viene rilasciata una ricevuta fiscale. Il costo complessivo si può ricavare a partire dal costo della manodopera, dalle ore necessarie per gli interventi e dal costo dei pezzi ordinati grazie alla seguente espressione:

$$ImportoRicevuta = \left(\sum_{j=0}^m CostoParteOrdinata_j + \sum_{i=0}^n CostoTecnico_i * OreIntervento_i \right)$$

4. Area smontaggio

Entità	Attributi	Identificatore
Test	Codice, Nome, Politica di sostituzione (PdS)	Codice

Lotto ricondizionato

Relazione	attributi	Entità coinvolte
Linked Test		Test, Test
Riguarda		Test, Parte
Fallimento	Sostituzione	Test, Reso
Test Tree		Test, Prodotto

Con l'area smontaggio, si introduce il concetto di prodotto ricondizionato. Per quanto sia già stato affrontato in alcuni punti (come nell'entità prodotto) il modo in cui si è scelto di descriverlo all'interno della progettazione, preferiamo qui riprenderlo per averne una trattazione chiara e completa.

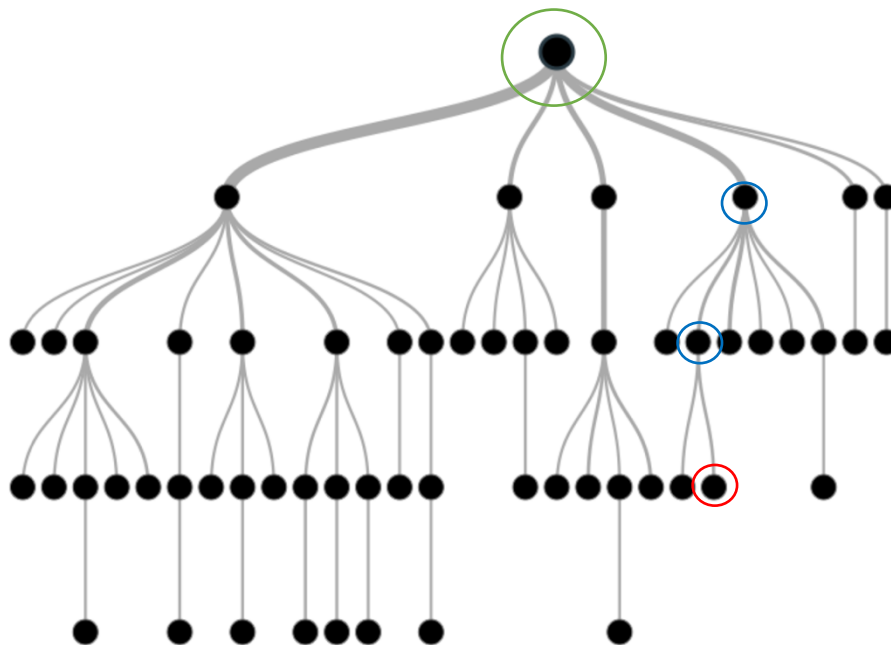
Una scelta molto discussa è stata quella sull'introdurre o meno il concetto di 'prodotto ricondizionato' in termini di entità. In un primo momento, avevamo optato per una generalizzazione su prodotto, che differenziasse in termini di prodotto nuovo e prodotto ricondizionato. Questo tuttavia portava a diverse complicazioni. Ad esempio, solo i prodotti nuovi hanno una linea di assemblaggio. Ciononostante, viene naturale associare l'entità linea di produzione con 'prodotto' piuttosto che con 'prodotto nuovo'. La soluzione al problema di dove andare a posizionare concettualmente i prodotti ricondizionati si è ottenuta in modo naturale introducendo l'entità 'variante di prodotto'. Un prodotto ricondizionato è a sua volta una variante di un prodotto, che può avere associati alcuni upgrade (vale ancora la relazione con variante) e il cui prezzo viene ribassato nei termini espressi dalla formula (1).

Lotto ricondizionato

Un lotto ricondizionato non nasce a seguito di un processo di assemblaggio come accade per i lotti nuovi (non a caso, la relazione 'assemblaggio' è legata con lotto nuovo, non con lotto), ma per un processo di ricondizionamento di una serie di elementi resi. Si noti che non vi è alcuna relazione tra reso e lotto ricondizionato. Questo perché le singole unità, una volta ricondizionate, hanno un nuovo codice seriale (che esprime il loro essere ricondizionate) e possono essere modificate con delle varianti, quindi, stabilito a priori che il processo che porta i prodotti ad avere una nuova vita sia impeccabile, non c'è necessità di conoscere quali prodotti avessimo in precedenza. Si può risalire al prodotto di un lotto tramite la relazione 'produzione', così come avviene con i prodotti nuovi, con il vincolo per cui, se un lotto è ricondizionato, allora può essere costituito solo da un prodotto ricondizionato (come per lotto, anche questi sono omogenei sulla base del prodotto). Il lotto 451 conterrà quindi solamente iPhone Gialli Ricondizionati, mentre il lotto 472 Lavatrici Samsung 4L con cestello maggiorato. Questo ci consente di risalire, a partire dai lotti presenti, a quanti e quali prodotti siano disponibili per l'acquisto.

Test

La gestione dell'albero dei test, per come richiesta dal progetto, si implementa all'interno della base di dati tramite l'entità *Test* e la relazione ricorsiva *Linked Test*. Si osservino le due etichette sugli archi che collegano i due elementi del diagramma ER. In termini di strutture dati, quello che dobbiamo implementare è un albero generico, tale per cui ogni nodo A può avere un numero indefinito di figli $A_1 \dots A_k$. SQL si presta ad una memorizzazione di questo tipo anche senza dover ricorrere a strutture dati 'non isomorfe' con quella che è la realtà. Infatti, la relazione linked test associa ad ogni nodo tutti i possibili figli. Il risultato è proprio un albero N-ario.



Grazie a tale scelta, una volta arrivati ad un certo test, è possibile andare a quelli successivi esplorando la relazione *Linked Test*. In particolare, se in questa relazione non vi sono record che hanno quel test come padre, allora siamo giunti ad una foglia, e si effettua una sostituzione della parte / delle parti in gioco. Queste sono associate ad ogni test tramite la relazione 'riguarda'. La relazione 'TestTree' è il collegamento tra un prodotto (indipendentemente dalla variante, si ha sempre lo stesso albero) e la root, o meglio, il primo test che viene eseguito (quello che, nella figura sopra, è segnato in verde). Il testo poi richiede che, se una percentuale di sotto-test fallisce, sia modificata l'intera parte. Questo può essere fatto tramite il dato (opzionale) 'politica di sostituzione', che esprime tale indice. Se una percentuale di sotto-test rispetto

a quella in 'politica' fallisce, allora tutte le parti che si riferiscono al dato test sono sostituite.

Per quanto non sia necessario risalire a quale prodotto diventi cosa, può essere importante, anche per fini statistici, sapere quali sono le parti che più vengono sostituite, oppure dove si hanno più fallimenti di test, in vista di un possibile ripensamento del processo produttivo. È possibile, a partire da ciascun reso, sapere quali test falliscono tramite la relazione con test 'Fallimento'. In particolare, in esso vi è un attributo (non ridondante) che esprime se le parti relative a quel test sono state sostituite o meno. Facciamo un esempio, con riferimento alla figura.

Si immagini che, per il reso che ha identificativo 3120 si abbiano tutti i test positivi fino a quello cerchiato in blu al primo livello dell' albero (test 7). Allora, per come avviene il processo di ricondizionamento, si effettuano tutti i sotto test figli di quello fallito. Se fallisce nel test blu al secondo livello (test 10), si passa a quelli sottostanti. Arrivati al test fallito cerchiato in rosso (test 15), essendo una foglia, si effettua anche una sostituzione della relativa parte. Rispetto a questa situazione, la relazione, in forma di tabella, potrebbe avere il seguente aspetto.

Reso	Test	Sostituzione
	...	
3120	7	0
3120	10	0
3120	15	1
	...	

4. Ristrutturazione del diagramma ER

In questa fase ci siamo occupati della ristrutturazione del diagramma, in modo da andare a togliere tutte quelle componenti che, seppur utili durante la progettazione concettuale per visualizzare al meglio alcune caratteristiche della realtà da modellizzare, non sono direttamente traducibili nello schema logico.

1. Rimozione delle generalizzazioni

La quasi totalità delle generalizzazioni presenti nel database, più che per rappresentare delle forme di 'ereditarietà' (in termini di linguaggi di programmazione) dei concetti, ci sono serviti per avere, subito di primo impatto, alcune caratteristiche delle entità nell'ER. Con riferimento all'entità 'ordine', è evidente che, nella nostra

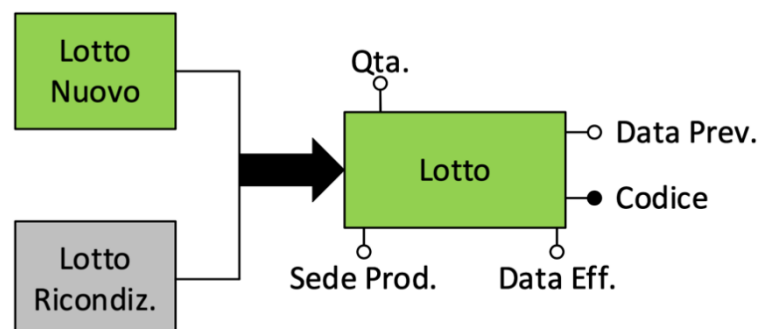
progettazione semantica del database, lo stato sia un elemento fondamentale, su cui andremo a basare anche molte delle nostre operazioni. In nome di questo, abbiamo reputato utile poter visualizzare immediatamente tutti i possibili stati dell'ordine, consapevoli che, per evitare anomalie di sorta, in ogni caso sarebbero confluiti in un attributo del tipo 'stato'.

Entità Ordine

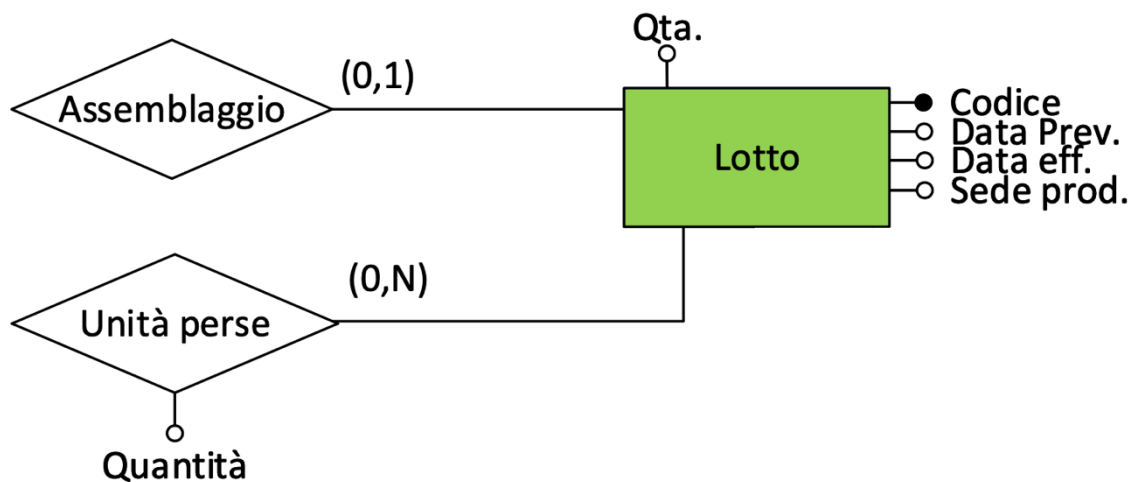
Inizialmente abbiamo rappresentato gli stati dell'ordine con una generalizzazione totale esclusiva. Per ristrutturare questa entità abbiamo optato per un attributo di **stato**, in quanto ogni specializzazione non risulta avere attributi o relazioni esclusive.

Entità Lotto

Un lotto può essere prodotto, e quindi avere le relazioni con la sequenza di produzione e con unità perse, oppure ricondizionato, senza le caratteristiche di una filiera di produzione. La generalizzazione è la seguente:



Un approccio naturale a questo caso sarebbe quello di inserire un attributo di stato, che permetta di dire se il lotto in questione è ricondizionato o meno. Tuttavia, ciò sarebbe una ridondanza, e porterebbe delle problematiche in fase di normalizzazione. Infatti, l'essere ricondizionato o meno è direttamente associato alla variante di prodotto che il lotto istanza, e non ha senso che la medesima informazione stia anche in lotto. Per tale motivo, scegliamo di non introdurre alcun attributo, ma di modificare solamente le cardinalità delle relazioni associate (la linea di assemblaggio è una caratteristica propria dei lotti di prodotti nuovi).



Entità Spedizione

Una spedizione può trovarsi in vari stati in base al momento che si sta considerando: *spedito*, *in transito*, *in consegna* o *consegnato*. Dato che nessuna specializzazione ha attributi o relazioni esclusive, si opta per una traduzione tramite attributo di tipo.

Entità Motivazione

Una motivazione può appartenere a due motivi generici: *diritto di recesso* o *difetto*. Essendo una differenziazione logica più che contenutistica si decide di trattarla con un attributo.

Entità Intervento

Un intervento può avvenire presso il domicilio del cliente o presso un centro di assistenza. Anche in questo caso, abbiamo introdotto un attributo di tipo per distinguere i due casi.

Entità Formula di Garanzia

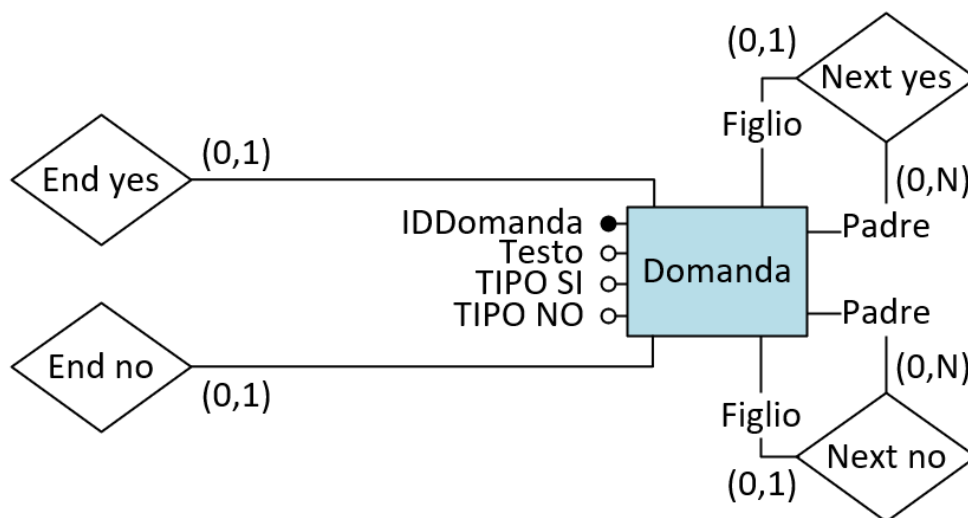
Una formula di garanzia può essere di carattere *temporale* oppure può riferirsi a determinate classi di guasto; in questo secondo caso ha logicamente delle relazioni con le relative classi di guasti. Si decide di tradurla con un attributo di tipo, modificando la cardinalità con l'entità 'guasto'.

Entità Domanda

Una domanda costituisce un nodo dell'albero dell'assistenza virtuale: a ciascuna domanda si può rispondere affermativamente o negativamente e, per qualsiasi risposta, deve essere possibile proseguire nell'albero passando alla prossima domanda o ad un relativo rimedio (nel caso in cui non siamo ad una foglia), così come già esposto nel paragrafo precedente. La generalizzazione

in questione è totale e sovrapposta, in quanto ogni entità appartiene sicuramente sia ad una specializzazione del sì che ad una del no.

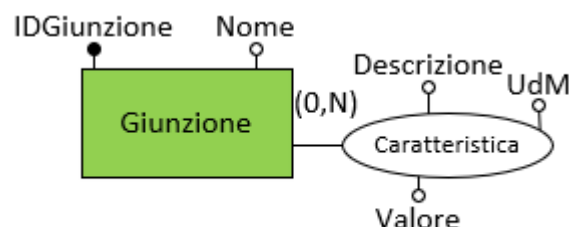
Per non creare tutte le possibili combinazioni, durante la fase di ristrutturazione, la scelta è stata quella di tradurla come se le generalizzazioni fossero due, una che coinvolge la risposta negativa, mentre l'altra che coinvolge la risposta positiva. Abbiamo quindi inserito nella ristrutturazione due attributi di tipo (necessari per sapere, in caso di risposta positiva/affermativa, se il passo successivo nell'albero sia una domanda o un intervento, così come per mantenere la consistenza). Le cardinalità delle relazioni si adeguano di conseguenza.



2. Conversione degli attributi composti e multi-valore

Entità Giunzione ed attributo composto e multi-valore caratteristica

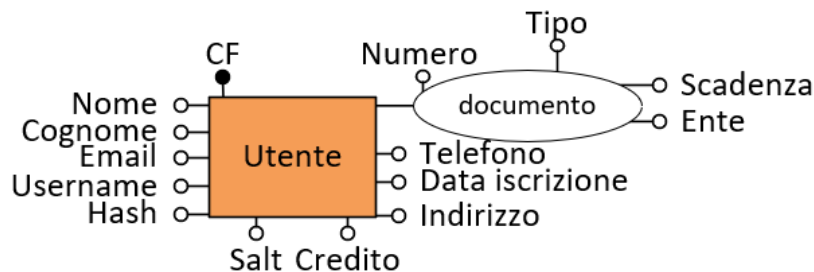
Una giunzione può avere varie caratteristiche. In una prima sede, abbiamo creato un attributo composto e multi-valore:



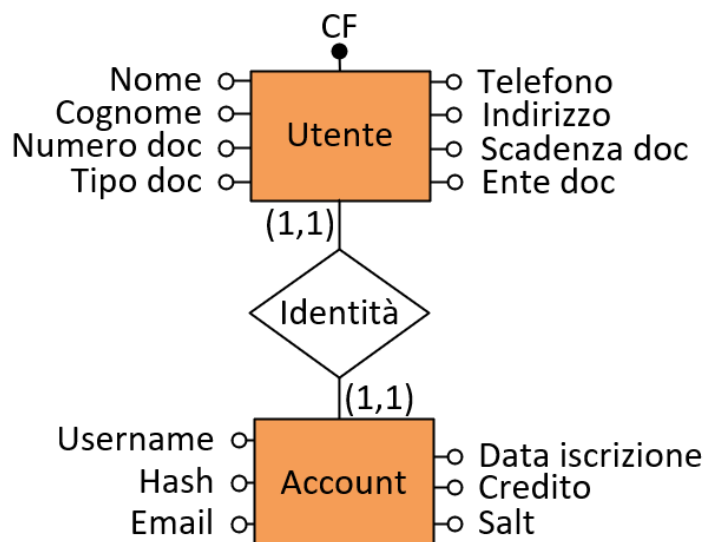
Nella traduzione, abbiamo tradotto l'attributo in una nuova entità con una relazione uno a molti per avere maggiore flessibilità negli inserimenti. La chiave scelta per questa nuova entità è IDCaratteristica.

Entità utente ed attributo composto documento:

Un utente contiene le informazioni che permettono il login al servizio, le informazioni relative alla persona e tra questi tutte le informazioni relative ad un documento. Nella progettazione concettuale abbiamo inserito tutto all'interno di una singola entità:



In fase di ristrutturazione ci siamo tuttavia accorti che, inserendo tutti gli attributi di *documento* nell'entità, avremo ottenuto un oggetto inutilmente grande. Basti pensare alla fase di login al servizio: non serve prendere tutte le informazioni relative ad un utente, ma solo username, e-mail e password. Si procede dunque scorporando l'entità in due più piccole, divise logicamente in una porzione "online" ed "offline":



Entità intervento ed attributo composto posizione

Un intervento può doversi tenere presso un centro di assistenza o presso il domicilio del cliente: in quest'ultimo caso, è necessario memorizzarne la posizione geografica. Abbiamo optato per un semplicissimo attributo composto da latitudine e longitudine, che ci permetta di eseguire operazioni in modo più formale rispetto ad altre scelte più approssimative. Durante la traduzione abbiamo eliminato l'attributo composto e spostato i due attributi che lo componevano direttamente nella entità.

