



UNIVERSITÀ DI PISA

**RELAZIONE DEL PROGETTO
DI BASI DI DATI**

Daniele Giaquinta

Anno Accademico 2019/2020

Indice

GLOSSARIO	4
AREA PRODUZIONE	4
AREA VENDITA.....	5
AREA ASSISTENZA.....	6
AREA SMONTAGGIO	7
DESCRIZIONE DEL DIAGRAMMA ER	8
AREA PRODUZIONE	8
AREA VENDITA.....	13
AREA ASSISTENZA.....	16
AREA SMONTAGGIO	18
RISTRUTTURAZIONE DEL DIAGRAMMA ER	20
ELIMINAZIONE DELLE GENERALIZZAZIONI	20
RIDONDANZE	21
ELIMINAZIONE ATTRIBUTI COMPOSTI, MULTIVALEORE	21
PARTIZIONAMENTO/ACCORPAMENTO DI ENTITA' E RELAZIONI	22
TAVOLA DEI VOLUMI.....	22
AREA PRODUZIONE	22
AREA VENDITA.....	24
AREA ASSISTENZA.....	26
AREA SMONTAGGIO	28
OPERAZIONI E AGGIUNTA RIDONDANZE	29
LISTA OPERAZIONI	29
ANALISI OPERAZIONI	29
MODELLO LOGICO.....	39
TABELLA DEI VINCOLI DI INTEGRITA' REFERENZIALE.....	41
NORMALIZZAZIONE BCNF.....	43
VINCOLI GENERICI	50
DATA ANALYTICS – Diagnosi Intelligente dei Guasti CBR.....	51
1) FASE DI RETRIEVE	51
2) FASE DI REUSE	53
3) FASE DI REVISE.....	57
4) FASE DI RETAIN	59

GLOSSARIO

Per prima cosa sarà mostrato il Glossario dei termini, da consultare in caso ci fosse ambiguità con delle parole presenti nelle spiegazioni a seguire o nel Diagramma Entità-Relazioni

AREA PRODUZIONE

<i>Termine</i>	Descrizione	Sinonimi	Collegamenti
<i>Prodotto</i>	Prodotto concettuale, distinto dall'unità fisica che viene venduta	Apparecchiatura, Dispositivo	Parte, Lotto, Faccia, Magazzino, Cliente, Ordine, Recensione, Garanzia, Reso, Guasto, Prodotto Ricondizionato, Ricondizionamento
<i>Unità</i>	Prodotto fisico, realmente esistente e dotato di codice seriale	Unità fisica	Parte, Lotto, Faccia, Magazzino, Cliente, Ordine, Recensione, Garanzia, Reso, Guasto, Prodotto Ricondizionato, Ricondizionamento
<i>Parte</i>	Una qualche parte che compone un Prodotto	Componenti	Prodotto, Materiale, Operazione, Guasto, Preventivo, Ordine di Parti di Ricambio, Ricevuta Fiscale, Test
<i>Materiale</i>	Materiale/i di cui una Parte è composta		Parte
<i>Operazione</i>	Step che porta avanti l'Assemblaggio di un Prodotto montando una parte soltanto		Parte, Operatore, Sequenza, Faccia, Stazione, Tempo
<i>Sequenza</i>	Insieme di operazioni che monta un prodotto dall'inizio alla fine		Operazione, Stazione
<i>Lotto</i>	Un certo numero di unità fisiche dello stesso prodotto. Generalizzazione totale ed esclusiva che comprende Lotto Prodotto e Lotto di prodotti ricondizionati		Prodotto
<i>Stazione</i>	Postazione in cui viene gestita una parte della Sequenza		Operatore, Sequenza, Operazione
<i>Operatore</i>	Un Operatore lavora	Lavoratore	Operazione, Stazione

	in una Stazione ed esegue un certo numero di operazioni		
<i>Tempo</i>	Tempo che ogni Stazione ha a disposizione per eseguire le operazioni ad esse assegnate		Stazione, Operazione
<i>Faccia</i>	La Faccia su cui poggia un Prodotto in una data Stazione. Ogni prodotto ha un certo numero di facce	Orientazione	Prodotto, Operazione
<i>Magazzino</i>	Nei Magazzini vengono stoccati Lotti di Prodotti e lotti ricondizionati e sono suddivisi sulla base delle classi di Prodotti che stoccano		Prodotto, Lotto, Ricondizionamento, Unità

AREA VENDITA

<i>Termine</i>	Descrizione	Sinonimi	Collegamenti
<i>Cliente</i>	Persona che usa il Market-Place online per acquistare Prodotti	Utente	Account, Prodotto, Ordine, Recensione, Garanzia, Reso
<i>Account</i>	Lista di informazioni del Cliente		Cliente
<i>Ordine</i>	L'ordine contiene la lista dei Prodotti acquistati dal Cliente		Cliente, Prodotto, Spedizione, Garanzia, Reso, Guasto
<i>Spedizione</i>	Processo di Spedizione di un Ordine suddivisa in diversi stati		Ordine, Hub
<i>Hub</i>	Tappe intermedie che effettua un Ordine durante la Spedizione	Centro di smistamento	Spedizione
<i>Recensione</i>	Valutazione facoltativa del Cliente che può assegnare punteggi su una scala prefissata in determinate categorie		Prodotto, Cliente

<i>Garanzia</i>	Periodo entro il quale il Cliente è tutelato in caso di Guasti prematuri di un Prodotto		Prodotto, Cliente, Ordine
<i>Reso</i>	La procedura che un Cliente effettua entro un numero di giorni prestabilito dall'azienda in caso di malfunzionamento o insoddisfazione		Prodotto, Cliente, Ordine, Unità, Ricondizionamento

AREA ASSISTENZA

<i>Termine</i>	Descrizione	Sinonimi	Collegamenti
<i>Guasto</i>	Insieme di informazioni per l'identificazione del Malfunzionamento	Malfunzionamento	Prodotto, Parte, Ordine, Guasto
<i>Assistenza Fisica</i>	Insieme di interventi fisici per la riparazione di un prodotto		Guasto, Unità
<i>Intervento</i>	Generalizzazione totale ed esclusiva che si divide in Intervento a domicilio e presso un centro di assistenza		Assistenza Fisica
<i>Assistenza Virtuale</i>	L'Assistenza Virtuale è composta da una piattaforma che tenta di risolvere il problema dell'Utente tramite un meccanismo di Trouble-Shooting		Guasto, Rimedio, Unità
<i>Rimedio</i>	Soluzione suggerita dal sistema virtuale che applica l'Utente stesso per cercare di riparare autonomamente un malfunzionamento		Assistenza Virtuale
<i>Assistenza Fisica</i>	Qualora l'Assistenza Virtuale non sia		Tecnico, Centro d'Assistenza

	sufficiente è necessario ricorre a quella Fisica		
<i>Tecnico</i>	Dipendenti qualificati alla riparazione dei Prodotti		Assistenza Fisica, Preventivo, Centro d'Assistenza
<i>Centro d'Assistenza</i>	Centro in cui possono essere effettuati interventi di individuazione di Guasti o riparazioni dai Tecnici		Assistenza Fisica, Tecnico
<i>Preventivo</i>	Preventivo effettuato dal Tecnico al Cliente con le informazioni sul costo e sulle Parti necessarie		Tecnico, Parte, Ordine di Parti di Ricambio, Ricevuta Fiscale
<i>Ordine di Parti di Ricambio</i>	Ordine del Tecnico delle Parti necessarie alla riparazione		Parte, Preventivo
<i>Ricevuta Fiscale</i>	Ricevuta con codice univoco che contiene le informazioni della transazione		Parte, Preventivo

AREA SMONTAGGIO

<i>Termine</i>	Descrizione	Sinonimi	Collegamenti
<i>Test</i>	Verifica del funzionamento di una o più Parti		Parte, Controllo Generale, Guasto, Ricondizionamento
<i>Controllo Generale</i>	Sequenza di Test di un Prodotto	Test Tree	Test, Ricondizionamento
<i>Ricondizionamento</i>	Processo di ricondizionamento di un Prodotto reso da un Cliente		Prodotto, Reso, Test, Controllo Generale, Prodotto Ricondizionato
<i>Prodotto Ricondizionato</i>	Prodotto sottoposto al processo di ricondizionamento		Lotto Ricondizionato, Prodotto, Ricondizionamento
<i>Lotto Ricondizionato</i>	Lotto di Prodotti Ricondizionati		Prodotto Ricondizionato

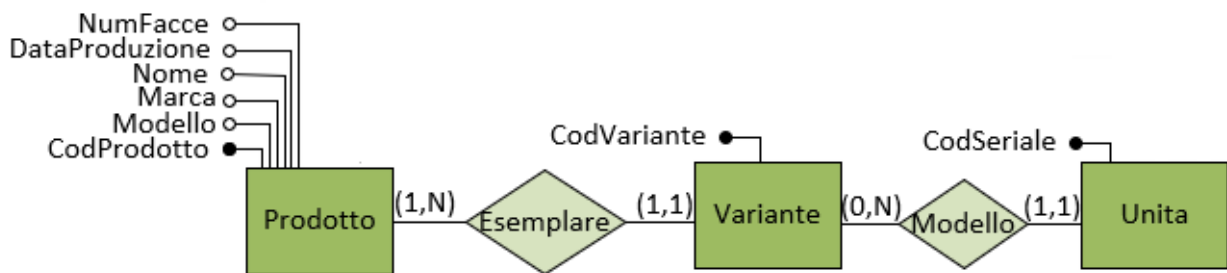
DESCRIZIONE DEL DIAGRAMMA ER

Le prossime pagine si occuperanno di descrivere brevemente il Diagramma Entità-Relazioni per spiegarne alcune scelte fatte e il significato di alcuni elementi presenti.

! NOTA: le cardinalità in corsivo sono cardinalità di una relazione, mentre quelle non in corsivo sono cardinalità di una relazione ma solo dalla parte dell'entità che sta venendo affrontata

AREA PRODUZIONE

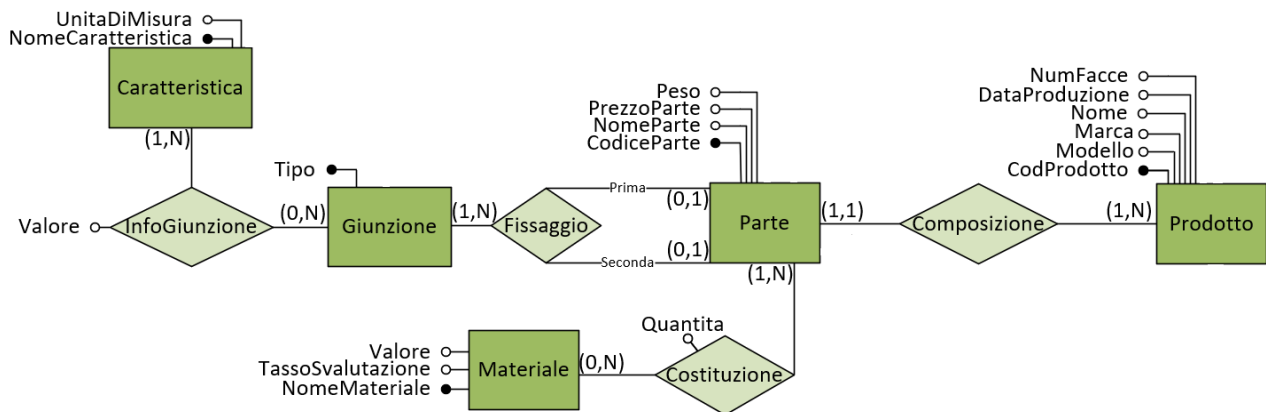
-Vediamo come sono strutturate nel diagramma le basiche informazioni sul prodotto, ovvero i suoi attributi e la gestione delle sue varianti e della sua composizione in diverse parti:



Il prodotto in questo diagramma è sostanzialmente un concetto; ogni prodotto è identificato da un codice generico che identifica quel determinato prodotto, ad esempio un iPhone 13 PRO avrà un codice che lo identifica (e ovviamente l'insieme di attributi richiesti) ma potrà avere diverse varianti, da questo deriva la relazione. Ogni variante ha un proprio codice che la identifica. Riprendendo l'esempio, se avessimo un iPhone 13 con 8gb di RAM e un modello che invece ne ha 12, questi due avrebbero lo stesso Codice Prodotto ma un diverso Codice Variante, che risulterebbe diverso anche dai codici varianti degli altri prodotti. In Variante è inoltre memorizzato il prezzo del prodotto in quella determinata variante (qua non si vede).

È inoltre importante che ogni prodotto fisico, cioè realmente esistente e non solo concettuale, abbia un proprio codice seriale per essere identificato univocamente, caratteristica utile in innumerevoli contesti.

-Vediamo adesso la parte del diagramma che riguarda la composizione di un prodotto;



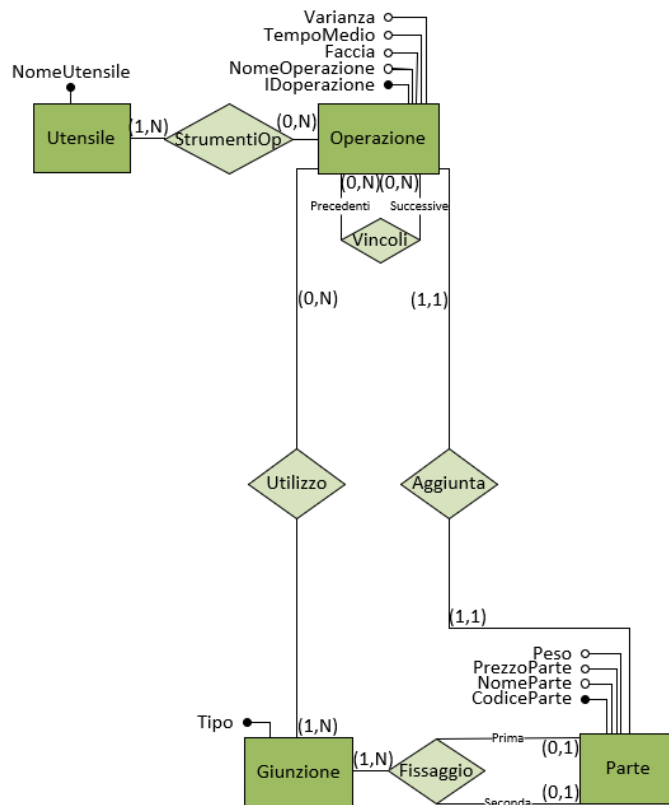
si nota che il prodotto è costituito di un certo numero di parti (1,N) ma una parte è univocamente identificata tramite il Codice Parte e appartiene ad un solo prodotto (1,1).