5. Individuazione delle operazioni e analisi delle prestazioni

1. Tavola dei volumi

Concetto	Tipo	Motivazione	Volume
Prodotto	Entità	Ipotesi	50
Aggiunta	Relazione	Da Variante, relazione (1,1)	100
Variante	Entità	Ipotesi: ogni prodotto ha in media 2 possibili varianti	$50 \times 2 = 100$
Variazione	Relazione	Da Variante di Prodotto, relazione (1,1)	150
Variante di prodotto	Entità	Si ipotizza la produzione di 3 modelli diversi per ciascun prodotto	$50 \times 3 = 150$
Modello	Relazione	Ogni variante di prodotto usa in media 1.5 varianti	$1.5 \times 150 = 225$
Parte	Entità	Ipotesi: ogni prodotto ha in media 100 parti + le parti intermedie	$(100 + (100-1)) \times 50 = 9.950$
Componente	Relazione	Da Parte relazione (1,1)	9.950
Materiale	Entità	Ipotesi: numero di materiali utilizzati	10
Composizione		Ipotesi: ogni parte è composta in media da due materiali (si contano solo le parti base)	$100 \times 2 = 200$
Giunzione	Entità	Ipotesi: numero di giunzioni disponibili	100
Caratteristica	Entità	Ipotesi: ogni Giunzione ha in media 1.5 caratteristiche	$100 \times 1.5 = 150$
Proprietà	Relazione	Da Caratteristica, relazione (1,1)	150
Unione	Relazione	Da Operazione, relazione (1,1)	1450
Operazione	Entità	<i>Vedere nota 1</i>	$(100-1) \times 50 = 4.950$
Assemblaggio	Relazione	Da Operazione, relazione (1,1)	4.950
Utensile	Entità	Ipotesi	20

Uso	Relazione	In media, in ogni operazione si usano 1.5 utensili	7.425
Faccia	Entità	Ipotesi: Ogni prodotto ha in media 4 facce (in genere o 2 o 6)	$4 \times 50 = 200$
Posizione	Relazione	Da Faccia, relazione (1,1)	200
Appoggia	Entità	Da Operazione, relazione (1,1)	4.950
Sequenza di operazioni	Entità	Ipotesi: ogni prodotto ha due possibili sequenze di produzione	$2 \times 50 = 100$
Sequenze possibili	Relazione	Da Sequenza di Operazioni, relazione (1,1)	100
Lotto	Entità	Ipotesi: ogni variante di prodotto ha in media 50 lotti	$50 \times 150 = 7.500$
Produzione	Relazione	Da Lotto, relazione (1,1)	7.500
Assemblaggio	Relazione	Ipotesi: 4/5 dei lotti sono prodotti e hanno una relativa sequenza	6.000
Stazione	Entità	Ipotesi: media di 30 stazioni per sequenza	$30 \times 100 = 3.000$
Catena	Relazione	Da Stazione, relazione (1,1)	3.000
Fase	Relazione	99 Operazioni per prodotto * 100 sequenze possibili	9.900
Operatore	Entità	Ipotesi	300
Lavora	Relazione	Da Stazione, relazione (1,1)	3.000
Operazione Campione	Entità	Ipotesi	5
Capacità	Relazione	Ogni operatore si sottopone a tutti i test	$300 \times 5 = 1.500$
Unità perse	Relazione	Ipotesi: ciascun lotto prodotto in media perde 5 unità	$5 \times 6000 = 30.000$
Ubicazione	Entità	Ipotesi: ogni magazzino ha 150 ubicazioni	$5 \times 150 = 750$
Magazzino	Entità	Ipotesi	5

Compone	Relazione	Da Stazione, relazione (1,1)	750
Stoccaggio attuale	Relazione	Ipotesi: media di lotti all'attivo	600
Stoccaggio Storico	Relazione	Differenza tra lotti e stoccaggio attuale	$7.500 - 600 = 6.900$

1. Ogni operazione unisce due parti di un prodotto, generando quindi una terza parte intermedia. Supponendo di avere inizialmente 100 parti, dopo una prima operazione avrò 99 parti ancora da assemblare, di cui una intermedia e 98 base. Continuando in questo modo, indipendente dall'ordine scelto, dopo la seconda operazioni avrò 98 parti, dopo la terza 97 e così via. In definitiva, quando arriverò ad avere il prodotto concluso (e quindi, quando avrò terminato la produzione) avrò compiuto 99 operazioni. Generalizzando:

Per assemblare un prodotto di N parti, ho bisogno di N-1 operazioni.

Concetto	Tipo	Motivazione	Volume
Unità acquistata	Entità	Ipotesi: 20 prodotti per lotto di media, venduti almeno tutti quelli di stoccaggio storico	$20 * 6.900 = 138.000$
Istanza	Relazione	Da Unità Acquistata, relazione (1,1)	138.000
Recensione	Entità	Ogni due unità acquistate c'è una recensione	$138.000 / 2 = 69.000$
Valutazione	Relazione	Da recensione, relazione (1,1)	69.000
Giudizio	Entità	Ipotesi	5
Caratterizzata	Relazione	Ogni recensione risponde a ciascun giudizio	$5 * 69.000 = 345.000$
Richiesta di reso	Entità	Ipotesi: per il 7% delle unità vendute è richiesto il reso	$138.000 * 0,07 = 9660$
Restituzione	Relazione	Da Richiesta di Reso, relazione (1,1)	9.660

Reso	Entità	Ipotesi: $\frac{3}{4}$ delle richieste sono accettate	$9.660 \cdot 0.75 = 7.245$
Reso Accettato	Relazione	Da Reso, relazione (1,1)	7.245
Locazione	Relazione	Si considerano solo i resi non ancora ricondizionati. Ipotesi: $\frac{2}{5}$	2.890
Motivazione	Entità	Ipotesi	20
Motivo	Relazione	Da Richiesta di Reso, relazione (1,1)	9.660
Utente	Entità	Ipotesi	46.000
Identità	Relazione	Da Identità, relazione (1,1)	46.000
Account	Entità	Uno per ciascun utente	46.000
Sicurezza	Relazione	Da Account, relazione (1,1)	46.000
Domanda di sicurezza	Entità	Ipotesi	20
Ordine	Entità	Ogni utente effettua due ordini	92.000
Acquisto	Relazione	Da Unità Acquistata, relazione (1,1)	138.000
Catalogo	Relazione	Da Ordine, relazione (1,1)	92.000
Spedizione	Entità	Ipotesi	87.000
Consegna	Relazione	Da Spedizione, relazione (1,1)	87.000
Hub	Relazione	Ipotesi	7.000
Passaggio	Relazione	Ipotesi: ogni spedizione in	$4 \cdot 87.000 = 348.000$

		media passa per 4 hub	
Formula di garanzia	Entità	Ipotesi: 2 temporali + 10 classi di guasto	12
Specifica	Relazione	Ipotesi: ogni classe di guasto ha in media 4 guasti	$4 \times 10 = 40$
Applicabile	Relazione	Ipotesi: ogni prodotto può avere le 2 estensioni temporali + 2 specifiche	4
Estensione	Relazione	$\frac{1}{4}$ di unità acquistate hanno una garanzia, $\frac{1}{4}$ ne hanno due	$138.000 \times 0.25 + 138.000 \times 0.25 \times 2 = 103.500$

Entità	Tipo	Motivazione	Volume
Guasto	Entità	Ipotesi	200
Codice di errore	Entità	Ogni prodotto ha in media 7 codici di errore	$7 \times 50 = 350$
Riferisce	Relazione	Da Codice di Errore, relazione (1,1)	350
Fornisce	Relazione	Da Codice di Errore, relazione (1,1)	350
Rimedio	Entità	Ipotesi	420
Soluzione	Relazione	Ipotesi: ogni codice di errore per ciascun prodotto ha 3 rimedi	$3 \times 350 = 1.050$
Assistenza virtuale	Entità	Ipotesi: ogni prodotto ha in media tre guasti per i quali c'è un'assistenza virtuale	$50 \times 3 = 150$
Domanda	Entità	Ogni assistenza virtuale ha in media 5 domande associate	$5 \times 150 = 750$

Next Yes	Relazione	Tutte le domande che non sono root hanno un padre	$750-150=600$
Next No	Relazione	Tutte le domande che non sono root hanno un padre	$750-150=600$
End Yes	Relazione	Vedi nota 2	$750 * 0,5 - 750 * 0,2 = 225$
End No	Relazione	Vedi nota 2	$750 * 0,5 - 750 * 0,2 = 225$
Assistenza fisica	Entità	In media, il 12% delle unità acquistate necessita dell'assistenza fisica	$138.000 * 0.12 = 16.560$
Riparazione	Relazione	Da Assistenza Fisica, relazione (1,1)	16.560
Intervento	Entità	Ipotesi: in media, due interventi per assistenza (sopralluogo e ritiro)	$16.560 * 2 = 33.120$
Prenotazione	Relazione	Da intervento, relazione (1,1)	33.120
Diagnosi	Relazione	Ipotesi: in ogni intervento si ha in media la diagnosi di un guasto	$33.120 * 1 = 33.120$
Effettua	Relazione	Da intervento, relazione (1,1)	33.120
Tecnico	Entità	Ipotesi	10
Ordine Parti	Entità	Ipotesi: in media, il 65% delle assistenze fisiche richiede un ordine	$16.560 * 0,65 = 10.764$
Ricambio	Relazione	Ipotesi: per ogni ordine, si acquistano tre parti	$10.764 * 3 = 32.292$
Richiede	Relazione	Da Ordine Parti, relazione (1,1)	10.764
Ordinazione	Relazione	Da Ordine parti, relazione (1,1)	10.764
Ricevuta fiscale	Entità	Ipotesi: in media, sono accettate l'80% delle assistenze fisiche	$16.560 * 0,80 = 13.248$
Fattura	Relazione	Da Fattura, relazione (1,1)	13.248

2. Pur avendo già esposto dettagliatamente il funzionamento della rappresentazione dell'albero per l'assistenza virtuale, riteniamo corretto esplicitare il motivo di tale volume, che rappresenta comunque un dato approssimativo. Supponiamo di essere su una

domanda che ha come TIPO SI 'Domanda' e come TIPO NO 'Rimedio'. Questo significa che tale domanda avrà, a patto che non sia una foglia, una relazione con End No (che porta al rimedio associato). Se i tipi fossero stati invertiti l'End opposto. Ciò significa che, presa una domanda, ho in media il 50% di probabilità che questa abbia un collegamento con End Yes, e altrettanto con End No. A tale valore aggiungiamo una stima delle domande che non hanno un rimedio associato, il 20% di tutte le domande. I collegamenti in End Yes e End No sono quindi $750 * 0,5 - 750 * 0,2 = 225$.

Concetto	Tipo	Motivazione	Volume
Test	Entità	Ipotesi: un prodotto ha in media 20 test	$20 * 50 = 1.000$
TestRoot	Relazione	Da Prodotto, relazione (1,1)	50
Linked Test	Relazione	Ipotesi: ogni test, tranne il padre, ha un padre	$1.000 - 50 = 950$
Riguarda	Relazione	Ipotesi: ogni test riguarda in media 2 parti	$1.000 * 2 = 2000$
Fallimento	Relazione	Ipotesi: l'1% dei test su ciascun reso fallisce	$20 * 7.245 * 0.01 = 1.449$

2. Operazioni scelte

Si propongono le seguenti operazioni da effettuare all'interno del database. La scelta di ciascuna è stata effettuata partendo sia da quelle che possono essere le quotidiane richieste di accesso a determinati dati, sia da quanto ciascuna di esse risulti rilevante sul carico applicativo ordinario. Senza un ordine specifico nell'elenco, sono:

1. Verifica disponibilità delle unità per un ordine
2. Incasso complessivo dell'azienda in un intervallo temporale
3. Ricerca rimedi a partire da un codice di errore ed un prodotto
4. Inserimento di un lotto di prodotti e gestione stoccaggio
5. Tracking di una spedizione
6. Gestione termine dell'assistenza fisica
7. Calcolo dei periodi di guasto relativi ad una unità acquistata
8. Visualizzazione del giudizio complessivo di un prodotto

3. Analisi costi delle operazioni e introduzione di ridondanze

Verifica disponibilità delle unità per un ordine

Descrizione: Nel momento in cui il cliente sceglie i prodotti da inserire nell'ordine e, dalla piattaforma online, richiede l'acquisto, è necessaria una verifica della presenza di tali unità all'interno dei magazzini dell'azienda. Lo stesso deve accadere quando, periodicamente, si verifica la disponibilità di elementi per gli ordini pendenti. Questa funzionalità, alla richiesta del passaggio di un ordine allo stato 'in processazione' si occupa di verificare tale presenza.

Input: Codice Ordine

Output: Segnale di riuscita o meno dell'operazione

Riuscita

Frequenza giornaliera: 65¹

1. Anche nelle successive operazioni che riguardano gli ordini giornalieri, si ipotizza che quotidianamente passino dallo stato 'in processazione' allo stato 'in preparazione' 50 ordini. A questi, vanno sommati gli ordini di cui si verifica la disponibilità ma che, non potendo essere realizzati per assenza di materiale, passano allo stato 'pendente': questi saranno per ipotesi 5 al giorno. Infine, si ipotizza che ogni giorno si debba verificare la presenza di unità per 10 lotti che si trovavano in uno stato 'pendente'.

Porzione diagramma Interessato:

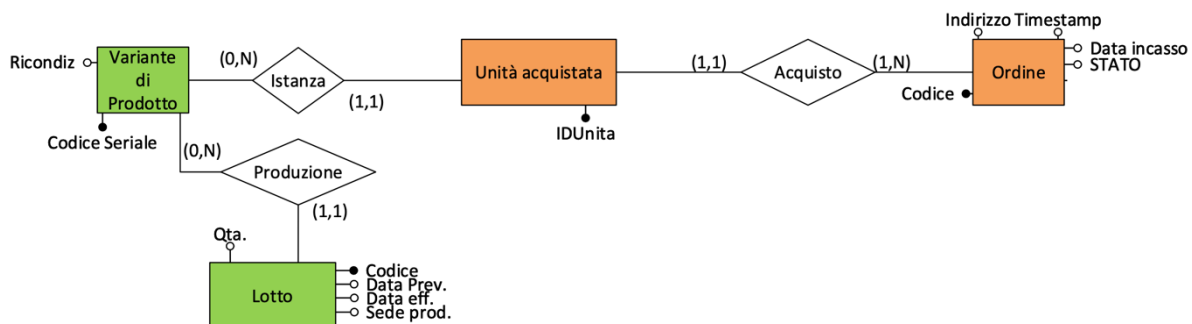


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Acquisto	Relazione	138.000
Unità Acquistata	Entità	138.000
Istanza	Relazione	138.000
Variante di prodotto	Entità	150
Produzione	Relazione	7.500
Lotto	Entità	7.500

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Acquisto	R	2	L	Conoscendo il codice dell'ordine, si va a prendere le unità ad esso associate.
Unità Acquistate	E	2	L	Lettura dei codici delle unità acquistate
Istanza	R	$2+1.840^2$	L	Si cerca a che modello si riferisce ciascuna unità acquistata e si cercano le altre unità acquistate
Variante di prodotto	E	2	L	Si leggono le unità acquistate
Produzione	R	100^3	L	Si risale ai lotti prodotti per ciascun modello
Lotto	E	100	L	Si controlla il numero di unità prodotte per il dato modello sommando i dati in 'quantità' di ciascun lotto
Costo in termini di operazioni elementari				2.048

Costo giornaliero: 133.120

- Non si deve solo cercare a quale modello si riferiscono le singole unità acquistate, ma anche verificare quante unità acquistate di quel relativo modello sono state vendute. Infatti, all'interno di lotto non abbiamo disponibili il numero di elementi presenti in magazzino per ciascun modello, ma è necessario ricavarli facendo la differenza tra il numero di quantità prodotte per ciascun modello (informazione contenuta in lotto) e quante unità del dato modello sono state vendute. Considerando che il numero di modelli è 150 e le unità vendute sono 138.000 si può stimare che per ciascun modello in media siano state vendute $138.000/150 = 920$ unità. Il valore è moltiplicato per due in quanto tale ricerca va eseguita per ciascuna delle due unità dell'ordine.
- Poiché i lotti prodotti sono 7.500, si può stimare che per ciascun modello siano prodotte $7.500/150=50$ lotti.

Ridondanze: L'utilizzo di un attributo per ciascun lotto che tenga conto di quante unità disponibili sono presenti all'interno dello stesso può aiutare per la verifica della disponibilità di elementi.

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Acquisto	R	2	L	Conoscendo il codice dell'ordine, si va a prendere le unità ad esso associate.
Unità Acquistate	E	2	L	Lettura dei codici delle unità acquistate
Istanza	R	2	L	Si cerca a che modello si riferisce ciascuna unità acquistata e si cercano le altre unità acquistate
Variante di prodotto	E	2	L	Si leggono le unità acquistate
Produzione	R	100	L	Si risale ai lotti prodotti per ciascun modello
Lotto	E	100	L	Si controlla il numero di unità prodotte per il dato modello sommando i dati in 'quantità' di ciascun lotto
Costo in termini di operazioni elementari				208
Costo giornaliero:				13.520

Il numero di operazioni elementari risparmiate è

$$\Delta_{\text{read}} = n - n_{\text{RID}}^T = 133.120 - 13.520 = 119.600$$

Bisogna dunque aggiornare questa ridondanza: ogni qualvolta che un ordine passa allo stato 'in preparazione', e quindi viene effettivamente acquistato, si va a numericamente modificare il valore delle unità rimaste all'interno di un lotto, cioè la disponibilità. Nella realizzazione, è necessario osservare la business rule dell'azienda onorando la quale si dà priorità di evasione ai lotti prodotti da più tempo. Per fare questo, basta prendere le singole unità dal lotto che, avendo ancora elementi disponibili, è stato prodotto da più tempo.

La frequenza giornaliera di questo update è $g^A=50$, e corrisponde al numero di volte in cui ogni giorno un ordine passa allo stato 'in preparazione'.

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Acquisto	R	2	L	Conoscendo il codice dell'ordine, si va a prendere le unità ad esso associate.
Unità Acquistate	E	2	L	Lettura dei codici delle unità acquistate
Istanza	R	2	L	Si cerca a che modello si riferisce ciascuna unità acquistata e si cercano le altre unità acquistate
Variante di prodotto	E	2	L	Si leggono le unità acquistate
Produzione	R	100	L	Si risale ai lotti prodotti per ciascun modello
Lotto	E	2	S	Si cerca il lotto con unità disponibili del dato modello e si aggiorna il valore del contatore 'rimanenti'.
Costo in termini di operazioni elementari				110
Costo giornaliero:				5.500

$$n_{rid}^T + o^A = 13.520 + 5.500 = 19.020$$

Il costo dell'operazione con la ridondanza è (decisamente) inferiore rispetto a $n^A = 133.120$, quindi manteniamo la ridondanza introducendo in lotto l'attributo 'rimanenti'.

Incasso complessivo dell'azienda in un intervallo temporale

Descrizione: A partire da due date, si ricava l'incasso dell'azienda in quell'intervallo di tempo. Con tale operazione è possibile richiedere l'incasso giornaliero (si inserisce in entrambi i campi la CURRENT_DATE), l'incasso settimanale, trimestrale...

Il dato in questione si può ottenere considerando gli ordini effettuati nella data specificata e il costo complessivo delle unità al loro interno secondo l'espressione (1) del paragrafo 3.1. A questi vanno aggiunte le formula di garanzia relative ai prodotti acquistati e le ricevute fiscali emesse a seguito della conclusione di interventi di assistenza fisica (la ricevuta fiscale si ottiene a partire dall'ultimo intervento effettuato per un dato processo di assistenza, che coincide con l'emissione della ricevuta fiscale). Per calcolare il costo di questa operazione è necessario dare una stima di quale sia l'intervallo medio di tempo richiesto da ogni operazione. Infatti, modificando le due

date, dovrò accedere a più elementi all'interno della base di dati, e quindi il costo sarà maggiore. Possiamo ipotizzare che la funzionalità sia richiesta quotidianamente su un intervallo di un giorno, settimanalmente su un intervallo di 7 giorni, e così via a seconda delle esigenze di bilancio dell'azienda. Una media verosimile è che tale operazione sia richiesta quotidianamente su un intervallo di due giorni.

Input: PrimaData, SecondaData

Output: Incasso

Prima Data Seconda Data Incasso

Frequenza giornaliera: 1 richieste giornaliera

Porzione diagramma Interessato:

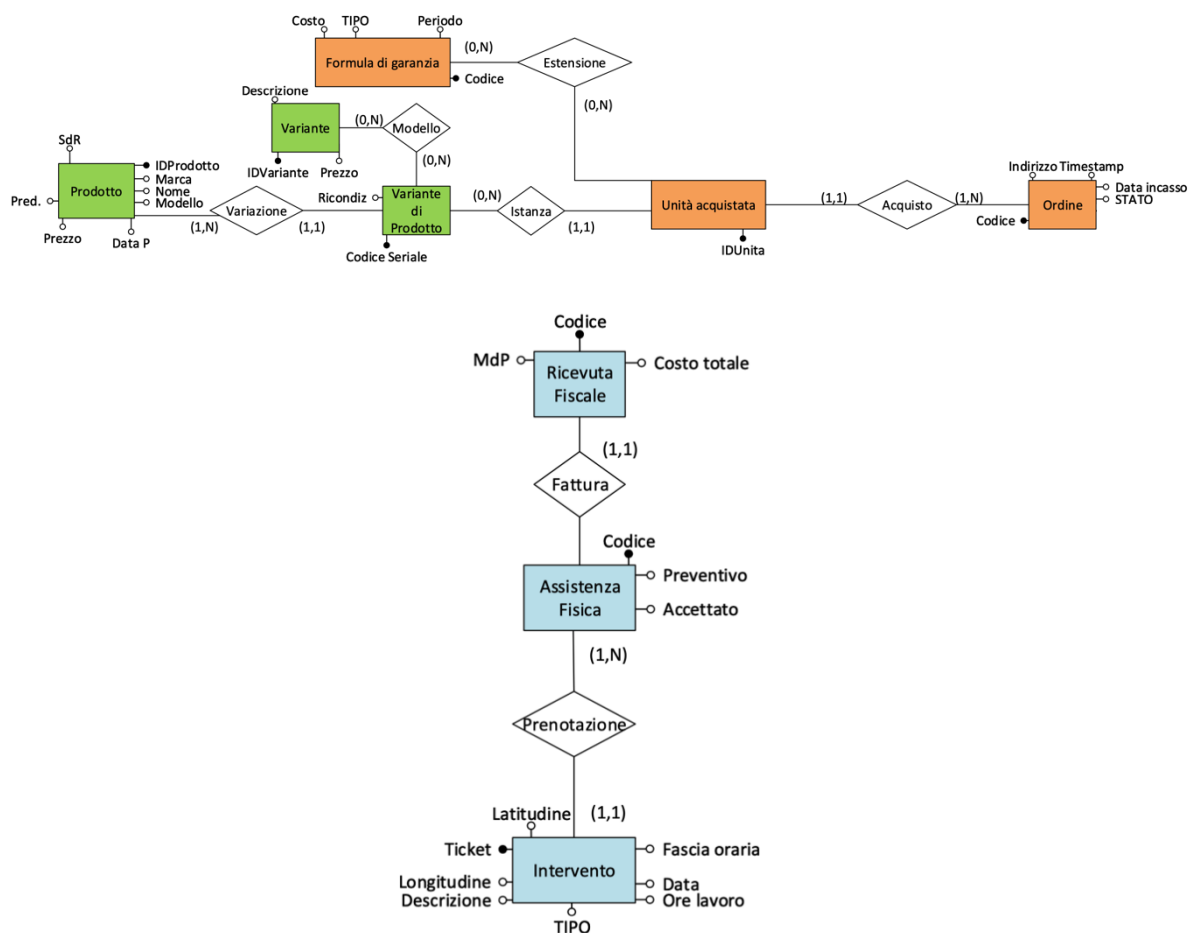


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Ordine	Entità	92.000
Acquisto	Relazione	138.000

Unità Acquistata	Entità	138.000
Istanza	Relazione	138.000
Variante di prodotto	Entità	150
Variazione	Relazione	150
Prodotto	Entità	50
Modello	Relazione	225
Variante	Entità	150
Estensione	Relazione	103.500
Formula di garanzia	Entità	12
Intervento	Entità	33.120
Prenotazione	Relazione	33.120
Assistenza Fisica	Entità	16.560
Fattura	Relazione	13.248
Ricevuta Fiscale	Entità	13.248

Tavola degli accessi:

Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazioni	Motivazione
Ordine	E	100 ¹	L	Si prendono gli ordini incassati in quel periodo
Acquisto	R	200	L	Si considerano le unità per ciascun ordine
Unità Acquistata	E	200	L	Come sopra
Istanza	R	200	L	Si risale al modello di ciascuna unità
Variante di prodotto	E	200	L	Come sopra
Variazione	R	200	L	Si cercano i modelli da cui provengono le singole varianti
Prodotto	E	200	L	Si legge il prezzo di ciascun prodotto
Modello	R	300 ²	L	Si leggono le varianti di ciascun modello
Variante	E	300	L	Si legge il prezzo di ciascuna variante

Estensione	R	150 ³	L	Si cercano le formule di estensione aggiunte agli ordini
Formula di garanzia	E	150 ⁴	L	Si leggono i prezzi delle formule di estensione
Intervento	E	24	L	Si cercano gli interventi effettuati in tale data.
Prenotazione	R	24	L	Si risale alle rispettive assistenze fisiche
Assistenza Fisica	E	24	L	Si verifica che l'intervento in tale data sia l'ultimo (emissione della ricevuta)
Fattura	R	12 ⁵	L	Si accede alla ricevuta
Ricevuta Fiscale	E	12	L	Lettura dell'importo complessivo
Costo in termini di operazioni elementari:				2.296
Costo Giornaliero				2.296

1. La data che si utilizza è quella in cui l'ordine passa da uno stato 'in processazione' o 'pendente' a 'in preparazione', e quindi *DataIncasso*.
2. Come specificato all'interno della tavola dei volumi, si suppone che in media il numero di varianti per ciascun modello sia 1.5.
3. Con riferimento alla tavola dei volumi, ogni 4 prodotti acquistati si comprano 3 formule di garanzia
4. Non è curioso che il numero di letture su garanzia sia molto più grande del numero di effettive formule di estensione della garanzia. Infatti, per ciascuna formula acquistata è necessario effettuare nuovamente la lettura.
5. Se, come riportato nella tavola dei volumi, in media ogni assistenza fisica richiede due interventi per la sua conclusione, allora ogni due interventi letti uno porta all'emissione della ricevuta

Ridondanze:

Come ridondanze per tale relazione abbiamo pensato di introdurre un attributo su ordine che tenga conto dell'importo complessivo.

Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazioni	Motivazione
Ordine	E	100	L	Si prendono gli ordini incassati in quel periodo

Intervento	E	24	L	Si cercano gli interventi effettuati in tale data.
Prenotazione	R	24	L	Si risale alle rispettive assistenze fisiche
Assistenza Fisica	E	24	L	Si verifica che l'intervento in tale data sia l'ultimo (emissione della ricevuta)
Fattura	R	12	L	Si accede alla ricevuta
Ricevuta Fiscale	E	12	L	Lettura dell'importo complessivo
Costo in termini di operazioni elementari:				196
Costo Giornaliero				196

Il numero di operazioni elementari risparmiate è

$$\Delta_{\text{read}} = n - n_{\text{RID}}^T = 2.296 - 196 = 2.073$$

Per quanto riguarda la consistenza della ridondanza, si dovranno fare delle operazioni aggiuntive all'incasso di un nuovo ordine. Ricordando che la frequenza giornaliera di passaggio di un ordine allo stato *in preparazione* è $g^A=50$, il costo complessivo dell'inserimento di un ordine si aggrava della ricerca del costo delle singole componenti:

Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazioni	Motivazione
Ordine	E	1	L	Si prendono gli ordini incassati in tale data
Acquisto	R	2	L	Si considerano le unità per ciascun ordine
Unità Acquistata	E	2	L	Come sopra
Istanza	R	2	L	Si risale al modello di ciascuna unità
Variante di prodotto	E	2	L	Come sopra
Variazione	R	2	L	Si cercano i modelli da cui provengono le singole varianti
Prodotto	E	2	L	Si legge il prezzo di ciascun prodotto

Modello	R	3	L	Si leggono le varianti di ciascun modello
Variante	E	3	L	Si legge il prezzo di ciascuna variante
Estensione	R	2	L	Si cercano le formule di estensione aggiunte agli ordini
Formula di garanzia	E	2	L	Si leggono i prezzi delle formule di estensione
Costo in termini di operazioni elementari:				24
Costo Giornaliero				1.200

Il costo giornaliero dell'operazione con la ridondanza sommato al costo per la consistenza dell'attributo è

$$n_{\text{rid}}^T + o^A = 1.200 + 196 = 1.396$$

Che è minore di $n^A = 2.296$, il costo dell'operazione senza la ridondanza. Si sceglie quindi di mantenere la ridondanza.

Durante questa fase ci siamo resi conto che, andando a considerare il costo complessivo dell'ordine come un attributo ridondante, si aumenta la flessibilità del database: infatti, ci possiamo permettere di modificare il costo delle varianti, dei prodotti o delle formule di garanzia nel tempo senza avere dei problemi nel risalire alla spesa per gli ordini precedenti a tale modifica.

Ricerca rimedi a partire da un codice di errore

Descrizione: Dato un codice di errore e un prodotto, è possibile risalire tramite questa funzione a tutti i rimedi possibili per tale problema.

Input: Codice prodotto, Codice di errore

Output: Rimedi

Rimedi

Frequenza giornaliera: 150 richieste giornaliere

Porzione diagramma Interessato:

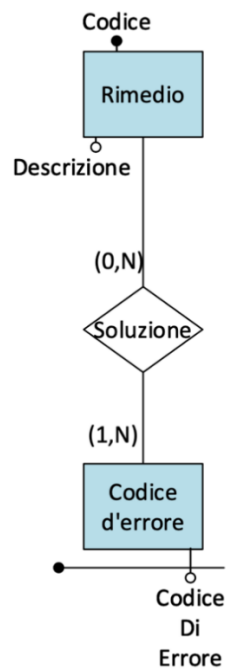


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Soluzione	Relazione	420
Rimedio	Entità	1.050

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Soluzione	R	3 ¹	L	Tramite il prodotto e il codice ricerchiamo i rimedi associati.
Rimedio	E	3	L	Lettura dei rimedi associati.
Costo in termini di operazioni elementari				6
Costo giornaliero:				900

1. Come specificato nella tavola dei volumi, si suppone che ogni codice prodotto abbia in media tre rimedi associati.

Ridondanze: non si ritiene necessario l'introduzione di ridondanze.

Inserimento di un lotto di prodotti e gestione stoccaggio

Descrizione: Nell'ambito della produzione, una delle operazioni più significative di inserimento è quella che riguarda un nuovo lotto. Infatti, tutto quello che riguarda gli altri ambiti (prodotto, parti, operazioni...) risulta statico successivamente all'inserimento nella base di dati, mentre i lotti sono realizzati con regolarità. L'operazione in questione si interessa all'inserimento di un lotto di prodotti (sia nuovo che ricondizionato) all'interno della base di dati. Se il lotto è nuovo, si deve specificare anche una sequenza di produzione e una data prevista di produzione (le unità perse dello specifico lotto saranno aggiornate durante la produzione vera e propria del lotto, così come la data di produzione effettiva sarà inserita al termine del processo). Se il lotto è ricondizionato, la data prevista e la data effettiva di produzione si riferiscono al tempo che si impiega durante il processo di ricondizionamento ad eseguire tutti i test. In ogni caso, si deve verificare che, nella produzione di un lotto del modello di prodotto X, un lotto ricondizionato sia associato necessariamente ad un prodotto ricondizionato (e uguale con i lotti di prodotti nuovi). Durante l'inserimento, viene specificato anche un magazzino di preferenza per quanto riguarda lo stoccaggio del lotto. Nel caso in cui in tale magazzino non vi siano ubicazioni libere, allora si cerca un'altra qualsiasi ubicazione tra tutti i magazzini. Il caso limite è quello per cui non vi sia modo di stoccare il lotto: in tal caso, l'operazione solleva un errore.

Alla funzionalità di inserimento di un lotto si affianca un trigger che, nel momento in cui il lotto non ha più elementi rimanenti, viene spostato da stoccaggio attuale a stoccaggio storico, liberando l'ubicazione. Sebbene questo non rientri completamente nella produzione del lotto, ne riteniamo necessaria una discussione in quanto permette un mantenimento in stato consistente dell'intera area produzione. Teniamo presente che tale funzionalità sfrutta in modo pesante la ridondanza introdotta per l'operazione *Verifica disponibilità delle unità per un ordine*.

Input: Modello di produzione, Quantità da produrre, Data Prevista di produzione, Sede produzione, Sequenza di operazione da utilizzare, Magazzino.

Output: Segnale di buon fine dell'operazione di inserimento

Successo

Frequenza giornaliera: 5

Porzione diagramma Interessato:

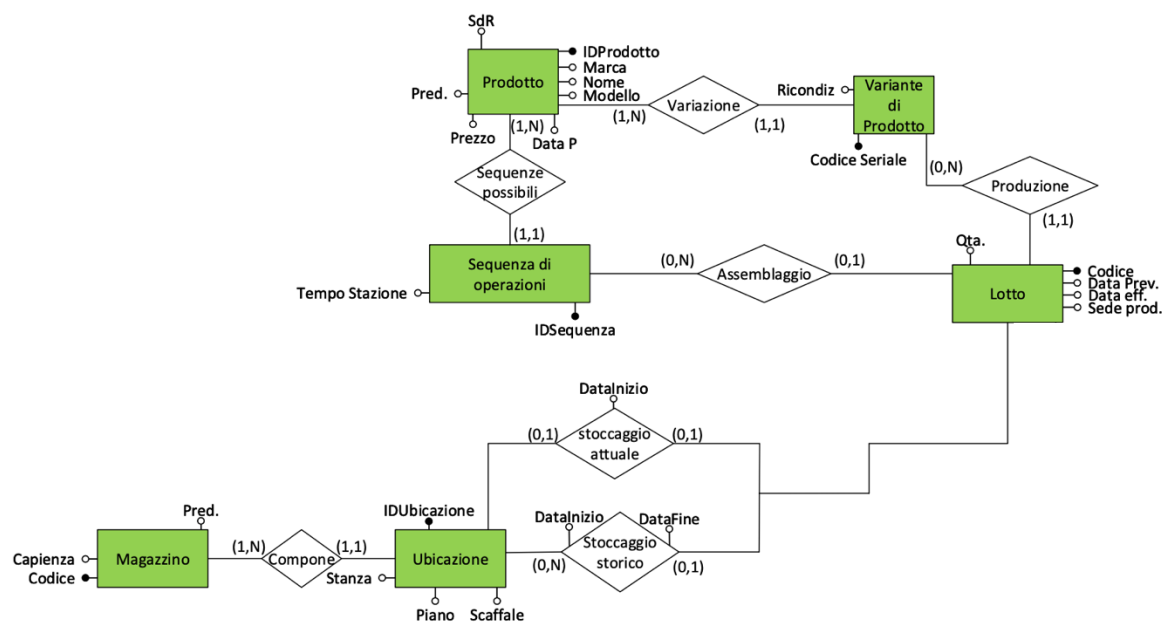


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Variante di prodotto	E	150
Variazione	R	150
Prodotto	E	50
Sequenze Possibili	R	100
Sequenza di operazioni	E	100
Produzione	R	7.500
Lotto	E	7.500
Assemblaggio	R	6.000
Stoccaggio Attuale	R	600
Ubicazione	E	750
Compos	R	750
Magazzino	E	5

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Variante di prodotto	E	1 ¹	L	Accedo alla relativa variante di prodotto

Variazione	R	1	L	Risalgo al prodotto
Prodotto	E	1	L	Come sopra
Sequenze possibili	R	1	L	Accedo alle possibili sequenze per il dato prodotto
Sequenza di operazioni	E	1	L	Verifico che la sequenza scelta sia valida
Magazzino	E	1	L	Verifico l'esistenza del magazzino scelto
Compone	R	150 ²	L	Accedo alle varie ubicazioni
Ubicazione	E	150	L	Come sopra
Stoccaggio attuale	R	150 ³	L	Si verifica la possibilità di inserire il lotto nell'ubicazione.
Lotto	E	1	S	Inserisco il lotto
Assemblaggio	R	1	S	Se non è ricondizionato, inserisco la sequenza di operazioni scelta
Stoccaggio Attuale	R	1	S	Inserisco il lotto nell'ubicazione trovata.
Costo in termini di operazioni elementari:			462	
Costo giornaliero:			2.310	

1. Tutte le operazioni di lettura che precedono riguardano il controllo di consistenza dell'input con i dati presenti nel database.
2. Nel caso peggiore, è necessario verificare tutte le ubicazioni all'interno dei magazzini per risalire a quelle disponibili. In media, si può stimare che si debbano solamente controllare le ubicazioni di due magazzini. Le ubicazioni per magazzino sono in media $750/5=150$.
3. Tale verifica conclude i controlli di consistenza.

Ridondanze: Per questa operazione, non si ritiene necessario l'inserimento di ridondanze

Trigger per la gestione delle relazioni di stoccaggio:

Ipotizziamo che ogni giorno siano 5 i lotti per cui non si hanno prodotti rimanenti. Infatti, in media ogni lotto ha all'incirca 20 prodotti (138.000 unità acquistate / 7.500 lotti) e le unità acquistate ogni giorno sono 100 (50 ordini al giorno * 2 unità al giorno).

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Stoccaggio attuale	R	1	S	Si elimina il lotto dalla relazione 'stoccaggio attuale' con ubicazione.
Stoccaggio Storico	R	1	S	Si inserisce il lotto nella relazione 'stoccaggio storico' con la medesima ubicazione in cui si trovava.
Costo in termini di operazioni elementari:			4	
Costo giornaliero:			20	

Tracking di una spedizione

Descrizione: Una volta che l'ordine passa allo stato spedito, si crea una relativa istanza in spedizione, la cui chiave è il codice tracking. Il cliente può utilizzare tale codice per sapere in ogni momento il percorso seguito dal suo pacco in termini di Hub in cui ha sostato e la data prevista di arrivo.

Input: Codice tracking

Output: Percorso tra gli Hub della spedizione:

IdHub	Città	DataPassaggio	OraPassaggio
-------	-------	---------------	--------------

Frequenza giornaliera: 100 richieste giornaliere

Porzione diagramma Interessato:

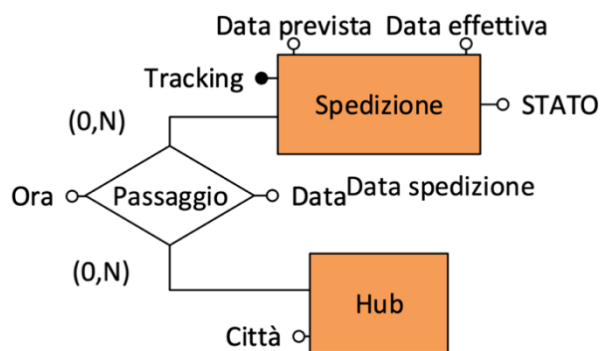


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Spedizione	Entità	87.000
Passaggio	Relazione	87.000
Hub	Entità	7.000

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Spedizione	E	1	L	Controllo della spedizione
Passaggio	R	4 ¹	L	Informazioni sul passaggio tra gli Hub
Hub	E	4	L	Informazioni sugli Hub
Costo in termini di operazioni elementari:			9	
Costo giornaliero:			900	

1. Come specificato nella tavola dei volumi, si suppone che ogni spedizione passi in media in 4 Hub diversi prima di arrivare a destinazione.

Ridondanze: non si ritiene necessario l'introduzione di ridondanze.

Calcolo dei periodi di guasto relativi ad una unità acquistata

Descrizione: Quando si effettua una riparazione, se l'unità riparata risulta essere sotto garanzia temporale o sotto una garanzia relativa al guasto, allora la serie di interventi risulta gratuita. La garanzia è inoltre prolungata all'atto della riparazione. È necessario

dunque accedere sia alle garanzie acquistate ed i guasti coperti da queste polizze che alle eventuali riparazioni di pezzi, non dimenticando il periodo standard di durata della garanzia.

Input: Codice unità acquistata

Output: Guasti e relative date di scadenza:

IDUnità	0	Nome unità	DataScadenza
IDGuasto	1	Nome Guasto	DataScadenza
CodiceParte	2	Nome Parte	DataScadenza

Frequenza giornaliera: 10 richieste giornaliere

Porzione diagramma Interessato:

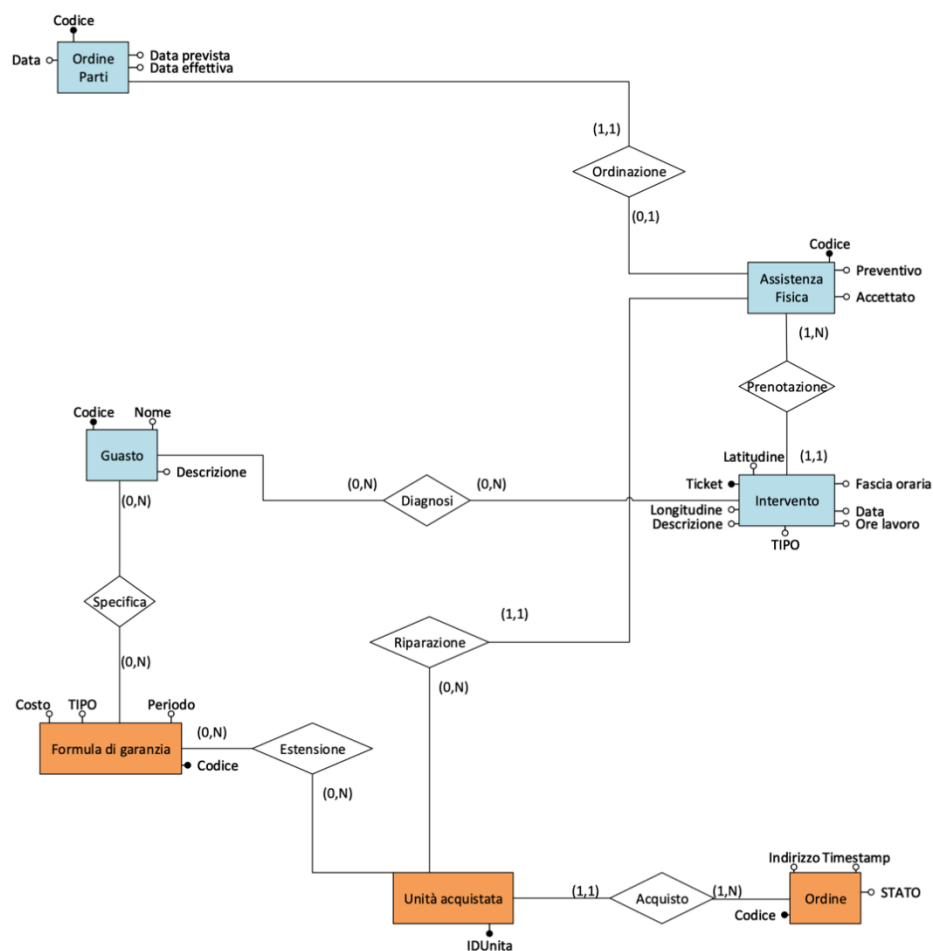


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Unità acquistata	Entità	138.000
Acquisto	Relazione	138.000
Ordine	Entità	92000
Estensione	Relazione	103.500
Formula di garanzia	Entità	12
Specifica	Relazione	40
Guasto	Entità	200
Riparazione	Relazione	16.560
Assistenza Fisica	Entità	16.560
Prenotazione	Relazione	33.120
Intervento	Entità	33.120
Diagnosi	Relazione	33.120
Ordinazione	Relazione	10.764
Ordine parti	Entità	10.764
Ricambio	Relazione	32.292
Parte	Entità	2.950

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Unità acquistata	Entità	1	L	Prendo le informazioni relative all'unità acquistata in questione
Acquisto	Relazione	1	L	Prendo la relazione con l'ordine
Ordine	Entità	1	L	Prendo la data dell'ordine per calcolare le garanzie
Estensione	Relazione	2	L	Prendo le tuple di relazione con formule di garanzia

Formula di Garanzia	Entità	2	L	Prendo le effettive garanzie applicate sulla unità in oggetto
Specifica	Relazione	8	L	Prendo le tuple di guasto associate alle singole tipologie di garanzia
Guasto	Entità	8	L	Prendo i guasti effettivi su cui vigono le varie garanzie dell'unità
Riparazione	Relazione	1	L	Prendo le riparazioni dell'unità
Assistenza Fisica	Entità	1	L	Prendo i dati relativi alle singole riparazioni
Prenotazione	Relazione	2	L	Prendo le tuple di relazione con interventi
Intervento	Entità	2	L	Prendo gli interventi effettivamente partecipati in questo processo di riparazione
Diagnosi	Relazione	1	L	Accedo alle tuple di relazione con diagnosi
Guasto	Entità	1	L	Prendo il guasto relativo a questa riparazione
Ordinazione	Relazione	1	L	Prendo la tupla di relazione con l'ordine parti
Ordine Parti	Entità	1	L	Prendo le informazioni delle parti ordinate
Ricambio	Relazione	3	L	Prendo la lista di parti ordinate

Parte	Entità	3	L	Prendo le informazioni delle parti sostituite
Costo in termini di operazioni elementari:				39
Costo giornaliero:				390

Ridondanze: non si ritiene necessario l'introduzione di ridondanze.