Ogni parte è costituita da un certo numero di materiali (1,N) e in una certa quantità, che è quindi un attributo della relazione tra Materiale e Parte.

Le parti sono fissate ad altre parti con uno o più elementi di giunzione (1,N), che giustifica la relazione ternaria tra Parte e Giunzione in cui Parte partecipa due volte. I due (0,1) dal lato di Parte sono giustificati dal fatto che la prima parte ad essere montata non avrà una parte precedente, e l'ultima non ne avrà una successiva.

Nel DataBase sono inoltre salvate una serie di caratteristiche delle giunzioni in modo generico, quindi per sapere per ogni giunzione quali sono le sue caratteristiche e il valore numerico serve una relazione (N,N) tra Caratteristica e Giunzione.

-Ora vediamo come sono strutturate nel diagramma le operazioni che aggiungono una parte al prodotto parziale in fase di costruzione:



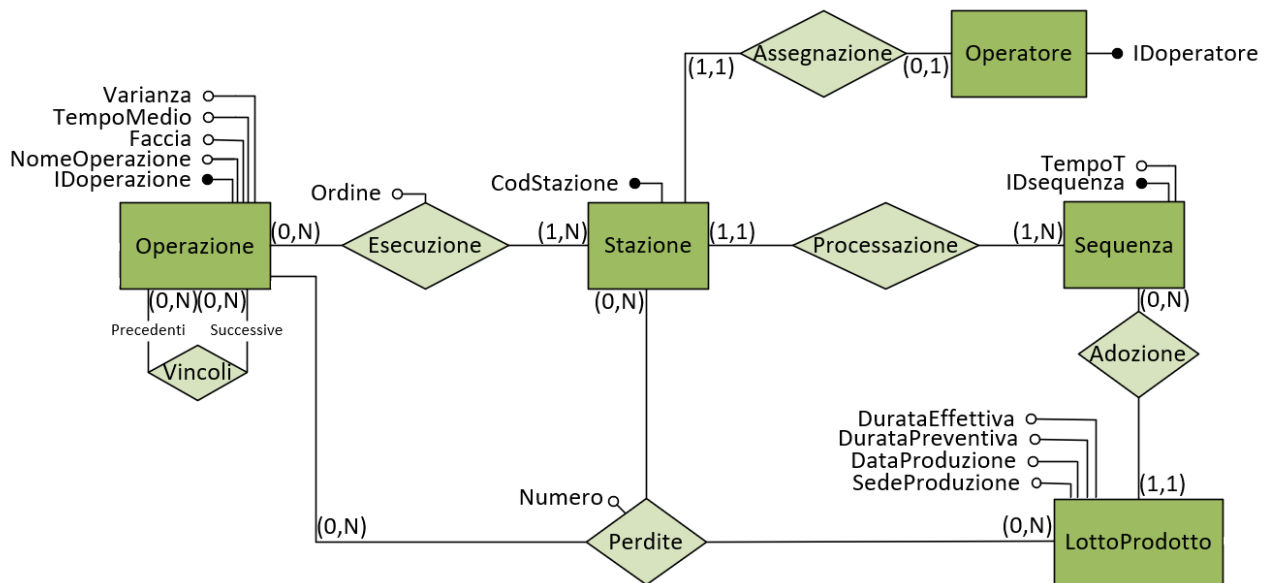
Poiché, come abbiamo visto le parti sono univocamente identificate e appartengono ad un solo prodotto, esiste per ogni parte una sola operazione che si occupa di montarla e ogni operazione si occuperà sempre di montare quella specifica parte.

Dal frammento di diagramma si nota che in un'operazione si impiegano un certo numero di utensili, e un certo numero di elementi di giunzione, quindi Operazione avrà due relazioni (N,N) rispettivamente con Giunzione e Utensile.

Inoltre Operazione ha una self-relation che serve a gestire i vincoli di precedenza tecnologica, che sinteticamente si traduce nella necessità di aver completato un certo numero di operazioni prima di una qualsiasi operazione presa in considerazione. Le due cardinalità sono

(0,N) perché la prima operazione di una catena di operazioni non ne ha nessuna precedente e l'ultima non avrà operazioni successive.

-Guardiamo adesso come è gestita dunque la produzione vera e propria di un lotto:



NOTA: in questo segmento di diagramma non è presente il Codice del Lotto né la sua relazione con l'unità fisica poiché LottoProdotto è figlio di una generalizzazione che sarà poi motivata nell'Area Smontaggio. In ogni caso questi due elementi sono presenti nel padre della generalizzazione.

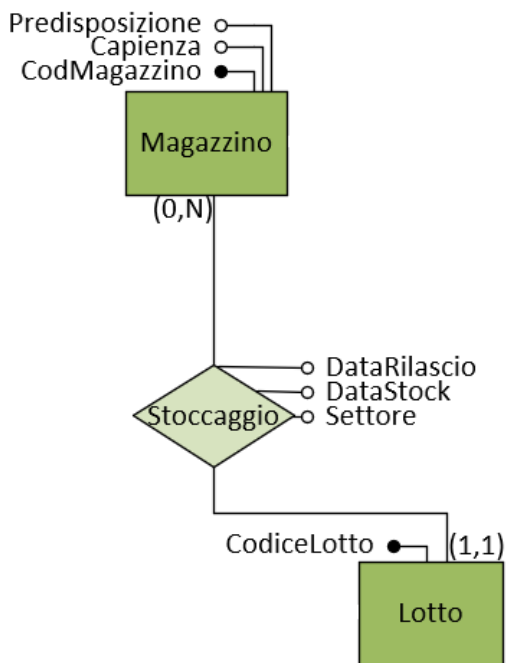
Si nota dal segmento di diagramma che ogni lotto, che è identificato da un codice, è prodotto integralmente servendosi della stessa sequenza per ogni prodotto di cui è composto, ma questo non implica che per due lotti diversi di prodotti uguali venga usata la stessa sequenza.

La divisione in stazioni delle operazioni si nota con la relazione (N,N). È importante notare che il Tempo T è un attributo della sequenza e non della stazione, perché è uguale per tutte le stazioni che stanno lavorando per realizzare la stessa sequenza, e si ricava facilmente in Stazione, visto che ha una relazione con Sequenza che dal lato di Stazione è (1,1). Si noti inoltre che il tempo medio in cui viene svolta una operazione da parte degli operatori e la varianza del tempo di esecuzione sono salvati in Operazione, così si hanno queste informazioni per ogni operazione.

Brevemente per quanto riguarda gli operatori è importante notare che non si tiene conto dello storico dei dati, e se un operatore dovesse essere riassegnato ad una differente stazione si dovrebbe modificare il DataBase; (0,1) dal lato di Operatore contempla eventuali sospensioni, ferie o malattie degli operatori, poiché in questi casi un operatore non è più assegnato a nessuna stazione nel DataBase.

Per realizzare il concetto dei prodotti che vengono perduti durante la produzione è necessaria una relazione ternaria tra LottoProdotto, Stazione e Operazione, così sappiamo per quale operazione è stato perduto, in quale stazione e a quale lotto sarebbe appartenuto.

-Per quanto riguarda lo stoccaggio dei lotti la struttura è abbastanza immediata:



Ogni lotto (che qui è solo il padre della generalizzazione a cui è stato fatto riferimento in precedenza, qui nascosta, motivo per cui il suo unico attributo è la chiave Codice Lotto) è stoccato in uno e un solo magazzino (1,1) e ogni magazzino può essere vuoto o contenere N lotti (0,N); si ha quindi una relazione (1,N).