Gestione del termine dell'assistenza fisica

Descrizione: Durante l'assistenza fisica ci potrebbe essere la necessità di ordinare alcune componenti per poterle sostituire nel prodotto acquistato dal cliente. Questo ordine viene richiesto dal tecnico dopo che il preventivo è stato accettato. Quando l'ordine di pezzi viene consegnato al tecnico esso deve proporre un nuovo intervento al cliente in modo da poter riparare il dispositivo o per l'eventuale ritiro, nel caso in cui l'elettrodomestico è stato portato in un centro di assistenza. Si vuole dunque creare una funzionalità che permetta di segnalare qualora ci fosse la necessità di contattare il cliente per un nuovo intervento. Essa sarà suddivisa in 3 macro-operazioni.

1. Completamento dell'ordine delle parti

Descrizione: Quando l'ordine è stato consegnato si avrà la modifica della data effettiva all'interno dell'entità "ordine parti".

Input: CodiceOrdineParti

Output: Non necessario

Frequenza giornaliera: 5 richieste giornaliere

Porzione diagramma Interessato:

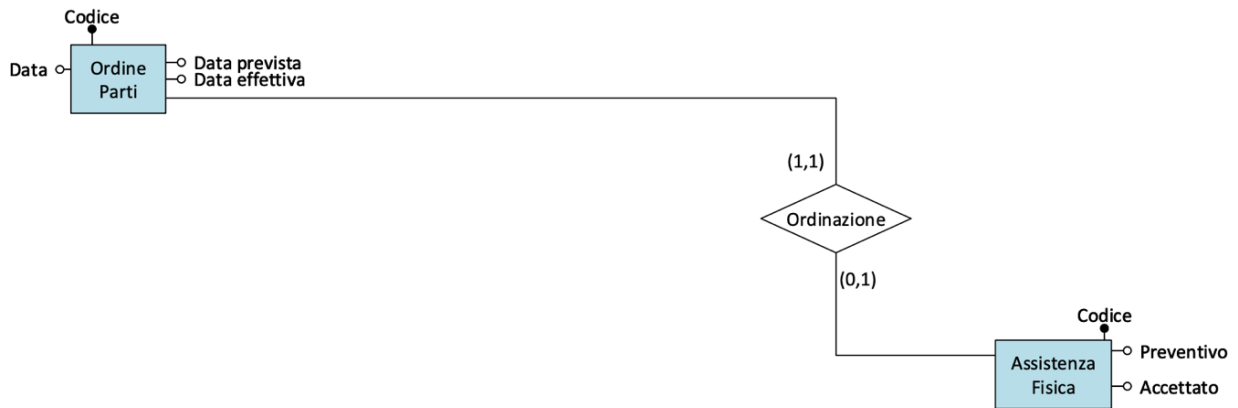


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Ordinazione	Relazione	10.764
Assistenza Fisica	Entità	16.560
Ordine Parti	Entità	10.764

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Ordinazione	Relazione	1	L	Prendo la tupla di relazione con assistenza fisica
Assistenza Fisica	Entità	1	L	Prendo le informazioni dell'assistenza fisica che ha richiesto all'ordine delle parti
Ordine Parti	Entità	1	S	Modifico la data effettiva.
Costo in termini di operazioni elementari:			4	
Costo giornaliero:			20	

Ridondanze

Non si ritiene necessario l'introduzione di ridondanze.

2. Aggiunta dell'intervento finale

Descrizione: Ogni volta che arriva una parte di ricambio, è necessario contattare il cliente affinché si possa concordare un nuovo intervento nell'ambito dell'assistenza fisica. Per fare questo, si devono andare a prendere quegli interventi per cui si è avuto un ordine parti ma non un intervento la cui data è successiva a quella di 'data effettiva' in ordine parti.

Input: Ticket

Output:

CodiceAssistenza

Frequenza giornaliera: 5 richieste giornaliere

Porzione diagramma Interessato:

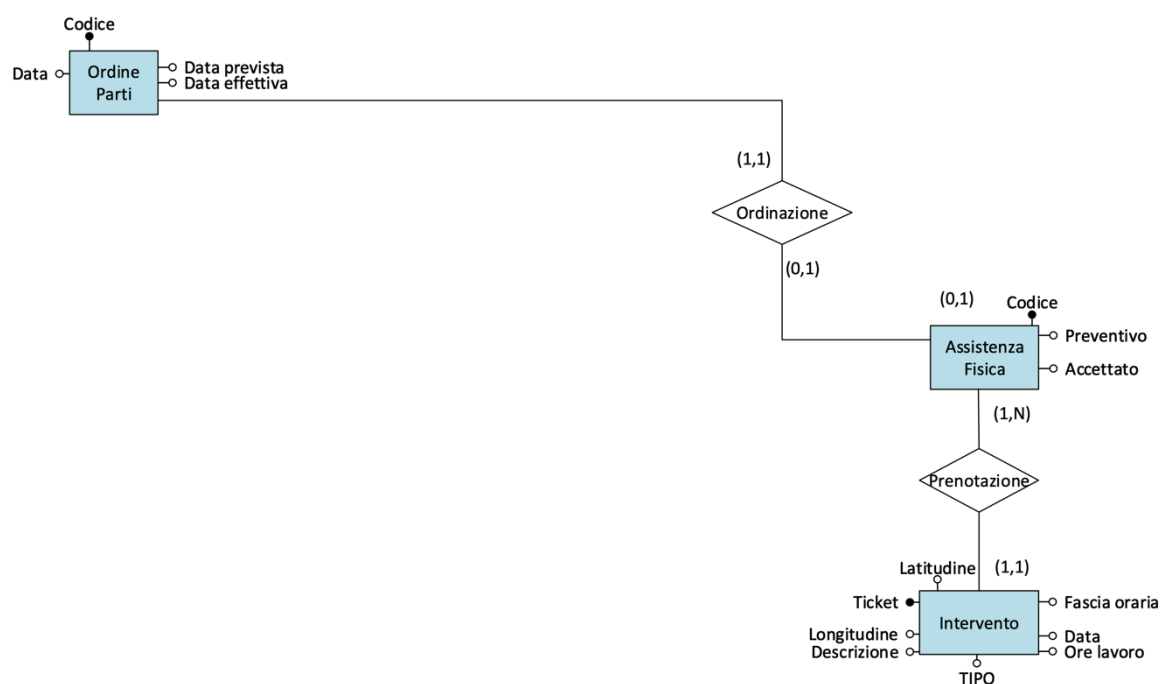


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Assistenza Fisica	Entità	16.560
Ordinazione	Relazione	10.764
Ordine Parti	Entità	10.764
Prenotazione	Relazione	33.120

Intervento	Entità	33.120
------------	--------	--------

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Assistenza Fisica	Entità	16.560	L	Si considerano tutte le assistenze fisiche
Ordinazione	Relazione	10.764	L	Per ciascuna di queste, si accede all'ordine parti eventualmente associato
Ordine Parti	Entità	10.764	L	Si preleva, per ciascuna assistenza, la data effettiva di arrivo delle parti
Prenotazione	Relazione	33.120	L	Si accede agli interventi per ciascuna assistenza fisica
Intervento	Entità	33.120	L	Per ognuna di questa, si cerca l'ultimo effettuato.
Costo in termini di operazioni elementari:			104.328 ¹	
Costo giornaliero:			521.640	

1. Non è affatto curioso il costo esorbitante di tale interrogazione. Infatti, si tratta di una query che necessita una verifica per ciascuna delle assistenze fisiche presenti nella base di dati.

Ridondanze

Il problema di accedere alle assistenze per cui non si è ancora effettuato la sostituzione delle parti di ricambio va ad incidere molto sul carico applicativo. La nostra idea è allora quella di sfruttare una Materialized View di appoggio. In questo modo, all'arrivo dell'ordine parti (quando, chiamando la prima delle sotto-operazioni elencate, si aggiunge la data effettiva dell'ordine) inseriamo un record in questa MV

che tiene conto delle assistenze fisiche per cui è necessario programmare un ulteriore intervento per il cambio di una parte. L'interrogazione si riconduce quindi ad una visualizzazione di questa tabella ausiliaria.

MemoProssimiInterventi(CodiceAssistenzaFisica)

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
MemoProssimiInterventi	MV	20 ²	L	Prendo la tupla di relazione con assistenza fisica
Costo in termini di operazioni elementari:			20	
Costo giornaliero:			100	

- Una stima realistica è che all'interno della MV vi siano sempre in media 30 codici di assistenze che hanno bisogno di una prenotazione.

Il numero di operazioni elementari risparmiate è

$$\Delta_{\text{read}} = n - n_{\text{RID}}^T = 521.640 - 100 = 521.540$$

La ridondanza, per sua natura, sarà gestita inserendo un record ogni volta che, come già detto, si ha l'aggiunta di 'data effettiva' in ordine parti, e provando a togliere un record ad ogni aggiunta di un intervento. La prima si effettua tante volte quanto l'uso della sotto-operazione precedente, quindi 5 volte al giorno. Per quanto riguarda il secondo, si stima che in media si abbiano 10 inserimenti in intervento al giorno.

Gestione per la modifica in Ordine Parti:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Ordine Parti	E	1	L	Verifico che l'aggiornamento sia quello necessario.
Ordinazione	R	1	L	Accedo alla relativa assistenza fisica.

Assistenza Fisica	L	1	L	Leggo il codice dell'assistenza fisica
MemoProssimiInterventi	MV	1	S	Aggiungo il relativo codice di assistenza fisica
Costo in termini di operazioni elementari:			5	
Costo giornaliero:			25	

Gestione per l'aggiunta in intervento:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Intervento	E	1	L	Accedo all'intervento appena inserito
Prenotazione	R	1	L	Accedo alla relativa assistenza fisica.
Assistenza Fisica	L	1	L	Leggo il codice dell'assistenza fisica.
MemoProssimiInterventi	MV	1	S	Provo ad eliminare un record.
Costo in termini di operazioni elementari:			5	
Costo giornaliero:			25	

Il costo giornaliero dell'operazione con la ridondanza sommato al costo per la consistenza della materialized view è

$$n_{rid}^T + o^A = 100 + 50 = 150$$

Che è minore di $n^A = 521.640$, il costo dell'operazione senza la ridondanza. Si sceglie quindi di mantenere la ridondanza.

3. Chiusura dell'assistenza fiscale

Descrizione: Una volta conclusa la riparazione si deve concludere la filiera emettendo una ricevuta fiscale, e dunque calcolando il costo effettivo ricordandosi che, nel caso in cui le parti modificate o il guasto diagnosticato sono coperti da garanzia allora la ricevuta fiscale emessa è vuota.

Input: Assistenza fisica

Output: Ricevuta fiscale

Codice ricevuta	Costo totale	Metodo di pagamento
-----------------	--------------	---------------------

Frequenza giornaliera: 5 richieste giornaliere

Porzione diagramma Interessato:

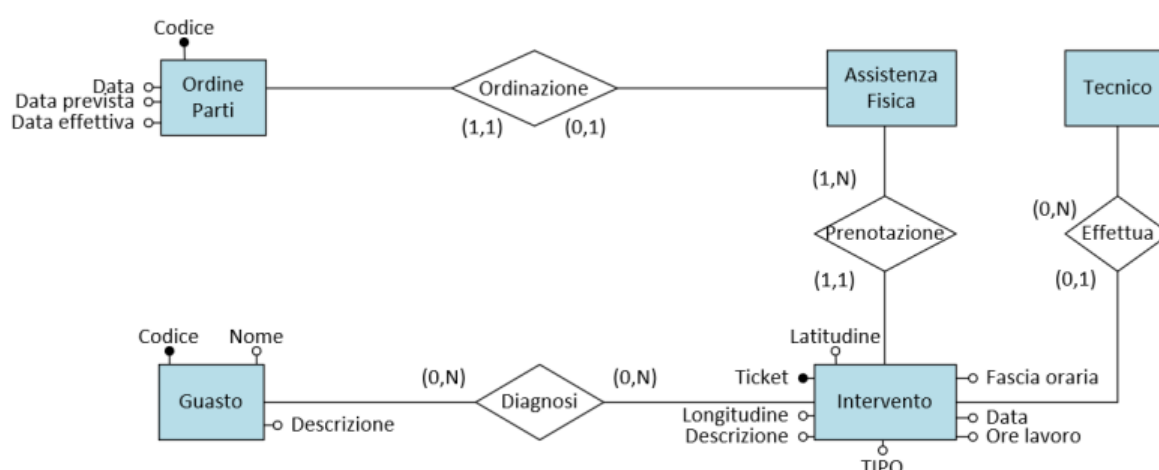


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Assistenza fisica	Entità	16.560
Prenotazione	Relazione	33.120
Intervento	Entità	33.120
Effettua	Relazione	33.120
Tecnico	Entità	10
Ordinazione	Relazione	10.764
Ordine Parti	Entità	10.764
Ricambio	Relazione	32.295
Parte	Entità	2.950

Fattura	Relazione	13.248
Ricevuta Fiscale	Entità	13.248

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Assistenza fisica	Entità	1	L	Prendo le informazioni dell'assistenza fisica e il preventivo
Prenotazione	Relazione	2	L	Prendo le relazioni verso gli interventi
Intervento	Entità	2	L	Prendo la lista di interventi eseguiti
Effettua	Relazione	2	L	Prendo la relazione verso il tecnico che ha eseguito l'intervento
Tecnico	Entità	2	L	Prendo le informazioni sul tecnico per poter calcolare il prezzo umano
Ordinazione	Relazione	1	L	Accedo all'ordine parti per quell'assistenza.
Ordine Parti	Entità	1	L	Leggo il codice ordine.
Ricambio	Relazione	3	L	In media, 3 parti per ciascun ordine.
Parte	Entità	3	L	Leggo il costo delle parti di ricambio.

Fattura	Relazione	1	S	Aggiungo la relazione per la ricevuta fiscale
Ricevuta fiscale	Entità	1	S	Aggiungo la tupla di ricevuta fiscale
Costo in termini di operazioni elementari:				
Costo giornaliero:			105	

Ridondanze

Non si ritiene necessario l'introduzione di ridondanze.

Visualizzazione giudizi prodotto

Descrizione: L'azienda ha a disposizione un portale web dal quale è possibile effettuare gli ordini. Un cliente, quando visita la pagina relativa ad un certo prodotto disponibile (si parla quindi delle varianti di prodotto), oltre a visualizzare delle immagini e una descrizione dello stesso, ha la possibilità di vedere il voto medio per ciascun giudizio sul prodotto. I voti dei singoli acquisti sono memorizzati legati alla singola recensione, quindi è necessario, ogni volta, andare ad effettuare il calcolo della media per ciascun giudizio.

Input: Variante di prodotto

Output: Prodotto inserito, Giudizio, Voto medio per quel giudizio su tale prodotto

Prodotto	Descrizione	Giudizio	Voto Medio
----------	-------------	----------	------------

Frequenza giornaliera: 500¹

Porzione diagramma Interessato:

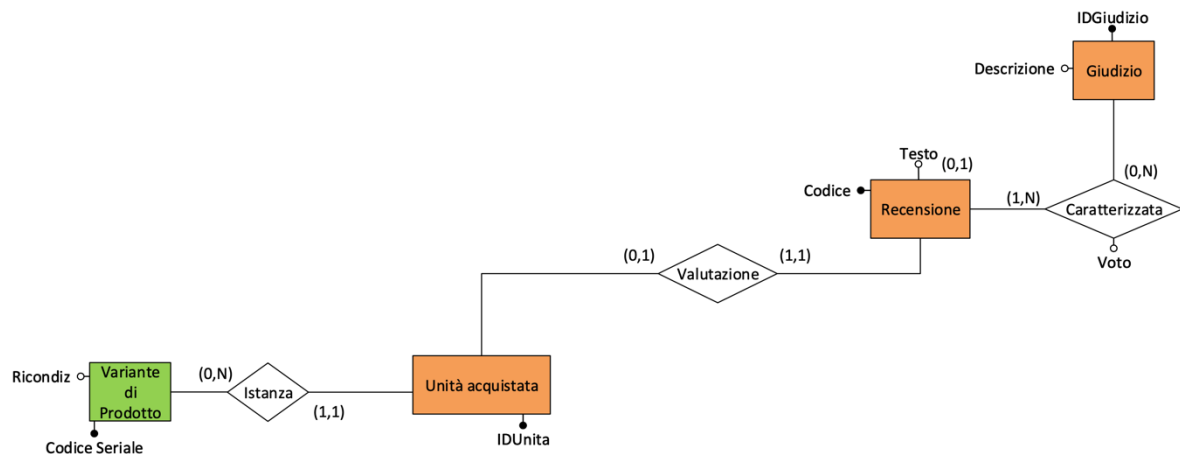


Tavola dei volumi interessata:

Oggetto	Tipo	Volume
Variante di prodotto	Entità	150
Istanza	Relazione	138.000
Unità acquistata	Entità	138.000
Valutazione	Relazione	69.000
Recensione	Entità	69.000
Caratterizzata	Relazione	345.000
Giudizio	Entità	5

Tavola degli accessi:

Nome Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazione	Motivazione
Variante di Prodotto	E	1	L	Lettura del prodotto inserito
Istanza	R	920 ²	L	Accedo alle unità acquistate del dato prodotto
Unità acquistata	E	920	L	Come sopra
Valutazione	R	460 ³	L	Accedo alle recensioni relative al prodotto
Recensione	E	460	L	Come sopra
Caratterizzata	R	2.300 ⁴	L	Leggo i voti

Giudizio	E	5	L	Leggo le descrizioni di ciascun giudizio
Costo in termini di operazioni elementari:			5.066	
Costo giornaliero:			2.533.000	

1. Avendo ipotizzato 50 ordini al giorno, che corrisponde a 100 unità acquistate al giorno, è ragionevole pensare che, per ogni unità comprata, vi siano altre 4 visualizzazioni che non si concludono con un acquisto.
2. In media, ogni prodotto è acquistato $138.000/150 = 920$ volte.
3. Un'ipotesi presente anche sulla tavola dei volumi è che, per ogni due unità acquistate, si inserisca una recensione
4. Ad ogni recensione sono associati 5 voti, uno per ciascun giudizio, quindi ottengo $460 * 5 = 2300$ accessi per prendere i voti

Ridondanze

Il costo eccessivo di tale operazione può essere smorzato in più modi. Una prima idea potrebbe essere quella di associare, per ciascuna variante di prodotto, le medie di ciascun giudizio, subito disponibili per la lettura. Questo, tuttavia, porterebbe ad un imponente costo di aggiornamento: le medie non si possono aggiornare subito, ma è necessario contare di nuovo il numero di recensioni per tale prodotto, sommare i valori di tutti i giudizi ed eseguire la divisione. Una soluzione alternativa potrebbe essere allora quella di inserire, all'interno della variante di prodotto, tanti campi quanti sono i giudizi che l'azienda mette a disposizione per le recensioni con il numero di recensioni effettuate, così che si possa subito calcolare la media accedendo alla variante. Tale idea comporta però un appesantimento della tabella variante di prodotto dovendo aggiungere $n+1$ colonne ad una tabella che risulta uno snodo cruciale per l'intero database. Abbiamo quindi optato per l'adozione di una materialized view volta proprio a salvare, per ciascun prodotto e per ciascun giudizio, la somma di tutti i giudizi e il numero di giudizi effettuati. In tale modo, un singolo accesso a questa tabella ci fornisce subito quanto richiesto, senza dover appesantire il database e l'aggiornamento prevede solo di dover aggiornare questi conteggi.

La materialized view in questione sarà nella forma:

GiudiziMV (CodiceSeriale, IDGiudizio, SommaVoti, countVoti)

Essendoci 5 giudizi e 150 prodotti, stimiamo il suo volume in $5*150=750$ istanze.

Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazioni	Motivazione
GiudiziMV	MV	5	L	Leggo tutto quello che riguarda i giudizi per un dato prodotto
Giudizio	Entità	5	L	Leggo la descrizione dei giudizi
Costo in termini di operazioni elementari:			10	
Costo Giornaliero			5.000	

Il numero di operazioni elementari risparmiate è

$$\Delta_{\text{read}} = n - n_{\text{RID}}^T = 2.533.000 - 5000 = 2.528.500$$

Per quanto riguarda la consistenza della ridondanza, ad ogni inserimento di una nuova recensione la MV perderà l'allineamento con la realtà. Dato che il numero di inserimenti giornalieri di una recensione non è eccessivo, si opta per un *immediate refresh*, rendendo la materialized view sempre aggiornata.

Costrutto	Tipo Costrutto	Numero operazioni	Tipo operazioni	Motivazione
Recensione	E	1	L	Leggo la recensione associata ai giudizi che sto inserendo
Valutazione	R	1	L	Risalgo all'unità a cui si riferisce la recensione
Unità acquistata	E	1	L	Come sopra
Istanza	R	1	L	Risalgo alla variante a cui si riferisce l'unità in questione
GiudiziMV	MV	5	S	Si aggiorna la MV modificando, per il dato prodotto, i campi 'Somma Voti' e 'Quanti Voti'.
Costo in termini di operazioni elementari:			14	
Costo Giornaliero			700 ⁵	

5. Sapendo che ogni due unità acquistate c'è una recensione, se le unità acquistate al giorno sono in media 100, effettuo l'inserimento di una recensione 50 volte al giorno, quindi il costo giornaliero è $14 \cdot 50 = 700$.