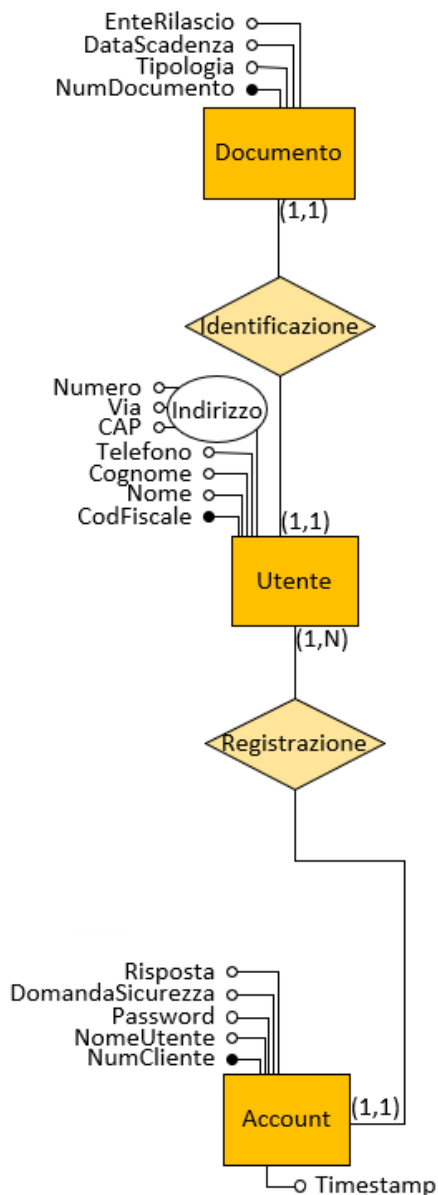
Ogni magazzino è identificato da un codice. È necessario aggiungere sottoforma di attributi di relazione la data di rilascio di un certo lotto (se questa è avvenuta), la data di stoccaggio e il settore in cui un dato lotto è conservato (si divide ogni magazzino in settori composti da una lettera e un numero per risalire alla posizione di un lotto).

Si noti che Lotto è in relazione (1,1) dalla parte di Lotto con Prodotto (anche se qua non si vede),

poiché un lotto è fatto da prodotti dello stesso tipo, ed è in relazione (1,N) dalla parte di Lotto con l'unità fisica (anche questo non è qui visibile).

AREA VENDITA

-Per prima cosa nell'Area Vendita è necessario curare la parte per la gestione degli Account; distinguo il concetto di utente e account poiché un utente, benché sia qualcuno che si è registrato sul sito creando un account, potrebbe aver creato più di un account:

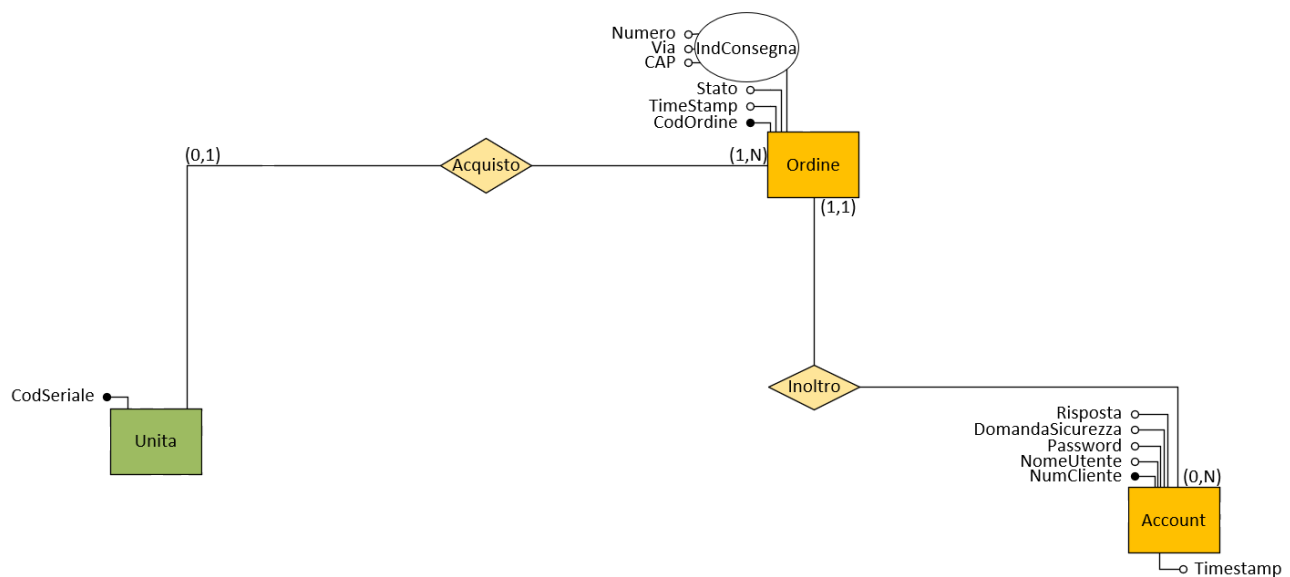


L'Account ha come chiave il Numero Cliente, fondamentale anche per gli acquisti effettuati; per la distinzione appena fatta si ha quindi che un account è creato e appartiene ad una sola persona (Utente) che è quindi caratterizzata dal proprio Codice Fiscale.

Si noti che in Utente l'attributo Indirizzo risulta essere un attributo composto in quanto si possono distinguere come concetti separati il CAP, la Via e il Numero Civico.

È inoltre richiesto che l'Utente fornisca un documento in corso di validità e per questo viene usata un'altra entità anche se la relazione è (1,1) poiché il concetto di documento è indipendente da quello di utente.

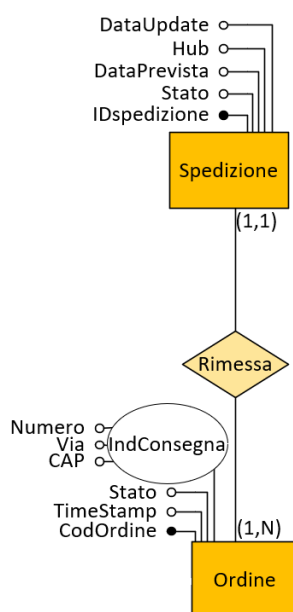
-Per quanto riguarda la gestione degli ordini si ha il seguente schema:



Un Ordine è effettuato ovviamente da un solo Account, quindi il Codice dell'Ordine identifica gli stessi univocamente.

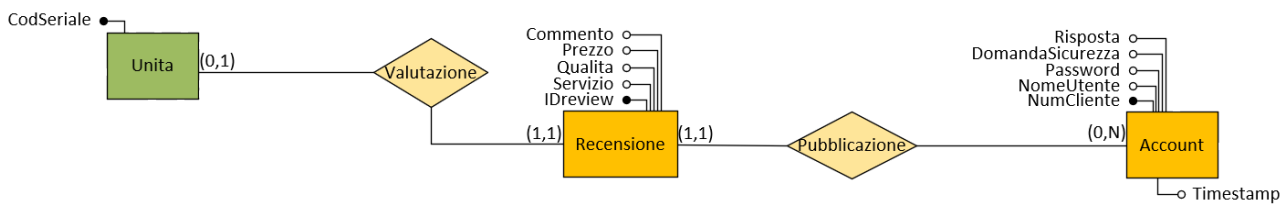
Ordine è in relazione con l'Unità Fisica, poiché è quella che di fatti si ordina. E' da tenere conto che anche se non si vede nel frammento di schema riportato sopra, da Unità si riesce a ricavare "gratuitamente" sia l'informazione riguardante il Prodotto che rappresenta, sia l'informazione riguardante quale Variante del Prodotto è (segue un percorso con cardinalità (1,1) dal suo lato). Questo fa sì che la relazione Ordine-Unità abbia senso visto che un cliente ordina un Prodotto in una certa sua Variante e non oggetti a lui ignoti tramite il loro Codice Seriale.