Il costo giornaliero dell'operazione con la ridondanza sommato al costo per la consistenza dell'attributo è

$$n_{rid}^T + o^A = 5000 + 700 = 5.700$$

Che è minore di $n^A = 2.533.000$, il costo dell'operazione senza la ridondanza. Si sceglie quindi di adottare la ridondanza.

6. Progettazione Logica

Dopo questa fase di progettazione concettuale, in cui abbiamo definito quelle che sono le specifiche della base di dati, si passa alla traduzione dello schema ER ristrutturato.

1. Schema logico

Prodotto(IDProdotto, Nome, Marca, Modello, DataProduzione, Prezzo, Predisposizione, SdR, CodiceTestRoot)
Variante(IDVariante, Prezzo, Descrizione, IDProdotto)
VarianteDiProdotto (CodiceSeriale, Ricondizionato, IDProdotto)
Modello(IDVariante, CodiceSeriale)
Parte(CodiceParte, Nome, Prezzo, Peso, CdS, Intermedio, IDProdotto)
Materiale(IDMateriale, Nome, VKG)
Composizione(CodiceParte, IDMateriale, Quantità)
Faccia(IDFaccia, IDProdotto)
Giunzione(IDGiunzione, Nome)
Caratteristica(IDCaratteristica, Descrizione, UDM, Valore, IDGiunzione)
Utensile(IDUtensile, Nome)
Uso(IDOperazione, IDUtensile)
Operazione(IDOperazione, Nome, Priorità, ParteA, ParteB, IDGiunzione, IDFaccia)
SequenzaDiOperazioni(IDSequenza, Indicatore, TempoStazione, IDProdotto)
Fase(IDStazione, IDOperazione)
Stazione(IDStazione, Precedenza, Operatore, IDSequenza)
Operatore(Badge, Nome, Cognome)
OperazioneCampione(Nome)
Capacità(Badge, Operazione, TempoMedio, Varianza)
UnitàPerse(IDStazione, CodiceLotto, Quantità)
Lotto(CodiceLotto, DataPrevista, DataEffettiva, SedeProduzione, Quantità,

Rimanente, IDVariante, IDSequenza)
 StoccaggioAttuale(CodiceLotto, IDUbicazione, DataInizio)
 Stoccaggiostorico(CodiceLotto, IDUbicazione, DataInizio, DataFine)
 Ubicazione(IDUbicazione, Piano, Stanza, Scaffale, CodiceMagazzino)
 Magazzino(CodiceMagazzino, Capienza, Predisposizione)
 FormulaGaranzia(CodiceGaranzia, Periodo, Costo, Tipo)
 Applicabile(CodiceGaranzia, IDProdotto)
 Estensione(IDUnità, CodiceGaranzia)
 UnitàAcquistata(IDUnità, CodiceSeriale, CodiceOrdine)
 Ordine(CodiceOrdine, Indirizzo, Timestamp, DataIncasso, Costo, Stato, Username)
 Account(Username, Email, DataIscrizione, Credito, Hash, Salt, CodiceFiscale,
 CodiceDomanda, Risposta)
 DomandaSicurezza(CodiceDomanda, Testo)
 Utente(CodiceFiscale, Nome, Cognome, Telefono, Indirizzo, NumeroDocumento,
 TipoDocumento, ScadenzaDocumento, EnteDocumento)
 Spedizione(Tracking, DataSpedizione, DataPrevista, DataEffettiva, Stato, IDOrdine)
 Passaggio(Tracking, IDHub, Data, Ora)
 Hub(IDHub, Città)
 Recensione(CodiceRecensione, Testo, IDUnità)
 Caratterizzata(CodiceRecensione, IDGiudizio, Voto)
 Giudizio(IDGiudizio, Descrizione)
 RichiestaReso(IDRichiesta, Accettato, IDUnità, CodiceMotivazione)
 Motivazione(CodiceMotivazione, Nome, Descrizione, Tipo)
 Reso(CodiceReso, Ricondizionato, IDRichiesta, CodiceMagazzino)
 Guasto(CodiceGuasto, Nome, Descrizione)
 Specifica(CodiceGuasto, CodiceGaranzia)
 CodiceErrore(CodiceErrore, IDProdotto, CodiceGuasto)
 Soluzione(CodiceErrore, IDProdotto, CodiceRimedio)
 Rimedio(CodiceRimedio, Descrizione)
 Domanda(IDDomanda, Testo, TipoSi, TipoNo, NextYes, NextNo)
 EndYes(IDDomanda, CodiceRimedio)
 EndNo(IDDomanda, CodiceRimedio)
 AssistenzaVirtuale(IDProdotto, IDGuasto, IDDomanda)
 AssistenzaFisica(CodiceAssistenza, Preventivo, Accettato, IDUnità)
 RicevutaFiscale(CodiceRicevuta, MDP, CostoTotale, CodiceAssistenza)
 Intervento(Ticket, Descrizione, Data, OreLavoro, FasciaOraria, Longitudine,
 Latitudine, Tipo, BadgeTecnico, CodiceAssistenza)
 Diagnosi(Ticket, CodiceGuasto)
 Tecnico(Badge, Nome, Cognome, CostoOrario)
 OrdineParti(CodiceOrdine, Data, DataPrevista, DataEffettiva, BadgeTecnico,
 CodiceAssistenzaFisica)
 Ricambio(CodiceOrdineParti, CodiceParte)

Test(CodiceTest, Nome, PdS, CodicePadre)
 Fallimento(CodiceTest, CodiceReso, Sostituzione)
 Riguarda(CodiceTest, CodiceParte)

2. Vincoli di integrità referenziale

Avendo ormai definito le relazioni (a meno di normalizzazione) che andranno a costituire la nostra base di dati, un approccio basato sulle aree può sembrare un po' forzato, dal momento in cui non vi è più tra queste una marcata separazione. Tuttavia, ignorando gli effetti bordo, quello che si va a rappresentare in tale sezione sono proprio i vincoli presenti per il modo con cui abbiamo costruito lo schema ER, e crediamo possa essere anche intuitivo riferirci a quello durante l'esposizione. In nome di ciò, per ciascuna riga, sulla prima colonna troviamo l'attributo della relazione (nel formato *RELAZIONE.ATTRIBUTO*), nella seconda l'elemento a cui si riferisce, come vuole la definizione di vincolo di integrità referenziale: il valore assunto dai record per gli attributi di sinistra deve apparire tra i record degli attributi di destra.

Attributo	Reference
Prodotto.CodiceTestRoot	Test.CodiceTest
Variante.IDProdotto	Prodotto.IDProdotto
VarianteDiProdotto.IDProdotto	Prodotto.IDProdotto
Modello.IDVariante	Varante.IDVariante
Modello.CodiceSeriale	VarianteDiProdotto.CodiceSeriale
Parte.IDProdotto	Prodotto.IDProdotto
Composizione.CodiceParte	Parte.CodiceParte
Composizione.IDMateriale	Materiale.IDMateriale
Faccia.IDProdotto	Prodotto.IDProdotto
Caratteristica.IDGiunzione	Giunzione.IDGiunzione
Uso.IDOperazione	Operazione.IDOperazione
Uso.IDUtensile	Utensile.IDUtensile
Operazione.ParteA	Parte.CodiceParte
Operazione.ParteB	Parte.CodiceParte
Operazione.IDGiunzione	Giunzione.IDGiunzione
Operazione.IDFaccia	Faccia.IDFaccia
SequenzaDiOperazioni.IDProdotto	Prodotto.IDProdotto
Fase.IDOperazione	Operazione.IDOperazione
Fase.IDStazione	Stazione.IDStazione
Stazione.IDSequenza	SequenzaDiOperazioni.IDSequenza
Stazione.Operatore	Operatore.Badge
Capacita.Badge	Operatore.Badge

Capacita.Operazione	OperazioneCampione.Nome
UnitaPerse.IDStazione	Stazione.IDStazione
UnitaPerse.CodiceLotto	Lotto.Codice
Lotto.IDVariante	Variante.IDVariante
Lotto.IDSequenza	SequenzaDiOperazioni.IDSequenza
StoccaggioAttuale.CodiceLotto	Lotto.CodiceLotto
StoccaggioAttuale.IDUbicazione	Ubicazione.IDUbicazione
StoccaggioStorico.CodiceLotto	Lotto.CodiceLotto
StoccaggioStorico.IDUbicazione	Ubicazione.IDUbicazione
Ubicazione.CodiceMagazzino	Magazzino.CodiceMagazzino

Attributo	Reference
Applicabile.CodiceGaranzia	FormulaGaranzia.CodiceGaranzia
Applicabile.IDProdotto	Prodotto.IDProdotto
Estensione.IDUnita	UnitàAcquistata.IDUnita
Estensione.CodiceGaranzia	FormulaGaranzia.CodiceGaranzia
UnitàAcquistata.CodcieSeriale	VarianteDiProdotto.CodiceSeriale
UnitàAcquistata.CodiceOrdine	Ordine.CodiceOrdine
Ordine.Username	Account.Username
Account.CodiceFiscale	Utente.CodiceFiscale
Account.CodiceDomanda	DomandaSicurezza.CodiceDomanda
Spedizione.IdOrdine	Ordine.CodiceOrdine
Passaggio.IDHub	Hub.IDHub
Passaggio.Tracking	Spedizione.Tracking
Recensione.IDUnita	UnitàAcquistata.IDUnita
Caratterizzata.IDGiudizio	Giudizio.IDGiudizio
RichiestaReso.IDUnita	UnitàAcquistata.IDUnita
RichiestaReso.CodiceMotivazione	Motivazione.CodiceMotivazione
Reso.IDRichiesta	RichiestaReso.IDRichiesta
Reso.CodiceMagazzino	Magazzino.CodiceMagazzino

Attributo	Reference
Specifica.CodiceGuasto	Guasto.CodiceGuasto
Specifica.CodiceGaranzia	Garanzia.CodiceGaranzia
CodiceErrore.IDProdotto	Prodotto.IDProdotto
CodiceErrore.CodiceGuasto	Guasto.CodiceGuasto
Soluzione.IDProdotto	CodiceErrore.IDProdotto
Soluzione.CodiceRimedio	CodiceErrore. CodiceRimedio
Soluzione.CodiceGuasto	CodiceErrore.CodiceGuasto,
Domanda.NextYes	Domanda.IDDomanda
Domanda.NextNo	Domanda.IDDomanda

EndYes.IDDomanda	Domanda.IDDomanda
EndYes.CodiceRimedio	Rimedio.CodiceRimedio
EndNo.IDDomanda	Domanda.IDDomanda
EndNo.CodiceRimedio	Rimedio.CodiceRimedio
AssistenzaVirtuale.IDProdotto	Prodotto.IDProdotto
AssistenzaVirtuale.IDGuasto	Guasto.CodiceGuasto
AssistenzaVirtuale.IDDomanda	Domanda.IDDomanda
AssistenzaFisica.IDUnita	UnitàAcquistata.IDUnita
RicevutaFiscale.CodiceAssistenza	AssistenzaFisica.CodiceAssistenza
Intervento.BadgeTecnico	Tecnico.Badge
Intervento.CodiceAssistenza	AssistenzaFisica.CodiceAssistenza
Diagnosi.Ticket	Intervento.Ticket
Diagnosi.CodiceGuasto	Guasto.CodiceGuasto
OrdineParti.BadgeTecnico	Tecnico.Badge
OrdineParti.CodiceAssistenza	AssistenzaFisica.CodiceAssistenza
Ricambio.CodiceOrdineParti	OrdineParti.Codice
Ricambio.CodiceParte	Parte.CodiceParte

Attributo	Reference
Test.CodicePadre	Test.CodiceTest
Fallimento.CodiceTest	Test.CodiceTest
Fallimento.CodiceReso	Reso.CodiceReso
Riguarda.CodiceTest	Test.CodiceTest
Riguarda.CodiceParte	Parte.CodiceParte

3. Vincoli di integrità generici

Inseriamo qui l'elenco di tutte le business rule (vincoli di integrità generici) che è necessario rispettare per il mantenimento della consistenza della base di dati.

1. La variante di prodotto è associata solo alle varianti relative a quel prodotto.
2. Le operazioni assegnate ad una stazione sono svolte tutte sulla stessa faccia.
3. Le operazioni assegnate ad una stazione sono tutte relative allo stesso prodotto della sequenza di operazioni.
4. Le parti intermedie non hanno collegamenti con materiale
5. [*] Ogni stazione di una linea ha una precedenza unica tra tutte le stazioni della stessa linea.
6. Ogni stazione ha solo operazioni con priorità maggiore o uguale delle operazioni nelle stazioni con precedenza minore.
7. Ogni operazione relativa ad un prodotto è usata una ed una sola volta per ogni sequenza di operazione relativa a quel prodotto.

8. [*] Ogni lotto ha associato una sequenza di operazioni attribuita al prodotto che compone il lotto.
9. Ogni operatore lavora ad una sola stazione per ogni sequenza di prodotto.
10. [*] Tempo medio e varianza nelle operazioni campione sono valori strettamente positivi.
11. Ogni lotto si trova in un magazzino con predisposizione maggiore o uguale a quella del prodotto.
12. Appena un ordine viene aggiunto, ha come stato "in processazione".
13. Un ordine che passa agli stati successivi ha associata almeno una unità acquistata.
14. [*] Nel passaggio da 'pendente' o 'in processazione' a 'in preparazione', devono essere disponibili tra i lotti abbastanza unità dei prodotti che si vogliono acquistare.
15. [*] Se un ordine supera lo stato 'in processazione', il credito dell'utente è sufficiente ad ultimare l'ordine.
16. Si possono eliminare ordini e unità solo se si è nello stato 'in processazione'.
17. Il passaggio della consegna allo stato 'consegnata' avviene dopo che è passato almeno per un hub.
18. Se un ordine si trova in "in consegna" o "consegnato" non vi si possono aggiungere hub.
19. Può esistere una spedizione solo se l'ordine è in "spedito".
20. Posso aggiungere una recensione solo se la spedizione è in "consegnato".
21. [*] Un giudizio assume valori compresi tra 1 e 5.
22. [*] Una garanzia temporale non ha guasti associati ad essa.
23. Una formula di garanzia associata ad una unità si riferisce a quelle applicabili all'unità stessa.
24. Se la richiesta di reso ha come motivazione il diritto di recesso deve essere compiuto entro 30 giorni dalla data di arrivo.
25. Se la richiesta di reso ha come motivazione il diritto di recesso allora è automaticamente accettata.
26. Il reso può esistere solo se la richiesta di reso è stata accettata.
27. Una volta che il reso viene ricondizionato deve essere cancellata la relazione con il magazzino.
28. Gli stati dell'assistenza virtuale devono essere coerenti con i valori di 'nextYes', 'nexyNo' e con le associazioni 'endYes' e 'endNo'.
29. Un intervento ha "fascia oraria", "latitudine", "longitudine" solo se è del tipo "a casa".
30. [*] Non possono esserci più interventi per lo stesso giorno e stessa fascia oraria di quanti siano i tecnici.
31. [*] Se il preventivo è zero la ricevuta fiscale deve essere zero, cioè coperto da garanzia.
32. Non ci può essere un ordine delle parti se il preventivo non è accettato.

- 33. Un ordine parti deve comprendere solo parti relative all'unità acquistata.
- 34. Un ordine parti non può comprendere parti intermedie.
- 35. Non si possono aggiungere altri interventi ad una procedura di assistenza fisica conclusa.
- 36. Ogni test deve riguardare solo parti che compongono il prodotto associato al test-tree.

7. Analisi delle dipendenze funzionali con normalizzazione

Una volta ottenuto lo schema della nostra base di dati, ci preoccupiamo della normalizzazione dello stesso. A dire la verità, abbiamo utilizzato un approccio volto alla forma normale fin dalla progettazione concettuale, cercando di scollegare concetti diversi, anche a costo di rendere, a livello visivo, il database più articolato. Un esempio: nel diagramma ER c'è una relazione che lega 'Stazione' con 'Operazione', una seconda che lega stazione con la relativa sequenza di operazioni. Inizialmente, avevamo optato per una relazione ternaria che collegasse i tre elementi. L'eventuale traduzione della stessa sarebbe stata

Fase(Stazione, Operazione, Sequenza)

Con le seguenti dipendenze funzionali:

Stazione -> Sequenza

Operazione, Sequenza -> Stazione

Tale relazione, sarebbe stata 3NF, ma, prima ancora di tradurre, ci siamo resi conto che non è in BCNF, e nemmeno vi si può portare in modo immediato. A ragione di ciò, abbiamo scartato tale opzione.

1. Dipendenze funzionali

Prodotto(IDProdotto, Nome, Marca, Modello, DataProduzione, Prezzo, Predisposizione, SdR, CodiceTestRoot)

IDProdotto -> Nome, Marca, Modello, DataProduzione, Prezzo, Predisposizione, SdR, CodiceTestRoot

La relazione è in forma normale di Boyce-Codd

Variante(IDVariante, Prezzo, Descrizione, IDProdotto)

IDVariante -> Prezzo, Descrizione, IDProdotto

La relazione è in forma normale di Boyce-Codd

VarianteDiProdotto(CodiceSeriale, Ricondizionato, IDProdotto)

CodiceSeriale -> Ricondizionato, IDProdotto

La relazione è in forma normale di Boyce-Codd

Modello(IDVariante, CodiceSeriale)

Non vi sono dipendenze funzionali all'interno della relazione

Parte(CodiceParte, Nome, Prezzo, Peso, CdS, Intermedio, IDProdotto)

CodiceParte -> Nome, Prezzo, Peso, CdS, Intermedio, IDProdotto

La relazione è in forma normale di Boyce-Codd

Materiale(IDMateriale, Nome, VKG)

IDMateriale -> Nome, VKG

La relazione è in forma normale di Boyce-Codd

Composizione(CodiceParte, IDMateriale, Quantità)

CodiceParte, IDMateriale -> Quantità

La relazione è in forma normale di Boyce-Codd

Faccia(IdFaccia, IDProdotto)

IdFaccia -> IDProdotto

La relazione è in forma normale di Boyce-Codd

Giunzione(IDGiunzione, Nome)

IDGiunzione -> Nome

La relazione è in forma normale di Boyce-Codd

Caratteristica(IDCaratteristica, Descrizione, UDM, Valore, IDGiunzione)

IDCaratteristica -> Descrizione, UDM, Valore, IDGiunzione

La relazione è in forma normale di Boyce-Codd

Utensile(IDUtensile, Nome)

IDUtensile -> Nome

La relazione è in forma normale di Boyce-Codd

Uso(IDOperazione, IDUtensile)

Non vi sono dipendenze funzionali all'interno della relazione

Operazione(IDOperazione, Nome, Priorità, ParteA, ParteB, IDGiunzione, IDFaccia)

IDOperazione -> Nome, Priorità, ParteA, ParteB, IDGiunzione, IDFaccia

ParteA, ParteB -> IDOperazione, Nome, Priorità, IDGiunzione, IDFaccia

La relazione è in forma normale di Boyce-Codd (è vero che vi sono due dipendenze funzionali, ma entrambe risultano essere chiavi per la relazione, indipendentemente dalla chiave primaria che abbiamo scelto).

SequenzaDiOperazioni(IDSequenza, Indicatore, TempoStazione, IDProdotto)

IDSequenza -> Indicatore, TempoStazione, IDProdotto

La relazione è in forma normale di Boyce-Codd

Fase(IDStazione, IDOperazione)

Non vi sono dipendenze funzionali all'interno della relazione

La relazione è in forma normale di Boyce-Codd

Stazione(IDStazione, Precedenza, Operatore, IDSequenza)

IDStazione -> Precedenza, Operatore, IDSequenza

La relazione è in forma normale di Boyce-Codd

Operatore(Badge, Nome, Cognome)

Badge -> Nome, Cognome

La relazione è in forma normale di Boyce-Codd

OperazioneCampione(Nome)

Non vi sono dipendenze funzionali all'interno della relazione

Capacità(Badge, Operazione, TempoMedio, Varianza)

Badge, Operazione -> TempoMedio, Varianza

La relazione è in forma normale di Boyce-Codd

UnitàPerse(IDStazione, CodiceLotto, Quantità)

IDStazione, CodiceLotto -> Quantità

La relazione è in forma normale di Boyce-Codd

Lotto(CodiceLotto, DataPrevista, DataEffettiva, SedeProduzione, Quantità, Rimanente, IDVariante, IDSequenza)

CodiceLotto -> DataPrevista, DataEffettiva, SedeProduzione, Quantità,

Rimanente, Ricondizionato, IDVariante, IDSequenza

La relazione è in forma normale di Boyce-Codd

StoccaggioAttuale(CodiceLotto, IDUbicazione, DataInizio)

CodiceLotto, IDUbicazione -> DataInizio

La relazione è in forma normale di Boyce-Codd

StoccaggioStorico(CodiceLotto, IDUbicazione, DataInizio, DataFine)

CodiceLotto, IDUbicazione -> DataInizio, DataFine

La relazione è in forma normale di Boyce-Codd

Ubicazione(IDUbicazione, Piano, Stanza, Scaffale, CodiceMagazzino)

IDUbicazione -> Piano, Stanza, Scaffale, CodiceMagazzino

La relazione è in forma normale di Boyce-Codd

Magazzino(CodiceMagazzino, Capienza, Predisposizione)

CodiceMagazzino -> Capienza, Predisposizione

La relazione è in forma normale di Boyce-Codd

Applicabile(CodiceGaranzia, IDProdotto)

Non vi sono dipendenze funzionali all'interno della relazione

Estensione(IDUnità, CodiceGaranzia)

Non vi sono dipendenze funzionali all'interno della relazione

UnitàAcquistata(IDUnità, CodiceSeriale, CodiceOrdine)

IDUnità -> CodiceSeriale, CodiceOrdine

La relazione è in forma normale di Boyce-Codd

Ordine(CodiceOrdine, Indirizzo, Timestamp, DataIncasso, Costo, Stato, Username)

CodiceOrdine -> Indirizzo, Timestamp, DataIncasso, Costo, Stato, Username

La relazione è in forma normale di Boyce-Codd

Account(Username, E-mail, DataIscrizione, Credito, Hash, Salt, CodiceFiscale, CodiceDomanda, Risposta)

Username -> E-mail, DataIscrizione, Credito, Hash, Salt, CodiceFiscale, CodiceDomanda, Risposta

CodiceFiscale -> Username, E-mail, DataIscrizione, Credito, Hash, Salt, CodiceDomanda, Risposta

La relazione è in forma normale di Boyce-Codd

DomandaSicurezza(CodiceDomanda, Testo)

CodiceDomanda -> Testo

La relazione è in forma normale di Boyce-Codd

Utente(CodiceFiscale, Nome, Cognome, Telefono, Indirizzo, NumeroDocumento, TipoDocumento, ScadenzaDocumento, EnteDocumento)

CodiceFiscale -> Nome, Cognome, Telefono, Indirizzo, NumeroDocumento, TipoDocumento, ScadenzaDocumento, EnteDocumento

La relazione è in forma normale di Boyce-Codd

Spedizione(Tracking, DataSpedizione, DataPrevista, DataEffettiva, Stato, IDOrdine)

Tracking -> DataSpedizione, DataPrevista, DataEffettiva, Stato, IDOrdine

IDOrdine -> Tracking, DataSpedizione, DataPrevista, DataEffettiva, Stato

La relazione è in forma normale di Boyce-Codd

Passaggio(Tracking, IDHub, Data, Ora)

Tracking, IDHub -> Data, Ora

La relazione è in forma normale di Boyce-Codd

Hub(IDHub, Citta)

IDHub -> Citta

La relazione è in forma normale di Boyce-Codd

Recensione(CodiceRecensione, Testo, IDUnità)

CodiceRecensione -> Testo, IDUnità

IDUnità -> CodiceRecensione, Testo

La relazione è in forma normale di Boyce-Codd

Caratterizzata(CodiceRecensione, IDGiudizio, Voto)

CodiceRecensione, IDGiudizio -> Voto

La relazione è in forma normale di Boyce-Codd

Giudizio(IDGiudizio, Descrizione)

IDGiudizio -> Descrizione

La relazione è in forma normale di Boyce-Codd

RichiestaReso(IDRichiesta, Accettato, IDUnità, CodiceMotivazione)

IDRichiesta -> Accettato, IDUnità, CodiceMotivazione

La relazione è in forma normale di Boyce-Codd

Motivazione(CodiceMotivazione, Nome, Descrizione, Tipo)

CodiceMagazzino -> Nome, Descrizione, Tipo

La relazione è in forma normale di Boyce-Codd

Reso(CodiceReso, Ricondizionato, IDRichiesta, CodiceMagazzino)

CodiceReso -> Ricondizionato, IDRichiesta, CodiceMagazzino

IDRichiesta -> CodiceReso, Ricondizionato, CodiceMagazzino

La relazione è in forma normale di Boyce-Codd

Guasto(CodiceGuasto, Nome, Descrizione)

CodiceGuasto -> Nome, Descrizione
La relazione è in forma normale di Boyce-Codd

Specifica(CodiceGuasto, CodiceGaranzia)
Non vi sono dipendenze funzionali all'interno della relazione.

CodiceErrore(CodiceErrore, IDProdotto, CodiceGuasto)
CodiceErrore, IDProdotto -> CodiceGuasto
La relazione è in forma normale di Boyce-Codd

Soluzione(CodiceErrore, IDProdotto, CodiceRimedio)
Non vi sono dipendenze funzionali all'interno della relazione

Rimedio(CodiceRimedio, Descrizione)
CodiceRimedio -> Descrizione
La relazione è in forma normale di Boyce-Codd

Domanda(IDDomanda, Testo, TipoSi, TipoNo, NextYes, NextNo)
IDDomanda -> Testo, TipoSi, TipoNo, NextYes, NextNo
La relazione è in forma normale di Boyce-Codd

EndYes(IDDomanda, CodiceRimedio)
Non vi sono dipendenze funzionali all'interno della relazione

EndNo(IDDomanda, CodiceRimedio)
Non vi sono dipendenze funzionali all'interno della relazione

AssistenzaVirtuale(IDProdotto, IDGuasto, IDDomanda)
Non vi sono dipendenze funzionali all'interno della relazione

AssistenzaFisica(CodiceAssistenza, Preventivo, Accettato, IDUnità)
Codice -> Preventivo, Accettato, IDUnità
La relazione è in forma normale di Boyce-Codd

RicevutaFiscale(CodiceRicevuta, MDP, CostoTotale, CodiceAssistenza)
CodiceRicevuta -> MDP, CostoTotale, CodiceAssistenza
CodiceAssistenza -> CodiceRicevuta, MDP, CostoTotale
La relazione è in forma normale di Boyce-Codd

Intervento(Ticket, Descrizione, Data, OreLavoro, FasciaOraria, Longitudine, Latitudine, Tipo, BadgeTecnico, CodiceAssistenza)

Ticket -> Descrizione, Data, OreLavoro, FasciaOraria, Longitudine, Latitudine, Tipo, BadgeTecnico, CodiceAssistenza
La relazione è in forma normale di Boyce-Codd

Diagnosi(Ticket, CodiceGuasto)

Non vi sono dipendenze funzionali all'interno della relazione

Tecnico(Badge, Nome, Cognome, CostoOrario)

Badge -> Nome, Cognome, CostoOrario

La relazione è in forma normale di Boyce-Codd

OrdineParti(CodiceOrdine, Data, DataPrevista, DataEffettiva, BadgeTecnico, CodiceAssistenza)

CodiceOrdine -> Data, DataPrevista, DataEffettiva, BadgeTecnico, CodiceAssistenza

CodiceAssistenza -> CodiceOrdine, Data, DataPrevista, DataEffettiva, BadgeTecnico

La relazione è in forma normale di Boyce-Codd

Ricambio(CodiceOrdineParti, CodiceParte)

CodiceOrdineParti -> CodiceParte

La relazione è in forma normale di Boyce-Codd

Test(CodiceTest, Nome, PdS, CodicePadre)

CodiceTest -> Nome, PdS, CodicePadre

La relazione è in forma normale di Boyce-Codd

Fallimento(CodiceTest, CodiceReso, Sostituzione)

CodiceTest, CodiceReso -> Sostituzione

La relazione è in forma normale di Boyce-Codd

Riguarda(CodiceTest, CodiceParte)

Non vi sono dipendenze funzionali all'interno della relazione

8. Implementazione su DBMS Oracle MySQL

In questo capitolo della documentazione si vuole fornire una sorta di indice per quanto riguarda lo script del database che abbiamo realizzato. In allegato, si trovano tre (+1, ma ne parleremo nel prossimo capitolo) file in formato .sql:

- a. eDevice.sql: In questo file vi sono tutte le operazioni DDL che riguardano la creazione e la definizione delle tabelle, con i vincoli di integrità referenziali e quei pochi vincoli riguardanti i domini degli attributi, espressi con la clausola CHECK.
- b. Operazioni.sql: Qui si trovano tutte le operazioni implementate, sia le 8 da noi proposte sia quelle che sono espressamente richieste dal testo del progetto. Si trovano inoltre alcuni dei trigger volti al rispetto dei vincoli di integrità generici.
- c. Popolazione.sql: Usando le operazioni DML e alcune delle funzioni che abbiamo realizzato, proponiamo una serie di valori delle tabelle per sperimentare quello che abbiamo realizzato, prima della messa in funzione del database.

In merito alle operazioni che si trovano nel secondo file, abbiamo scelto di commentare il codice dove ritenevamo necessari dei chiarimenti, soffermandoci tuttavia solo su quelle operazioni non precedentemente descritte. Per quanto concerne le 8 operazioni del capitolo 4, abbiamo eseguito alla lettera ciò che abbiamo esposto, e rimandiamo quindi a quella sezione per dei dubbi a riguardo. Sottolineiamo poi che l'intero database è stato testato, oltre che con i dati presenti nel terzo file, anche con una serie di casi limiti che garantissero il corretto funzionamento.

1. Indice del file Operazioni.sql

1. Verifica disponibilità delle unità per un ordine
2. Incasso complessivo dell'azienda in un intervallo temporale
3. Ricerca rimedi a partire da un codice di errore ed un prodotto
4. Inserimento di un lotto di prodotti e gestione stoccaggio
5. Tracking di una spedizione
6. Calcolo dei periodi di guasto relativi ad una unità acquistata
7. Gestione dell'intervento finale
8. Visualizzazione del giudizio complessivo di un prodotto

Seguono quelle richieste dal testo:

Ricavare le quantità di prodotti resi per ciascuna variante di prodotto, associati alle motivazioni

Questa funzionalità è stata implementata come una semplice query con raggruppamento, in modo da poter avere sulla stessa riga, non solo le quantità di prodotto rese, ma anche le motivazioni che hanno portato alla restituzione.

Controllo del numero di resi per avviare procedura di ricondizionamento

Grazie all'attributo 'soglia di ricondizionamento' presente in prodotto, è possibile sapere quali varianti hanno, nei vari magazzini dell'azienda, abbastanza unità rese da avviare una procedura di ricondizionamento. Tale funzionalità è stata implementata tramite una function di tipo NOT DETERMINISTIC, che permetta la verifica in modo più versatile anche all'interno di query complesse. Insieme alla funzione, c'è anche una procedura che si occupa di mostrare, per ciascuna variante di prodotto, se è possibile o meno il ricondizionamento.

Mostrare le fasce orarie disponibili per una giornata

Tramite il modo in cui abbiamo organizzato il concetto di fascia oraria nell'ambito degli interventi assistenza fisica, ci è stato possibile realizzare questa funzionalità che permette di sapere al cliente che richiede un intervento quali fasce orarie sono libere. Oltre alla temporary table utilizzata per tenere traccia degli identificativi associati a ciascuna fascia, il cuore della funzione è la query conclusiva, in cui isoliamo le fasce il cui numero di interventi giornalieri è minore del numero di tecnici disponibili.

```
INSERT INTO FasceOrarie
VALUES (1, 'Mattina 1'),
       (2, 'Mattina 2'),
       (3, 'Pomeriggio 1'),
       (4, 'Pomeriggio 2');

SELECT COUNT(*) INTO numeroTecnici
FROM Tecnico;

SELECT F.Nome
FROM FasceOrarie F
LEFT OUTER JOIN (
    SELECT *
    FROM Intervento I
    WHERE I.Data = GiornoTarget
) AS I ON I.FasciaOraria = F.ID
GROUP BY F.ID
HAVING SUM(IF(I.FasciaOraria IS NULL, 0, 1))
```

Funzionalità di report per analisi vendite

Come richiesto, abbiamo due tipi di analisi: la prima ci permette di sapere settimanalmente quali sono i prodotti che sono stati più volte coinvolti in ordini pendenti, la seconda restituisce i prodotti più venduti nella settimana.

La prima funzionalità è senza dubbio la più interessante, sia sul piano pratico che implementativo. Infatti, se un prodotto finisce più volte tra i pendenti, vuol dire che non si producono abbastanza lotti di questo: di conseguenza sarà necessario modificare i piani di produzione. Il testo richiede esplicitamente che i dati siano aggiornati settimanalmente. Abbiamo scelto di implementare una Materialized View che permetta di tenere traccia, per ciascuna variante di prodotto, di quante volte essa sia finita nella scorsa settimana tra gli ordini 'pendenti'. L'aggiornamento di una materialized view di questo tipo può essere fatto solo in due modi: o con un *immediate refresh* o con un *incremental refresh*. Infatti, una volta che un ordine passa dallo stato 'pendente' a 'in preparazione', non è più possibile sapere, nel database, se esso abbia transitato o meno dallo stato che ci interessa. Questo aspetto può essere compensato memorizzando immediatamente l'ordine come pendente, o direttamente nella MV o in una log table per un futuro aggiornamento. Abbiamo scelto proprio questa seconda via, facendo sì che il carico applicativo non risenta troppo di questa funzionalità.

Generazione di sequenze valide per la produzione

Tale procedura permette la generazione, per ciascun prodotto, di una sequenza di produzione valida, e che rispetti quindi i vincoli di precedenza tecnologica per come esposti nel capitolo 3 della documentazione. Un aspetto da tenere presente è che tale funzionalità non garantisce alcun parametro di efficienza della produzione. Proprio per questo, sarà necessaria un'ulteriore elaborazione del risultato, in modo specifico per ciascuna esigenza.

Assegnamento dei tecnici agli interventi

Questa funzione è stata spiegata anch'essa nel capitolo 3, durante la trattazione dell'assistenza fisica. Visto la difficoltà dell'implementazione di questa funzione con MySQL, abbiamo preferito scrivere preliminarmente l'algoritmo con un linguaggio a noi più comodo, C++, per avere un punto di riferimento più solido. Alleghiamo al progetto anche il file .cpp dove si trova l'algoritmo, anch'esso commentato.

Vincoli di integrità generici

Rimandiamo all'apposito paragrafo del capitolo 6 per l'elenco dei vincoli implementati. In particolare, sono quelli con il simbolo [*] che li precede.

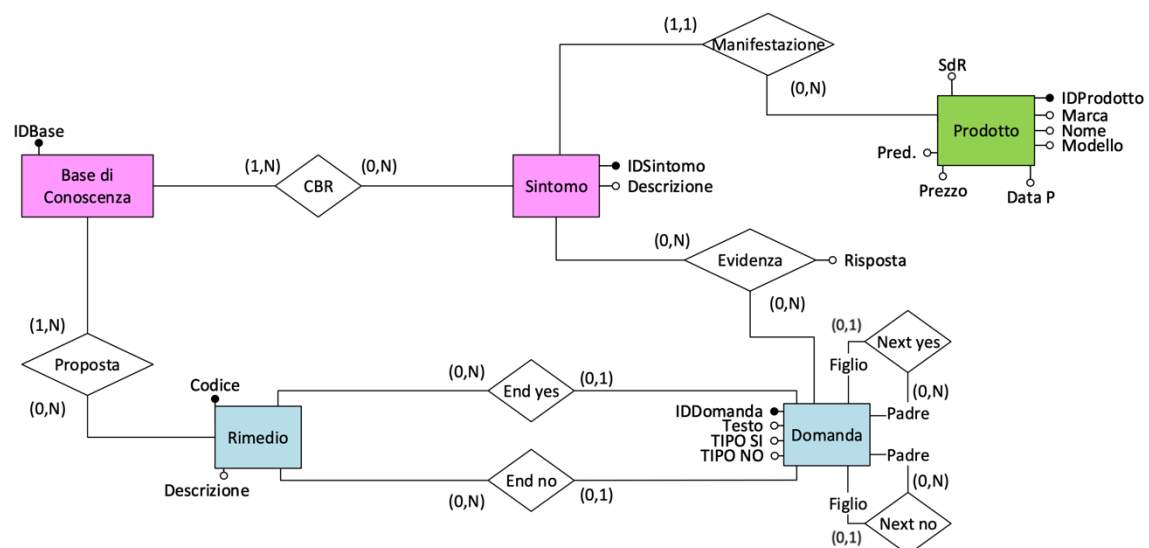
9. Data analytics

Quanto esposto finora termina la progettazione del database vero e proprio. Infatti, il sistema, a meno di concludere l'implementazione di tutti i controlli di consistenza, è operativo, e pronto per l'utilizzo quotidiano. All'interno di questa sezione proponiamo una serie di soluzioni per risolvere il problema dell'analisi dei dati, come richiesto dal testo del progetto.

Quanto presentato è frutto di una lenta maturazione, portata avanti lungo tutto il tempo di realizzazione del progetto. Se da una parte la libertà di assumere delle decisioni per la realizzazione di queste funzionalità ci ha un po' spiazzato, dall'altra ci ha permesso di ampliare i nostri orizzonti, portandoci a fare delle considerazioni, che esulano dal nostro quotidiano campo di interesse.

1. Diagnosi intelligente dei guasti: CBR

La prima data analytic la possiamo vedere come un complemento all'assistenza virtuale, se non come una vera e propria estensione quasi indipendente. Nonostante la similitudine tra le due funzionalità, si hanno caratteristiche abbastanza diverse, che impediscono una realizzazione della diagnosi intelligente dei guasti senza delle nuove aggiunte al database. Quella che proponiamo adesso è quindi la quinta area del database: l'area *base di conoscenza*, che, pur con le sue poche entità e relazioni, sarà identificata con il colore rosa.



Come è avvenuto per le aree precedenti, forniamo una breve illustrazioni delle fondamentali proprietà di ciascuna di queste, nonostante siamo consapevoli che il loro significato possa essere prevalentemente chiarito solo tramite la spiegazione del loro utilizzo.

Entità	Attributi	Identificatore
Sintomo	IDSintomo, Descrizione	IDSintomo
Base di conoscenza	IDBase	IDBase
Relazione	Attributi	Entità coinvolte
Proposta		Rimedio, Base di conoscenza
CBR		Base di conoscenza, Sintomo
Evidenza	Risposta	Sintomo, Domanda
Manifestazione		Sintomo, Prodotto

Sintomo

Un sintomo è una manifestazione di un guasto. In generale, ogni guasto ha tanti sintomi e, allo stesso modo, un sintomo può essere ricondotto a vari guasti. Ciononostante, non vi è alcuna relazione tra 'sintomo' e 'guasto'. Questo perché la nostra proposta in termini di data analytic ha un significato prettamente applicativo. La richiesta è quella di proporre una funzionalità che, a partire dai sintomi di un problema, fornisca dei possibili rimedi: perciò basta poter risalire solo ai rimedi per come risultano già inseriti nella base di conoscenza. Nel caso in cui ciò porti a risultati, potrà intervenire l'assistenza virtuale o quella fisica, ma l'obiettivo di tale strumento termina qua. Il sintomo ha un logico collegamento con un prodotto, nella misura in cui ciascuno di questi è specifico per una certa classe di prodotti, senza poter essere esteso ad altri. Infatti, un odore di bruciato da un computer o da una lavatrice è associato a guasti diversi (e quindi, soluzioni diverse): nel primo caso, evidentemente basta lasciare a riposo il telefono in modo che si possa raffreddare (presumibilmente si era avuto un utilizzo intensivo, magari sotto il sole), nel secondo caso il problema potrebbe essere molto più grave, e come minimo è necessario l'intervento dell'assistenza fisica. C'è anche una relazione tra sintomo e domanda. Questo aspetto sarà chiarito più avanti. Per adesso, basti sapere che la risposta di una certa domanda all'interno di un attraversamento dell'albero dell'assistenza virtuale corrisponde alla presenza di un certo sintomo. Per esempio, se si ha una risposta negativa a 'Si accende lo schermo del tuo dispositivo?' evidentemente si ha come sintomo il 'non si accende il dispositivo'. La nostra idea è quella di sfruttare l'assistenza virtuale per avere ulteriori dati per l'accrescimento della base di conoscenza, in modo da garantire una maggiore efficienza nella risoluzione di un problema.

Base di conoscenza

La base di conoscenza permette di associare ad una serie di sintomi di un prodotto un certo rimedio. Sarà l'elemento centrale che ci permetterà di implementare la funzionalità: data una serie di sintomi, si cerca nella base di conoscenza i casi già presenti, e sulla base di questi si cercano i vari rimedi. Ad ogni elemento dentro la base di conoscenza è associato almeno un sintomo e, dal capo opposto, almeno un rimedio.

Passando alla traduzione logica di questa quinta area, abbiamo:

Sintomo (IDSintomo, Descrizione, IDProdotto)

BaseDiConoscenza (IDBase)

CBR (IDBase, IDSintomo)

Proposta (IDBase, CodiceRimedio)

Evidenza (IDSintomo, IDDomanda, Risposta)

E, in termini di vincoli di integrità referenziale:

Attributo	Reference
Sintomo.IDProdotto	Prodotto.IDProdotto
CBR.IDBase	BaseDiConoscenza.IDBase
CBR.IDSintomo	Sintomo.IDSintomo
Proposta.IDBase	BaseDiConoscenza.IDBase
Proposta.CodiceRimedio	Rimedio.CodiceRimedio
Evidenza.IDSintomo	Sintomo.IDSintomo
Evidenza.IDDomanda	Domanda.IDDomanda

Le relazioni sono tutte in forma normale di Boyce-Codd.

Adesso passiamo alla spiegazione vera e propria della funzionalità. Essa è divisa in due parti: una di ricerca (lato client, ciò che permette all'utente di usare questo strumento), una di inserimento (lato server, permette di capire quando è necessario aggiungere un nuovo dato alla base di conoscenza).