-Per gestire la spedizione è sufficiente una relazione tra Ordine e l'entità Spedizione:



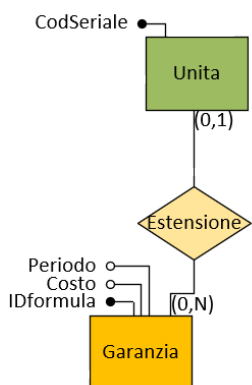
Ogni spedizione è univocamente identificata attraverso un codice. Si noti che dalla parte di Ordine la cardinalità è (1,N); questo perché ad esempio, in caso di resi del prodotto per difetti di fabbrica, se un cliente dovesse scegliere di farsi spedire una nuova unità del prodotto sarà necessario avviare una nuova spedizione tracciabile e relativa allo stesso ordine di partenza.

-Per quanto riguarda la gestione delle recensioni:



Un Account può lasciare recensioni su articoli che ha acquistato. Si noti che è possibile lasciare recensioni solo sugli articoli che un utente ha effettivamente acquistato con l'account attualmente utilizzato; infatti Recensione è in relazione con l'Unità fisica, non con il prodotto; ricordandoci da quanto visto sopra che anche Ordine è in relazione con l'Unità fisica e notando che partendo da recensione è possibile fare un percorso in cui la cardinalità massima è 1 che porta fino a Ordine si può dunque facilmente verificare questa proprietà del diagramma.

-Per la Garanzia è necessaria una relazione sia col Prodotto (concettuale) che con l'Unità Fisica (nel frammento è riportata solo la relazione con l'Unità fisica):

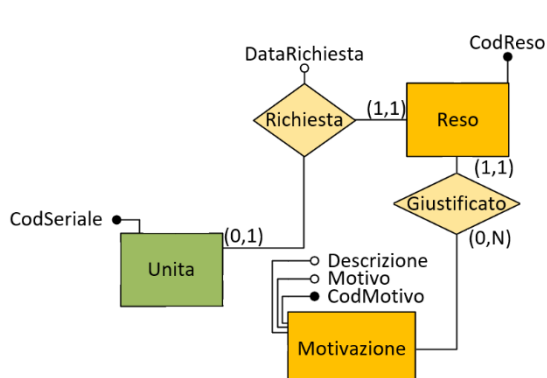


La relazione col Prodotto serve per avere un'idea per ogni prodotto di quali siano le sue formule di estensione di garanzia esistenti e selezionabili.

In questo modello viene fatta l'ipotesi che la garanzia che è valida di standard su tutti i prodotti non sia memorizzata; essendo sempre valida è inutile tenerne conto, la sua scadenza sarà due anni dopo la data di acquisto.

Quindi la relazione con l'Unità fisica serve solo a dirci a quali prodotti è stata associata dal cliente una estensione della garanzia o una formula modificata.

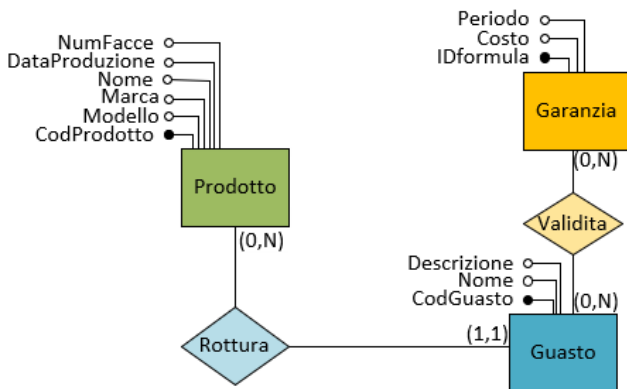
-I Resi vengono anch'essi gestiti con una relazione diretta con l'Unità fisica:



L'entità Reso è in relazione con l'Unità fisica poiché da quest'ultima è immediato raggiungere l'Ordine e vale anche il contrario.

AREA ASSISTENZA

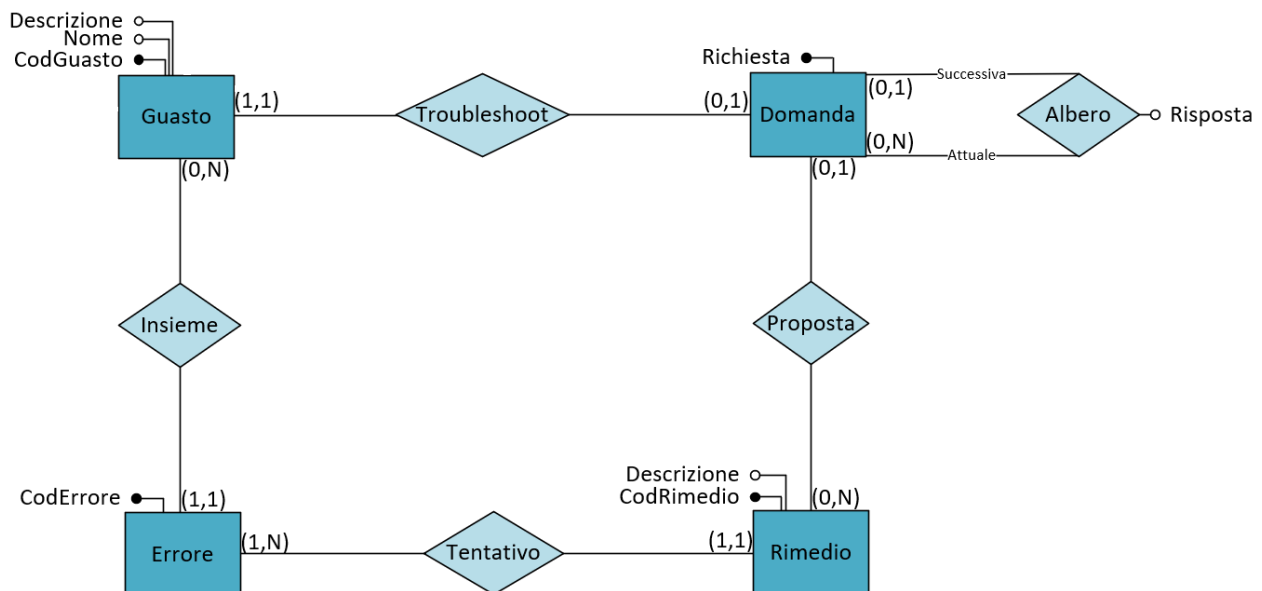
-Vediamo la parte centrale dell'area Assistenza; il Guasto:



Ogni guasto avrà un Codice che lo identifica univocamente, quindi un dato guasto appartiene a uno ed un solo prodotto.

La relazione tra Garanzia e Guasto serve invece a dare informazioni su eventuali tipi di guasto coperte dalle varie estensioni o formule personalizzate di garanzia.

-Analizziamo adesso tutta la parte del diagramma relativa all'Assistenza Virtuale:



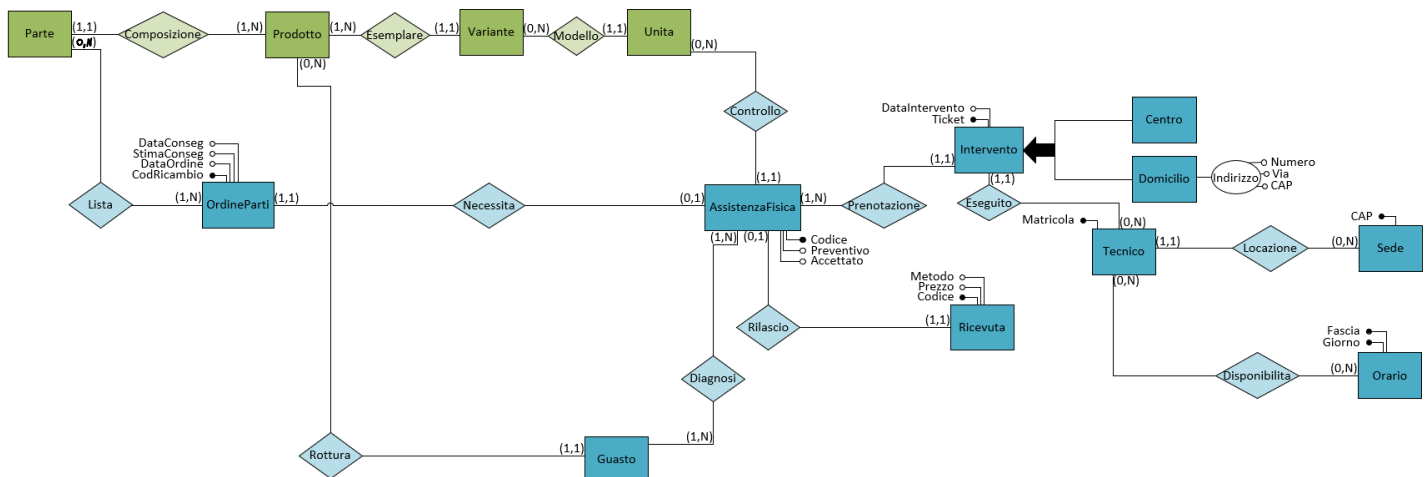
Partendo da Guasto, come da indicazione si nota che ad ogni guasto possono corrispondere dei Codici di Errore (0,N), inoltre viene indicato che i Codici di Errore siano relativi ad uno ed un solo guasto (1,1).

Quando si verifica un guasto il sistema online pone domande all'utente. L'idea è quella di avere per ogni guasto un albero di domande; la relazione tra Guasto e Domanda è quindi (1,1) e la domanda che corrisponde ad un certo guasto è la prima domanda dell'albero, la sua radice.

La self-relation in Domanda crea dunque l'albero di domande; in base alla risposta (sì o no) si arriva ad una domanda o ad un'altra. Se invece siamo arrivati ad una foglia, cioè alla fine delle domande, il sistema può proporre una soluzione.

In tal caso si sfrutta la relazione tra Domanda e Rimedio.

-Guardiamo ora l'Assistenza Fisica:



Per curare questa parte della traccia è stato necessario creare un'entità Assistenza Fisica che fosse in relazione con l'entità Intervento; questo perché con Assistenza Fisica si intende l'insieme di interventi che riceve una Unità dall'inizio fino alla fine.

Gli interventi si dividono a loro volta in interventi a domicilio e interventi effettuati nel Centro dell'azienda; da questo deriva una generalizzazione totale.

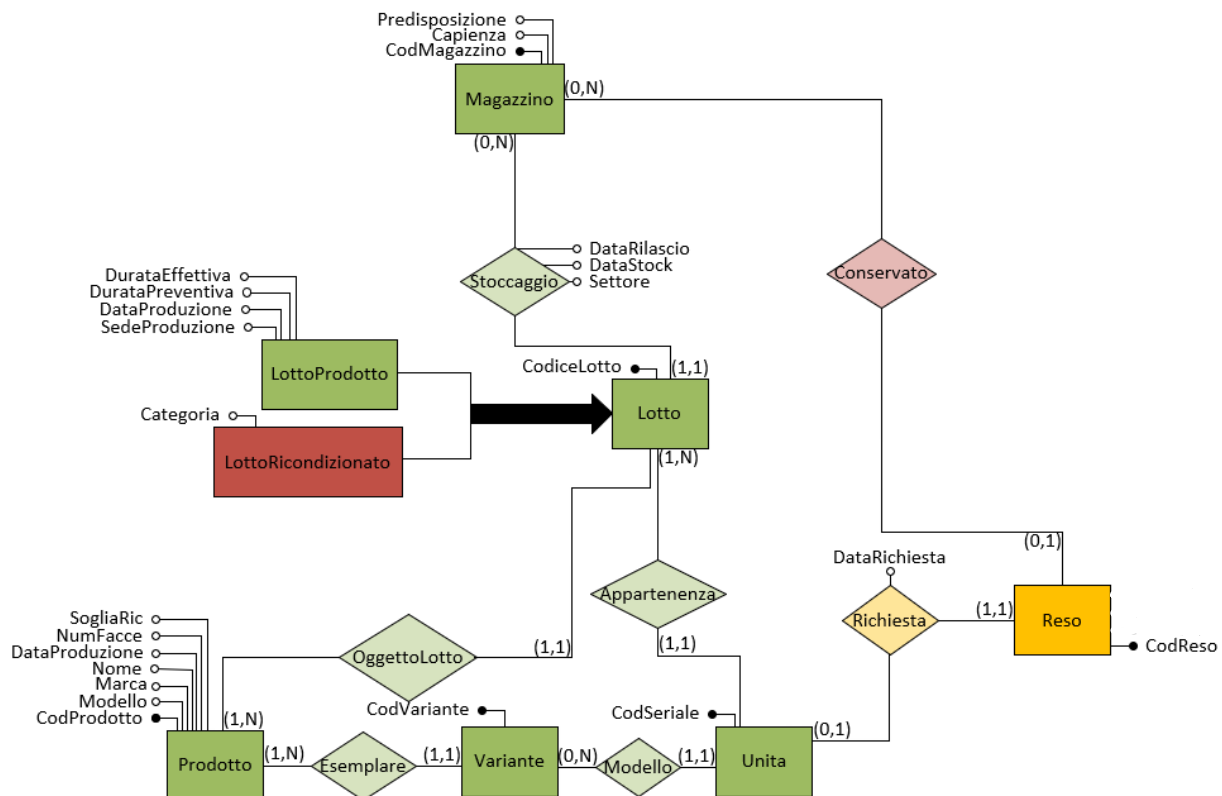
Ogni intervento è eseguito da un tecnico del quale è memorizzato nel DataBase, oltre ai giorni in cui è in servizio e le fasce orarie, anche il CAP della sede presso cui lavora; possono infatti essere sfruttate le vicinanze numeriche dei CAP per far fare ai tecnici meno strada possibile per gli interventi.

Tornando sull'entità Assistenza Fisica ovviamente essa è associata a uno o più guasti, inoltre come da traccia può prevedere un ordine di una o più parti.

Alla totalità degli interventi dell'Assistenza Fisica può essere associata la ricevuta fiscale col prezzo del lavoro (prezzo delle parti più mano d'opera). La cardinalità di questa relazione dalla parte di Assistenza Fisica è tuttavia (0,1) poiché la ricevuta non va rilasciata se la spesa non è a carico del richiedente. Questo è facilmente verificabile nel DataBase; se il prodotto per cui è richiesta assistenza ha la garanzia in corso di validità e la garanzia valida copre i guasti riscontrati, allora la spesa non è a carico del cliente. Tutte queste informazioni sono ricavabili dal DataBase.