L'utente fornisce in input il prodotto in questione e un numero illimitato di sintomi. Siano questi

$$A_1, A_2, \dots, A_n$$

Per semplicità, indichiamo con $|A|$ il numero di sintomi inseriti. All'interno della base di conoscenza, rispetto al dato prodotto vi saranno un certo numero di casi precedentemente risolti, ossia un gruppo formato dai sintomi che si sono manifestati

precedentemente e dai rimedi che sono stati utilizzati per risolvere il problema. Per ciascun caso C^i nella base di conoscenza, diciamo

$$B_1^i, B_2^i, \dots, B_{m^i}^i$$

I sintomi associati, e, come prima, il valore di m^i sarà pari a $|B^i|$. Per ciascun caso, è necessario definire uno score ψ^i , che consenta di definire obbiettivamente quale sia l'affinità tra il caso in input e quello già presente. Necessariamente, ciò dovrà avvenire sulla base dei sintomi in comune, aumentando lo score se il numero di questi è maggiore, diminuendolo se ci sono molti sintomi che non compaiono (infatti, ciò potrebbe significare che il guasto in esame, pur avendo molti sintomi in comune, è del tutto diverso da quello che dobbiamo risolvere). Sia allora

$$\vartheta^i = \#\{(a, b) \in A \times B \mid a = b\} = |A \cap B|$$

Definito il prodotto cartesiano tra i sintomi in input, insieme A , e quelli presenti nella base B , a noi interessa il numero di elementi che hanno lo stesso valore a destra e a sinistra. Ciò corrisponde proprio ad un sintomo in comune. Avendo ϑ^i , $|A|$ e $|B^i|$, possiamo definire lo score d'interesse come

$$\psi^i = \frac{\vartheta^i}{|A| + |B^i| - \vartheta^i}$$

Alcune particolarità di questo indice: se i sintomi in comune non ci sono, allora $\vartheta^i = 0$, e quindi vale 0. Nel caso in cui A e B hanno gli stessi sintomi senza alcuna estraneità, vale $\vartheta^i = |A| = |B^i|$ e quindi

$$\psi^i = \frac{|A|}{|A| + |B^i| - |A|} = \frac{|A|}{|A|} = 1$$

Queste due osservazioni ci fanno capire che l'indice può assumere valori tra 0 e 1, estremi inclusi, e che più siamo vicini all'1 più il caso precedente è simile a quello in esame. Ad esempio, supponiamo di avere un input i cui sintomi stanno tutti in $|B^i|$, il quale però ha anche altri sintomi. Si avrà $\vartheta^i = |A| < |B^i|$ e quindi

$$\psi^i = \frac{|A|}{|A| + |B^i| - |A|} = \frac{|A|}{|B^i|} < 1$$

Tale risultato ci permette proprio di confermare che l'indice dipende sia dal numero di sintomi in comune, ma anche dal numero di sintomi estranei. Come conclusione di questa breve analisi dell'indice scelto, si propone una piccola osservazione sul fatto

che il denominatore, supposti $|A|$ e $|B^i|$ non vuoti, non può mai annullarsi, e quindi tale divisione è sempre realizzabile.

Dalla teoria degli insiemi, vale la reazione

$$|X \cup Y| = |X| + |Y| - |X \cap Y|$$

Nel nostro caso quindi, il valore al denominatore può essere scritto come

$$|A| + |B^i| - \vartheta^i = |A \cup B^i|$$

Che vale 0 se e solo se entrambi gli insiemi sono vuoti. Poiché la consistenza di dati nel database ci evita tale possibilità, possiamo a tutti gli effetti stabilire che il denominatore è sempre diverso da zero.

Avendo i dati in input, il compito della funzionalità sarà quello di calcolare lo score per tutti i casi di guasto per quel dato prodotto presenti nella base di conoscenza. L'output saranno i relativi rimedi, ordinati sulla base dello score. Per esempio, dati i seguenti sintomi per un telefono:

$$A = \{Batteria\ con\ poca\ durata, Surriscaldamento\}$$

Un possibile output potrebbe essere

IDBase	Score	Rimedio
5	0.83	Disattivare il GPS
5	0.83	Modificare le impostazioni di luminosità
3	0.62	Disabilitare il risparmio energetico
13	0.46	Effettuare un hard reset
11	0.19	Sostituire la pellicola di protezione

Non è strano che, più lo score diminuisce, più si va fuori tema. Nei termini del testo, ciò corrisponde alla fase di *retrieve* e *reuse*, ma in certa misura anche a quella di *revise*: infatti, tramite ciò che è memorizzato all'interno della base di conoscenza, di fornisce all'utente la possibilità di fare vari test, e capire quali possono essere i rimedi più adatti al suo caso. Se questo può essere positivo per l'utente, dall'altra parte consente a noi di avere ulteriori dati per l'ampliamento della base, come vedremo in un attimo.

La seconda idea che sta dietro la nostra implementazione della base di conoscenza è che essa sia popolata a partire dalle interazioni che gli utenti hanno avuto con gli strumenti di assistenza, comprendendo in tal senso sia questa

funzionalità che l'assistenza virtuale. In nome di ciò, associato alla ricerca vi è l'inserimento di dati all'interno della base di conoscenza.

Questo può avvenire immediatamente dopo che un utente è riuscito a risolvere un problema. Egli presentava una serie di sintomi A_1, A_2, \dots, A_n e, tramite i rimedi R_1, R_2, \dots, R_m , è stato capace di risolvere il guasto. Un'analisi dei casi già presenti ci consentirà di capire se vale la pena inserire quello in esame dentro la base. In particolare, poiché la ricerca avviene sulla base dello score ψ^i , abbiamo scelto di usare lo stesso dato per capire se vale la pena l'inserimento. L'approccio sarà questo: dato un caso, si calcolano tutti gli score con i casi già presenti sulla base dei guasti. Questi saranno

$$\psi^1, \psi^2, \dots, \psi^k$$

Poiché l'inserimento deve basarsi non solo sulla differenza tra i sintomi, ma anche su quella tra i rimedi, definiamo un secondo score, ζ^i , che sia l'equivalente dei sintomi per i rimedi. Quindi, detto R l'insieme dei rimedi in input e RB^i quelli già presenti per il caso i -esimo nella base di conoscenza, sia

$$\zeta^i = \frac{|R \cap RB^i|}{|R| + |RB^i| - |R \cap RB^i|}$$

Le considerazioni fatte per ψ^i valgono anche in questo caso: lo score dei rimedi sarà un valore compreso tra 0 e 1, tanto più grande quanto è affine il caso rispetto ai rimedi in input.

Definiamo lo score da analizzare per l'inserimento come la media tra i due score:

$$\eta^i = \frac{\zeta^i + \psi^i}{2}$$

Detto questo, si procederà con l'inserimento nella base di conoscenza solo se si verifica la seguente situazione:

$$\text{MAX}\{\eta^1, \eta^2, \dots, \eta^k\} \leq 0.5$$

Questo mi sta proprio a significare che i dati già presenti devono essere abbastanza diversi da quelli che si vogliono inserire, sia sulla base dei rimedi sia sulla base dei sintomi. Il discriminante di 0.5 è stato scelto come un valore ritenuto abbastanza intermedio, ma senza la certezza empirica che possa essere valido per la nostra applicazione. Per tale motivo, invitiamo a rivalutare questo valore dopo la messa in funzione della base di conoscenza, in modo tale che si possano inserire casi effettivamente utili senza risultare ridondati (cosa che porterebbe, oltre che ad un

disorientamento di fronte ai dati, pure ad un eccessivo sovraccarico in termini di memoria occupata).

A questo punto, vogliamo spiegare il motivo della correlazione tra assistenza virtuale e base di conoscenza tramite la relazione 'Evidenza'. La nostra idea è che una parte di popolamento possa provenire proprio dall'uso dell'assistenza virtuale. Infatti, la risposta ad una certa domanda può essere l'evidenza di un sintomo, come già illustrato. Ciò significa che, se l'uso della funzionalità porta ad un risultato che non sia l'assistenza fisica, avremo un'ottima lista di sintomi + rimedio che permetterà l'eventuale accrescimento della base di conoscenza. Ma il processo può avvenire anche al contrario, e da ciò si spiega l'uso del termine 'complementari' per spiegare il legame tra assistenza fisica e base di conoscenza per i guasti usati poco sopra. Infatti, l'analisi dei casi può portare ad un miglioramento dei percorsi dell'assistenza virtuale, aggiungendo domande e relativi rimedi. Ciò potrebbe richiedere l'ausilio di esperti nell'analisi dei dati, ma l'implementazione della nostra data analytic rende tutto accessibile e immediatamente utilizzabile per tale scopo.

2. Efficienza del processo

La seconda data analytic richiede un metodo per l'analisi della qualità dei processi produttivi. Indubbiamente, vi sono vari punti di vista sotto il quale interpretare questa richiesta. Per esempio, potrebbe essere utile studiare l'efficienza del singolo operatore, osservando il numero di unità perse nelle stazioni in cui lavora in relazione ai valori salvati grazie all'entità 'operazioni campione'.

Sicuramente, una cosa di cui ci siamo subito resi conto è che era necessario far coesistere due aspetti in tale analisi: qualcosa che fosse relativo alla realtà di interesse, un altro che potesse essere oggettivo rispetto a parametri prefissati dall'azienda. Nel dettaglio, se dicessi che una linea di produzione perde in media 20 prodotti a lotto, non sarebbe possibile capire la validità del dato se non in relazione al numero complessivo di prodotti per lotto: 20 su 50 è un dato inaccettabile, 20 su 10.000 può essere un valore più che positivo. Il dato oggettivo può intervenire nello stabilire una percentuale di unità perse oltre la quale la linea di produzione deve essere, se non smantellata, quanto meno ridiscussa. Non potendoci occupare di questo aspetto, il nostro obiettivo è stato quello di standardizzare il più possibile i valori proposti, affinché l'analisi successiva possa essere realizzata in modo obiettivo.

Tenendo presenti questi aspetti, abbiamo deciso di realizzare una funzionalità che permetta di analizzare l'efficienza di un'intera linea di produzione. Come è stato infatti affermato più volte, sebbene le sequenze possibili per realizzare un prodotto siano molte, capire quale possa essere la più efficiente è un problema piuttosto complesso. Infatti, richiede un'elaborazione avanzata dei dati, mettendo insieme le capacità del singolo operatore con i criteri richiesti dall'azienda, senza dimenticare l'impossibilità di fare, nella stessa stazione, operazioni che coinvolgono facce

differenti. Usando la nostra funzionalità, si avranno a disposizione due indici, che permetteranno di sapere non tanto **dove** intervenire ma **se** intervenire. Entrambi gli indici sono espressi con un numero reale appartenente a $[0,1]$, secondo la convenzione per cui più si è vicini all'1, più la sequenza di operazioni risulta buona.

Indice per l'analisi delle unità perse

All'interno della base di dati, vi sono alcune informazioni che possono essere recuperate mettendo insieme più dati (e che quindi, sebbene importanti, sono 'nascoste'), dall'altra quelle che si presentano evidenti e pronte per essere utilizzate. Le unità perse rientrano in questa seconda categoria. Consapevoli che uno studio delle unità perse risulta essere piuttosto scontato nell'ambito dell'area produzione, riteniamo che ciò non tolga la loro centralità. Unità perse significa infatti tempo aggiuntivo, produzioni di lotti ritardate e consegne di ordini pendenti altrettanto posticipati (con conseguente insoddisfazioni dei clienti).

Un'obiezione immediata è che, certo, le unità perse possono essere centrali, ma non aggiungono niente se non confrontate al tempo che ciascuna stazione ha per eseguire le proprie operazioni. Facciamo un esempio:

Abbiamo due sequenze di produzioni per il prodotto A. Entrambe producono lotti di 1000 prodotti. La prima non perde nessuna unità, ma impiega 4 giorni. La seconda (per motivi che non ci è dato sapere) perde un'unità ogni due, ma alla fine realizza il lotto in 3 giorni.

Verrebbe spontaneo sostenere che la prima sia di gran lunga peggiore. Allora perché non abbiamo incluso, in qualche modo, anche il tempo di produzione? La risposta, in via indiretta, l'abbiamo già data, e si trova all'interno dell'entità (o tabella), 'Lotto'. Infatti, si trova sia una data di produzione prevista che una effettiva. La prima è calcolata sulla base del tempo associato alla linea di produzione, quindi si può considerare come del tutto estranea dalla relazione con altre sequenze di produzione per lo stesso prodotto. In nome di ciò, la prima linea dell'esempio produce tutto nei tempi previsti, mentre la seconda presumibilmente avrà un giorno (se non anche due) di ritardo coi tempi. Risultato: clienti che inseriscono recensioni negative.

Detto questo, passiamo alla definizione dell'indice, dando una serie di definizioni. Chiamiamo:

$$L_1, L_2, \dots, L_n$$

I lotti prodotti tramite la sequenza che stiamo momentaneamente analizzando. Per ciascuno di questi, definiamo

$$L_i^{quantità}$$

Come il numero di unità prodotte all'interno del dato lotto. Sia

$$L_i^{perse}$$

Come il numero di unità perse per ciascun lotto (dato che si ottiene senza difficoltà dalla relazione di lotto con unità perse). Definiamo quindi ρ come

$$\rho = \frac{1}{n} * \sum_{i=1}^n \frac{L_i^{perse}}{L_i^{quantità}}$$

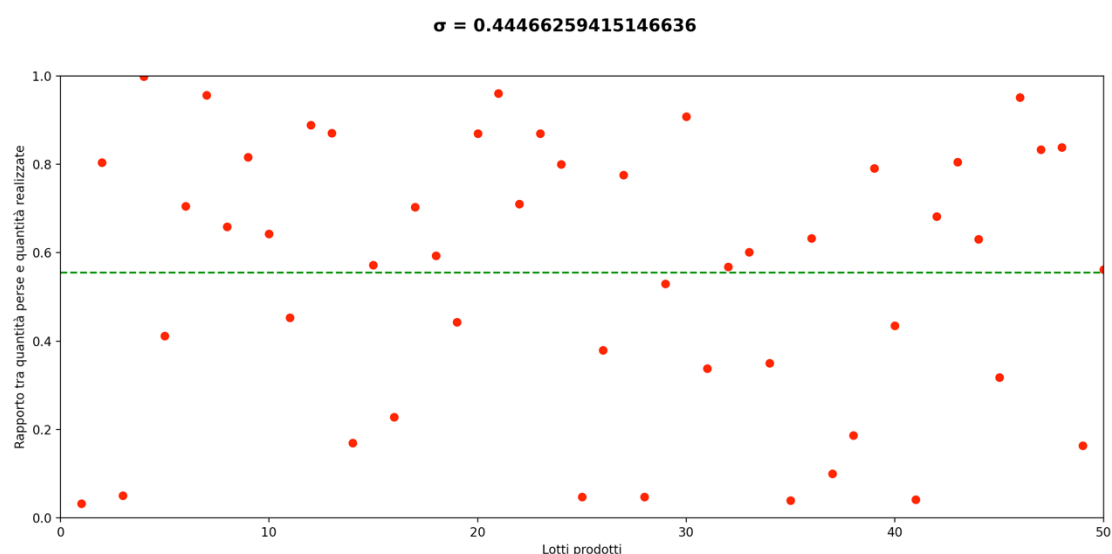
Infine, il nostro indice σ come

$$\sigma = 1 - \rho$$

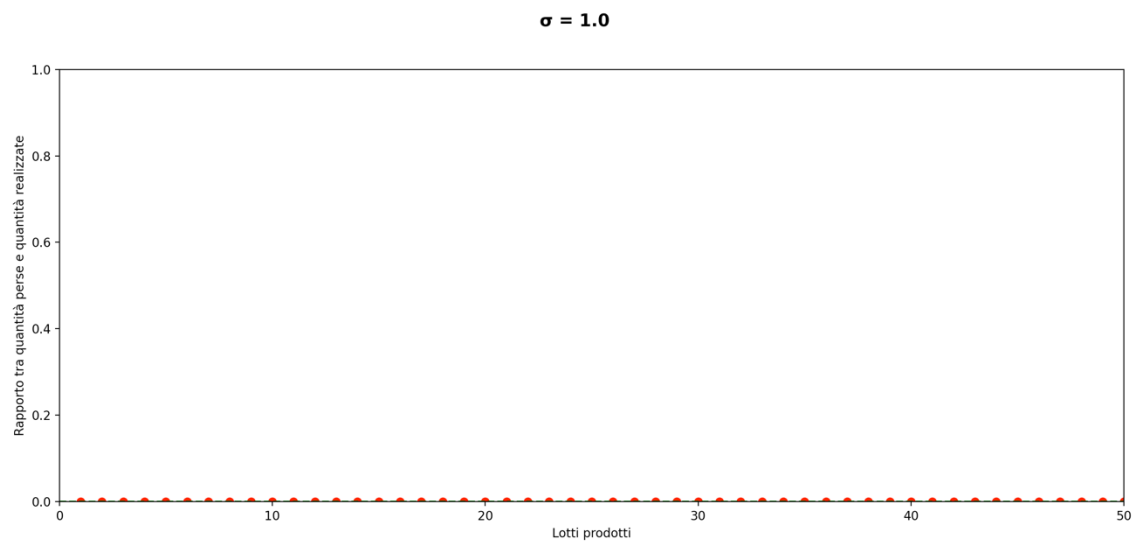
Tralasciando il formalismo, si tratta di calcolare, per ciascun lotto, il rapporto tra unità perse e unità prodotte, e poi fare la media. Questo metodo permette di non dare troppo valore al caso fortunato di un lotto realizzato senza perdite, così come a quello in cui il malumore generale ha portato a giorni e giorni di ritardo nella produzione. Inoltre, come affermato sopra, l'indice sarà tanto più alto quanto è basso il valore medio di unità perse per lotto.

Inseriamo alcuni grafici che mostrano, in casi reali e nei casi limite, come questo indice permetta di avere dei dati utili (tutti i grafici sono stati realizzati con la libreria *matplotlib* e *python*).

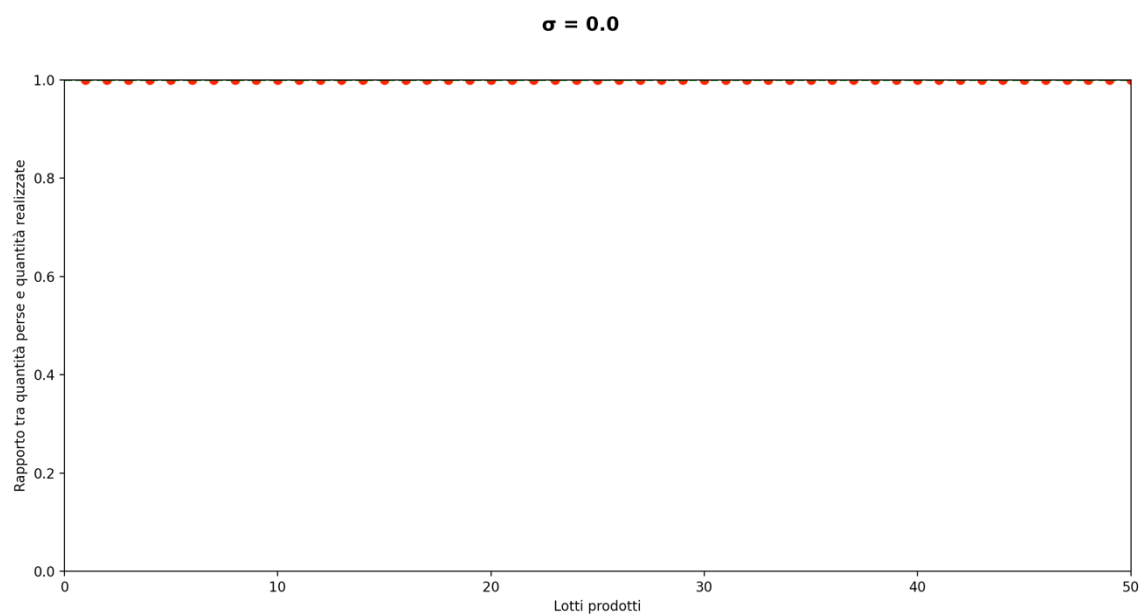
1. Distribuzione casuale di unità perse sulla quantità di prodotti per lotto



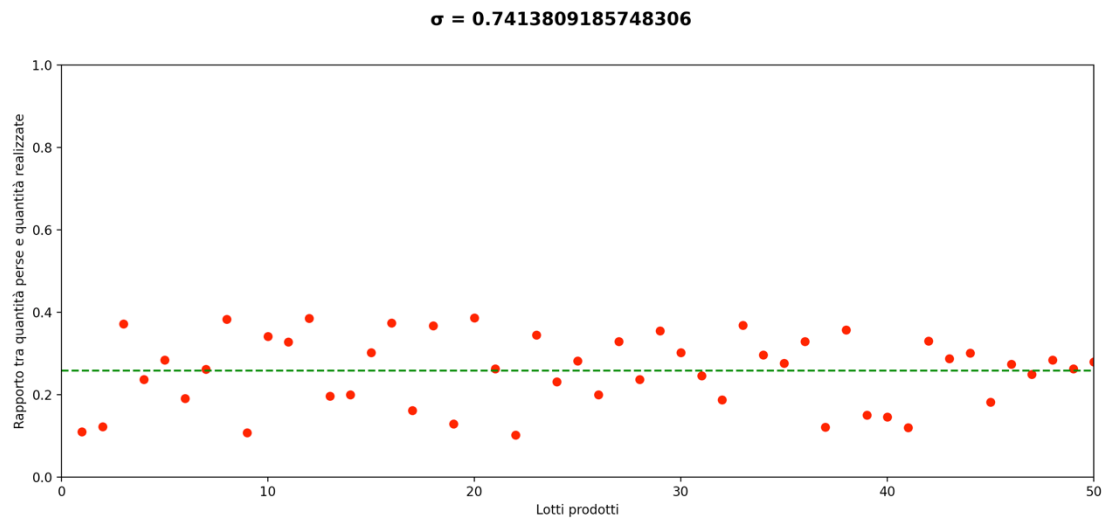
2. Nessuna unità persa in ogni produzione con tale sequenza



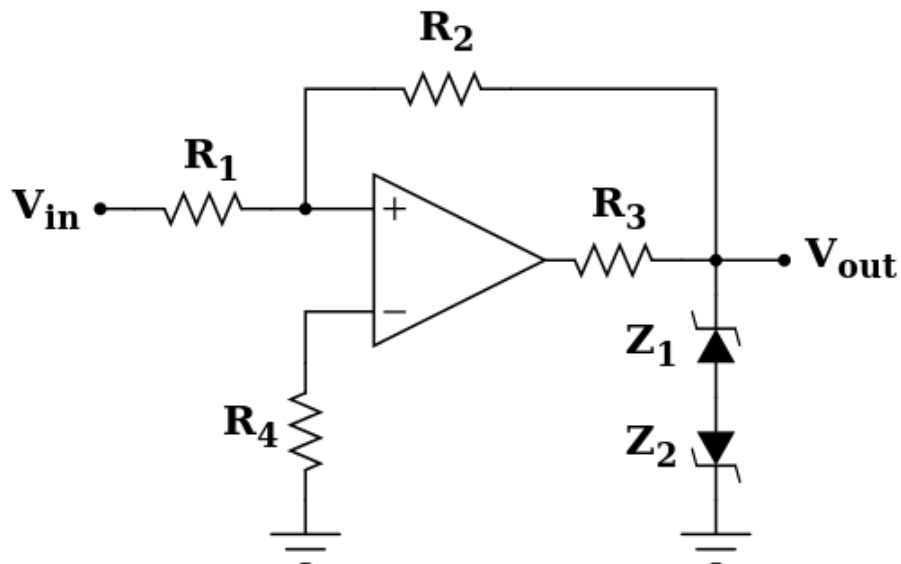
3. Tutte le unità sono perse in ogni utilizzo di tale sequenza



4. Distribuzione realistica: in media per ogni lotto si perde tra il 10% e il 30% di unità



Indice per l'analisi della distribuzione di operazioni per stazione



Quello in figura è uno Schmitt trigger. Esso è un particolare tipo di comparatore di soglia con isteresi, ovvero un circuito che consente di trasformare un segnale analogico in un'uscita che varia soltanto tra due valori di tensione a seconda che l'ingresso superi una certa soglia o sia inferiore a una seconda soglia (più bassa).