AREA SMONTAGGIO

-Analizziamo per prima cosa la parte relativa allo stoccaggio dei prodotti resi:

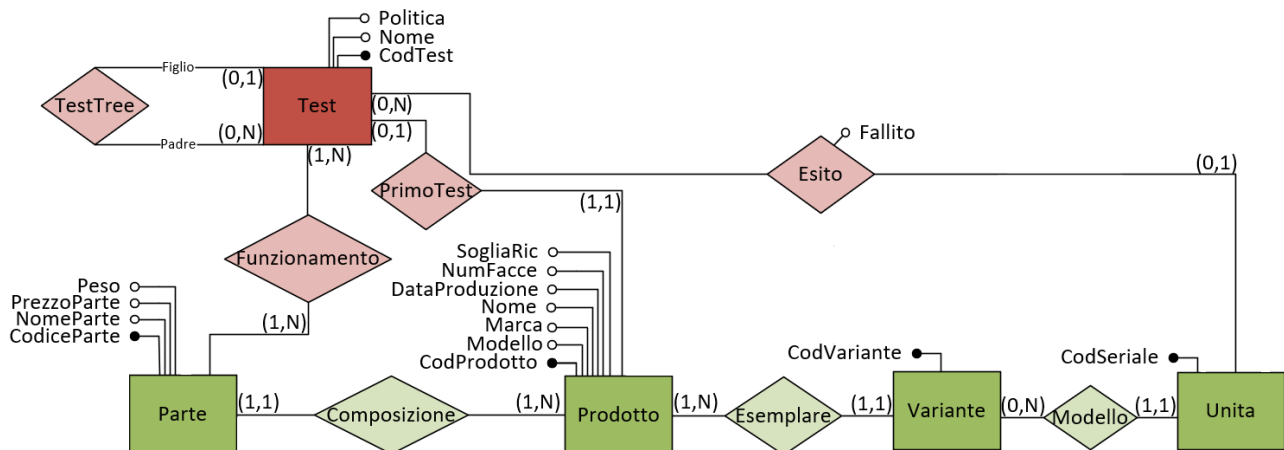


Viene indicato che i prodotti resi vengono stoccati nei magazzini dell'azienda e una volta raggiunta una certa soglia (che varia da prodotto a prodotto, motivo per cui è stato aggiunto un attributo SogliaRic a Prodotto) viene predisposto un lotto per il ricondizionamento.

A questo fine è stata creata una generalizzazione totale per Lotto. Il figlio Lotto Ricondizionato ha solo l'attributo Categoria (e la chiave Codice Lotto dal padre), poiché è sufficiente solo la categoria ad ottenere tutte le informazioni relative al prezzo del prodotto ricondizionato, supposto che l'azienda si sia fissata delle politiche di corrispondenza tra categoria e sconto.

Si noti infine che l'entità Reso non indica precisamente un oggetto reso da un cliente, ma indica la richiesta di reso che il cliente effettua sul portale; comunque questa entità è in relazione con Magazzino poiché dall'entità Reso si ricava facilmente l'unità fisica che è stata resa.

-Infine vediamo la parte relativa ai test e al controllo generale:



Per prima cosa è stato creato il Test-Tree attraverso la self-relation in Test (anche qui la radice non avrà ovviamente padre e le foglie non avranno figli, a questo sono dovuti gli zeri nelle cardinalità).

Il primo test nonché la radice dell'albero per un dato prodotto è dato dalla relazione (1,1) tra Prodotto e Test; ogni prodotto cioè avrà un suo test iniziale.

La relazione di Test con Parte serve ovviamente per memorizzare nel DataBase di quali parti verifica il funzionamento un dato test.

Per quanto riguarda invece lo storico dei controlli sulle vere e proprie unità fisiche, questo è dato dalla relazione tra Unità e Test, che ha un attributo di relazione Fallito. Più nel dettaglio:

nel DataBase viene memorizzato, per una certa unità sottoposta al controllo generale, solo il test che ha avuto esito negativo, a questo è dovuta la cardinalità massima 1 da entrambi i lati.

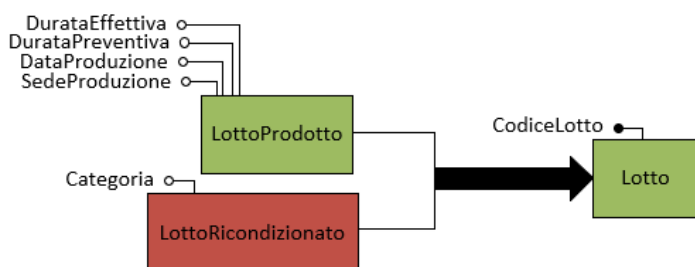
RISTRUTTURAZIONE DEL DIAGRAMMA ER

Affrontiamo rapidamente la ristrutturazione del Diagramma ER

ELIMINAZIONE DELLE GENERALIZZAZIONI

Come prima fase della ristrutturazione del Diagramma Entità-Relazioni è necessario rimuovere le generalizzazioni e trattarle nel modo opportuno.

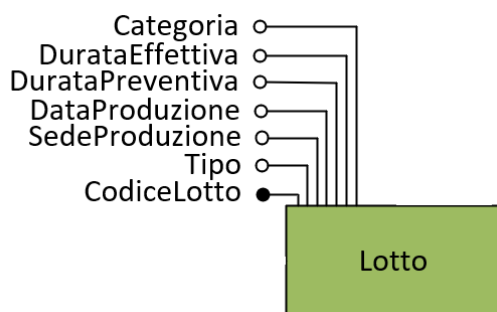
-Nel Diagramma presentato si trovano due generalizzazioni totali, vediamo la prima:



Questa generalizzazione distingue i lotti esistenti in lotti di unità uscite dal processo di produzione, e quindi nuove, a lotti di unità ricondizionate.

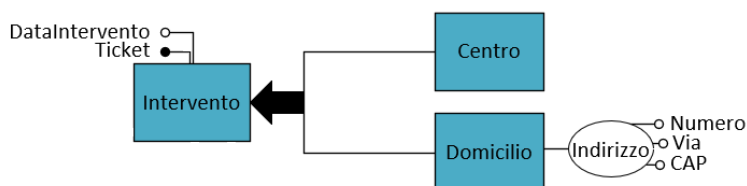
Si nota abbastanza facilmente che accessi al padre e ai figli sono contestuali; infatti ogni volta che accederemo a uno dei due

figli sarà sempre necessario accedere anche al padre per ricavare il Codice Lotto, che è l'attributo chiave. Per questo motivo i due figli verranno accorpati al padre al fine di semplificare inoltre le interrogazioni e le operazioni che riguardano i lotti.



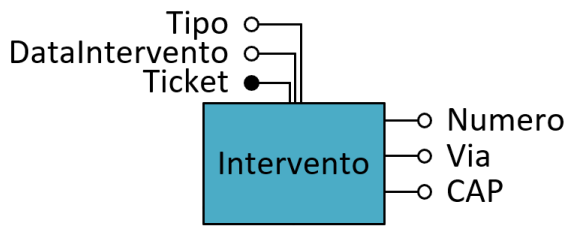
L'entità finale manterrà semplicemente il nome di Lotto e viene inoltre aggiunto l'attributo Tipo, che indicherà se si tratta di un lotto di unità nuove o di un lotto di unità ricondizionate.