Ovviamente questo non ha nulla a che fare con il nostro indice di efficienza, ma lo abbiamo utilizzato per pensare ad un modo con cui potesse essere possibile dire se un valore è buono o meno sulla base del suo discostamento da un altro. L'idea è quella per cui, considerando un plot, si disegna una fascia attorno ad un certo valore, e si contano i punti che stanno al suo interno.

In particolare, il secondo indice che vogliamo proporre è un modo per sapere se le operazioni sono accumulate su certe stazioni (e quindi se sono distribuite in modo abbastanza uniforme), lasciando scoperte altre. Infatti, il rispetto degli indici di efficienza, in termini, ad esempio, di minimizzazione delle rotazioni, potrebbero portare ad accumulare le operazioni in certi punti, rendendo impossibile, per l'operatore a cui viene affidata tale stazione, portare a termine il lavoro. Ovviamente, non si può dire a priori che un eccessivo numero di operazioni corrisponde ad una minore efficienza: se le operazioni consistono solo nell'avvitare una vite, si possono avere molte più di queste concertate nel tempo assegnato a ciascuna stazione. Allo stesso modo, l'inserimento di un motore in una lavatrice è un'operazione delicata che potrebbe compromettere l'intero prodotto se eseguita in modo scorretto, e richiede quindi più tempo. Ugualmente, potrebbe essere che una stazione abbia una sola operazione per un fatto di rispetto dei vincoli. Ciononostante, siamo convinti che questi siano casi limite che, seppur presenti, non vanno eccessivamente ad impattare nella produzione di un prodotto che può richiedere anche migliaia di operazioni, ove necessario. Una corretta omogeneità della distribuzione delle operazioni porta anche ad un adeguato sforzo degli operatori: se qualcuno deve fare molte più operazioni di altri, alla lunga risulterà stanco, cosa che comprometterà la sua efficienza.

Il nostro indice quindi, basandosi sul numero di operazioni per ciascuna stazione, permette di capire se è necessario effettuare degli interventi alla linea di produzione, eventualmente aggiungendo o togliendo stazioni. L'idea è questa: detta M il numero medio di operazioni per stazione, si dirà corretta una stazione se, il numero di operazioni che eseguono varia tra $M+\varepsilon$ e $M-\varepsilon$, estremi inclusi. Il valore ε è stato frutto di una lunga discussione. All'inizio, eravamo subito approdati allo scarto quadratico medio. Credevamo che l'uso di questo dato rendesse l'indice dinamico sulla base dei valori effettivamente presenti. Tuttavia, quello specifico dato era eccessivamente legato alla distribuzione di punti attorno alla media che non faceva per noi: situazioni in cui sembrava si dovesse avere valore 1 ci restituivano 0 e viceversa. Per tale motivo, abbiamo optato in un secondo momento per qualcosa di statico: $\varepsilon = 1$. Il problema stavolta era quasi l'opposto: non avendo un riferimento ai dati, poteva essere a seconda del contesto eccessivamente piccolo. La conclusione è stata quella di basarsi sul valore della media stessa, e dire $\varepsilon = \frac{M}{5}$. Di fatto, ci si può scostare al più di uno in positivo o in negativo dal valore medio. L'indice è dato dal rapporto tra il numero di stazioni corrette e il numero complessivo di stazioni. Formalmente, chiamiamo

$$S_1, S_2, \dots, S_n$$

Le stazioni per la sequenza di produzione che stiamo considerando, e

$$O_1, O_2, \dots, O_n$$

Il numero di operazioni per ciascuna stazione. Sia quindi

$$M = \frac{1}{n} * \sum_{i=1}^n O_i$$

La media delle operazioni assegnate a ciascuna stazione. Chiamiamo τ come

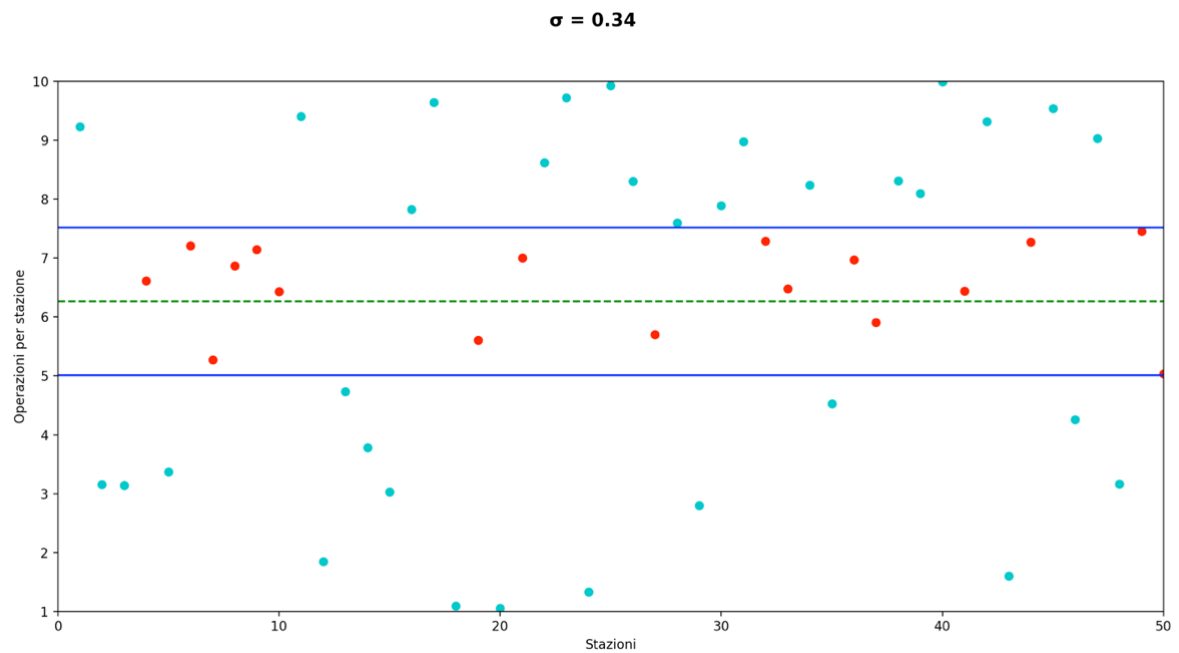
$$\tau = \#\{O_i \mid M + \varepsilon \geq O_i \geq M - \varepsilon\}$$

Ovvero la cardinalità dell'insieme delle stazioni che hanno un numero di operazioni che appartiene all'intervallo $[M - \varepsilon, M + \varepsilon]$. Con questo, il nostro indice σ definito come

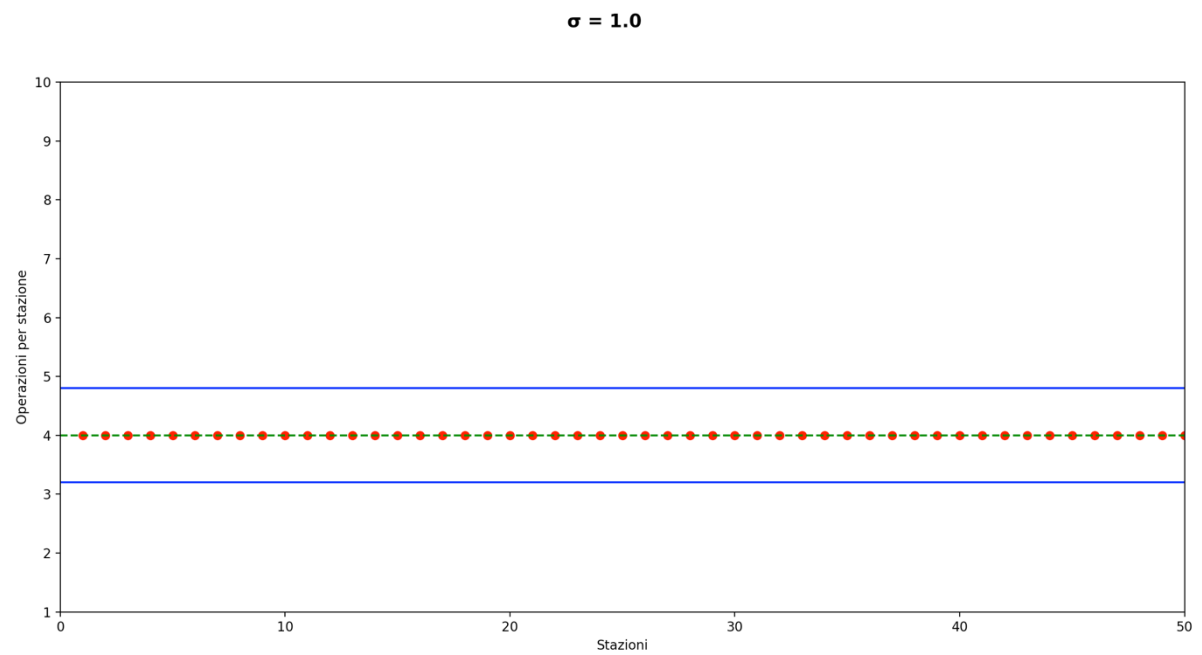
$$\sigma = \frac{\tau}{n}$$

In questo modo, se tutte le stazioni sono corrette (cosa improbabile) esso varrà 1. Se tutte le stazioni sono non-corrette, varrà 0, secondo lo standard che abbiamo già esposto in termini di interpretazione di tale indice. Come prima, inseriamo alcuni grafici che consentano di avere un'interpretazione visiva del dato

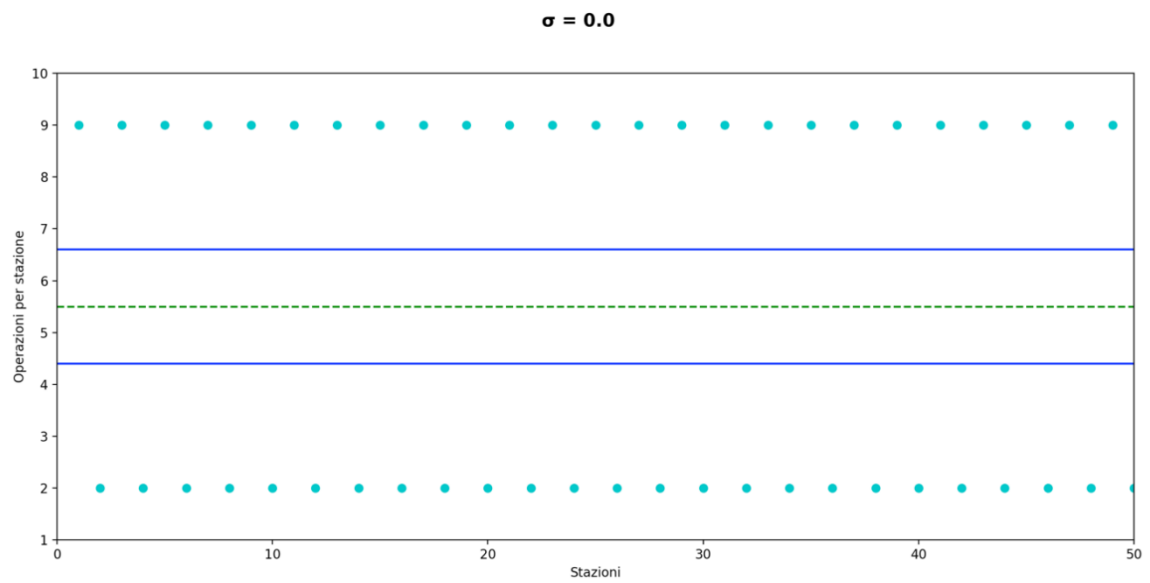
1. Distribuzione casuale di operazioni in un intervallo [1,10] su 50 stazioni



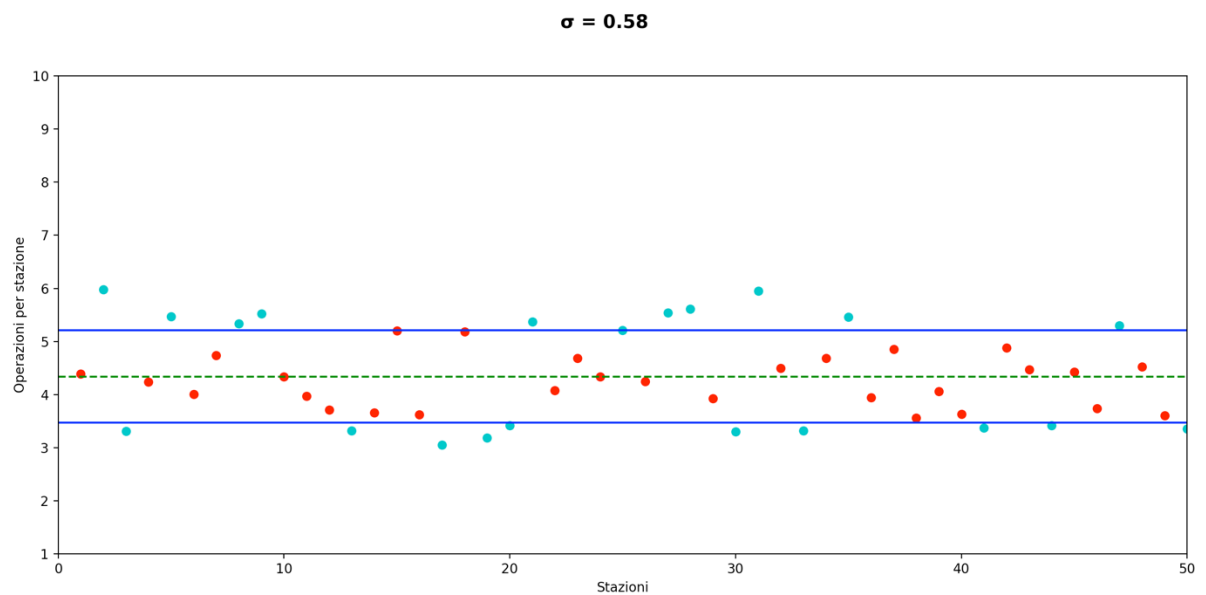
2. Tutte le stazioni hanno lo stesso numero di operazioni



3. Alcune stazioni hanno 9 operazioni, altre solo 2



4. Caso realistico: ogni stazione ha un numero di operazioni tra 3 e 6



10. API

Il database così strutturato oltre a fornire l'accesso ai raw data tramite interrogazioni formulate in SQL espone le seguenti API sotto forma di *stored procedure*:

IncassoAzienda

Descrizione: Calcola l'incasso dell'azienda in un determinato periodo.

Input:

- *primo parametro*: data di inizio del periodo
- *secondo parametro*: data di fine del periodo

Output:

- *terzo parametro*: incasso del periodo richiesto

DataInizio	DataFine	IncassoComplessivo
------------	----------	--------------------

RicercaRimedi

Descrizione: Cerca i rimedi relativi ad un codice errore di un determinato prodotto.

Input:

- *primo parametro*: CodiceErrore
- *secondo parametro*: IDProdotto

Output:

CodiceRimedio	Descrizione
---------------	-------------

AggiuntaLotto

Descrizione: Aggiunge le informazioni di un lotto alla base di dati.

Input:

- *primo parametro*: Codice seriale della variante di prodotto del lotto
- *secondo parametro*: Quantità di elementi nel lotto
- *terzo parametro*: Data prevista di fine produzione
- *quarto parametro*: Sede di produzione
- *quinto parametro*: ID Sequenza scelta per la produzione | NULL se il lotto è ricondizionato

- *sesto parametro*: ID Magazzino scelto per lo stoccaggio

Output:

Nessuno

FineProduzioneLotto

Descrizione: Aggiorna le informazioni di un lotto segnandolo come concluso ed aggiorna lo stato degli ordini pendenti che dipendono da unità di questa variante di prodotto.

Input:

- *primo parametro*: CodiceLotto

Output:

Nessuno

TrackSpedizione

Descrizione: Fornisce tutti gli Hub per i quali una spedizione è passata corredata di giorno ed ora.

Input:

- *primo parametro*: CodiceTracking

Output:

IDHub	Citta	Data	Ora
-------	-------	------	-----

GetGaranzie

Descrizione: Fornisce le date di scadenza delle garanzie sull'intero prodotto (TIPO=0), per un guasto (TIPO=1) o per una componente (TIPO=3) come temporary table.

Input:

- *primo parametro*: IDUnità

Output:

- *temporary table* ScadenzaGaranzie:

ID	TIPO	Descrizione	DataScadenza
----	------	-------------	--------------

AggiuntaDataArrivoOrdine

Descrizione: Assegna la data corrente alla data di arrivo di un ordine parti.

Input:

- *primo parametro:* CodiceOrdineParti

Output:

Nessuno

ChiudiAssistenzaFisica

Descrizione: Chiude una assistenza fisica creando un record per la ricevuta fiscale.

Input:

- *primo parametro:* Codice assistenza fisica
- *secondo parametro:* metodo di pagamento
- *terzo parametro:* in garanzia (True o False)

Output:

Nessuno

GetMemoAssistenzeFisiche

Descrizione: Fornisce gli ID delle assistenze per le quali è necessario fissare un intervento finale.

Input:

Nessuno

Output:

CodiceAssistenza

MostraGiudizi

Descrizione: Fornisce la media dei giudizi di una certa variante di prodotto.

Input:

- *primo parametro:* Codice seriale

Output:

CodiceSeriale	IDGiudizio	Descrizione	MediaVoti
---------------	------------	-------------	-----------

ResiPerVariante

Descrizione: Fornisce il numero di resi per ogni variante e motivazione di reso.

Input:

Nessuno

Output:

CodiceSeriale	CodiceMotivazione	Nome Motivazione	Quantità
---------------	-------------------	------------------	----------

VerificaRicondizionamento

Descrizione: Fornisce i codici delle varianti per i quali può partire una operazione di ricondizionamento.

Input:

Nessuno

Output:

CodiceSeriale	Sufficienza
---------------	-------------

DisponibilitaFasceOrarie

Descrizione: Fornisce le fasce orarie libere di un dato giorno.

Input:

- *primo parametro:* GiornataTarget

Output:

NomeFasciaOraria

OnDemandRefreshReportPendenti

Descrizione: Esegue l'aggiornamento della tabella sui report pendenti.

Input: nessuno

Output: nessuno

ReportMiglioriVendite

Descrizione: Fornisce la classifica delle varianti di prodotto e le quantità vendute per i 7 giorni precedenti.

Input:

Nessuno

Output:

CodiceSeriale	QuantiVenduti
---------------	---------------

GenerazioneSequenzaValida

Descrizione: Fornisce una sequenza di assemblaggio valida

Input:

- *primo parametro:* IDProdotto

Output:

IDOperazione	Nome	Priorità	Parte A	Parte B	IDGiunzione	IDFaccia
--------------	------	----------	---------	---------	-------------	----------

AssegnaInterventi

Descrizione: Assegna ad ogni intervento di un dato giorno il tecnico che lo deve eseguire

Input:

- *primo parametro:* Giorno target

Output:

Nessuno

CBR_retrieve

Descrizione: Fornisce i rimedi e lo score della base di conoscenza ai quali sono collegati partendo da una serie di sintomi

Input:

- *temporary table*

IDSintomo

Output:

IDBase	Score	Descrizione	CodiceRimedio
--------	-------	-------------	---------------

CBR_retain

Descrizione: Inserisce la base di conoscenza fornita in input se non ne esistono altre che sono simili per almeno il 50%

Input:

- *temporary table*

ID Sintomo

- *temporary table*

Codice Rimedio

Output:

Inserimento Avvenuto

Non è stato necessario inserire la base di conoscenza

ValutaSequenza

Descrizione: Fornisce il valore degli indici applicati su una sequenza di produzione

Input:

- *primo parametro:* IDSequenza

Output:

- *result set:*

Descrizione Indice	Score
--------------------	-------

Per aggiungere nuovi indici sui quali valutare la sequenza è necessario aggiungere una tupla con (Descrizione, NomeFunzione) nella tabella "*IndiciAnalitici*".