-Adesso trattiamo la seconda generalizzazione, anch'essa totale:



Analogamente alla prima generalizzazione, gli accessi ai figli sono contestuali a quelli sul padre, e inoltre tutte le relazioni che riguardano questa entità sono sul padre, poiché tutte le informazioni che queste ci procurano

possono essere richieste sia per un Intervento presso un centro sia per un Intervento presso il domicilio del cliente. Ancora una volta i figli verranno accorpati al padre.



Anche qua viene aggiunto un attributo Tipo.

Si noti che l'attributo composto è stato scomposto in tre attributi separati, come verrà accennato fra poco, tutti gli attributi composti (per altro sempre indirizzi) hanno ricevuto lo stesso trattamento.

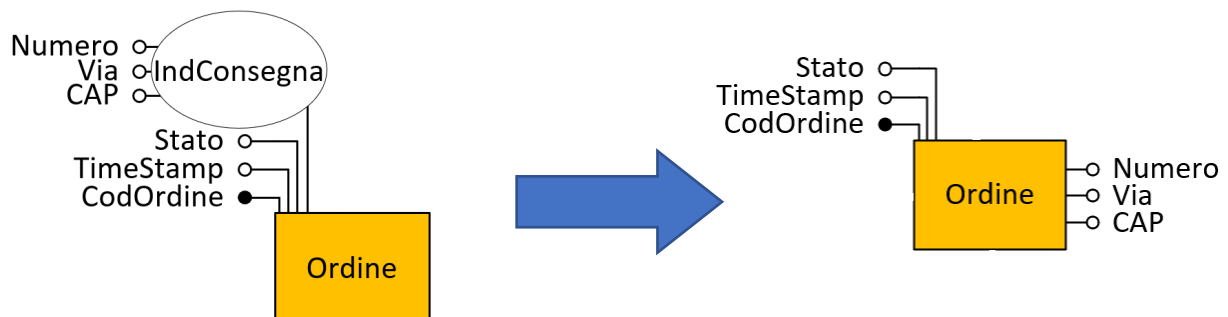
RIDONDANZE

Nel diagramma Entità-Relazioni presentato non sono presenti ridondanze. Verrà valutata l'aggiunta di ridondanze al momento della scelta delle operazioni da implementare nel DataBase.

ELIMINAZIONE ATTRIBUTI COMPOSTI, MULTIVALORE

E' necessario rimuovere gli attributi composti per dargli un significato più definito; dovrebbero essere inoltre trattati eventuali attributi multivalore ma non ne è presente alcuno nel Diagramma, fatta l'assunzione che un utente possa registrarsi inserendo un solo numero di telefono (possibile attributo multivalore).

-Gli attributi composti sono tutti indirizzi, questi vengono semplicemente scomposti in 3 attributi:



PARTIZIONAMENTO/ACCORPAMENTO DI ENTITA' E RELAZIONI

Non è stato ritenuto necessario un partizionamento o un accorpamento di entità e relazioni, in particolare, una zona del diagramma che avrebbe potuto essere soggetta a partizionamenti è la zona riguardante la memorizzazione dei dati di un Account, dell'Utente e del Documento di identità; questa parte però è stata partizionata da subito in tre entità distinte.

TAVOLA DEI VOLUMI

Prima di continuare con la scelta delle operazioni significative da effettuare sui dati, è necessaria una Tavola dei Volumi in cui si stima la quantità di occorrenze che ci sarà nel DataBase a regime per ogni entità e per ogni relazione.

-Segue la Tavola dei Volumi rappresentata con una tabella di 4 colonne; *NOME*, *TIPO*, *VOLUME*, *MOTIVAZIONE*.

Nota: quando si troverà la dicitura *ipotesi* nella sezione *MOTIVAZIONE*, vorrà dire che quel volume è ipotizzato essere tale e non deriva da nessun ragionamento particolare.

AREA PRODUZIONE

NOME	TIPO	VOLUME	MOTIVAZIONE
Prodotto	Entità	500	Ipotesi
Variante	Entità	1.000	Si ipotizza che un Prodotto abbia in media 2 varianti
Esemplare	Relazione	1.000	La relazione è (1,1) dal lato di Variante e eredita il suo volume
Unità	Entità	10.000	Ipotesi
Modello	Relazione	10.000	La relazione è (1,1) dal lato di Unità e eredita il suo volume
Parte	Entità	50.000	Si ipotizza che un Prodotto sia composto mediamente da 100 parti
Composizione	Relazione	50.000	La relazione è (1,1) dal lato di Parte e eredita il suo volume
Materiale	Entità	100	Ipotesi
Costituzione	Relazione	150.000	Si ipotizza che una parte sia composta in media da 3 materiali
Giunzione	Entità	1.000	Ipotesi

Fissaggio	Relazione	100.000	Si ipotizza che mediamente una parte sia fissata ad un'altra con 2 giunzioni
Caratteristica	Entità	30	Ipotesi
InfoGiunzione	Relazione	3.000	Si ipotizza che una Giunzione abbia in media 3 caratteristiche
Aggiunta	Relazione	50.000	La relazione è (1,1) dal lato di Parte e eredita il suo volume
Operazione	Entità	50.000	L'entità partecipa alla relazione Aggiunta con (1,1) e eredita il suo volume
Utilizzo	Relazione	100.000	Si ipotizza che mediamente una parte sia fissata ad un'altra con 2 giunzioni
Vincoli	Relazione	1.000.000	Si ipotizza che un'Operazione abbia in media 30 vincoli di precedenza
Utensile	Entità	100	Ipotesi
StrumentiOp	Relazione	100.000	Si ipotizza che un'Operazione adoperi in media 2 Utensili
Stazione	Entità	5.000	Ipotesi
Esecuzione	Relazione	500.000	Si ipotizza che mediamente una Stazione si occupi di 10 Operazioni
Assegnazione	Relazione	5.000	La relazione è (1,1) dal lato di Stazione e eredita il suo volume
Operatore	Entità	6.000	C'è un surplus di Operatori rispetto alle Stazioni per contemplare malattie, ferie o altri imprevisti
Processazione	Relazione	5.000	La relazione è (1,1) dal lato di Stazione e eredita il suo volume

Sequenza	Entità	1.500	Si ipotizza che mediamente un Prodotto possa esser realizzato con 3 diverse Sequenze
Lotto	Entità	200	Si ipotizza che in media un lotto sia composto da 50 Unità
Adozione	Relazione	200	La relazione è (1,1) dal lato di Lotto e eredita il suo volume
Perdite	Relazione	400	Si ipotizza che in media vengano perdute 2 Unità per un Lotto
Stoccaggio	Relazione	200	La relazione è (1,1) dal lato di Lotto e eredita il suo volume
Magazzino	Entità	10	Ipotesi
Appartenenza	Relazione	10.000	La relazione è (1,1) dal lato di Unità e eredita il suo volume
OggettoLotto	Relazione	200	La relazione è (1,1) dal lato di Lotto e eredita il suo volume

AREA VENDITA

NOME	TIPO	VOLUME	MOTIVAZIONE
Account	Entità	1.000	Ipotesi
Registrazione	Relazione	1.000	La relazione è (1,1) dal lato di Account e eredita il suo volume
Utente	Entità	900	Si ipotizza che non siano molte le persone a crearsi più Account
Identificazione	Relazione	900	La relazione è (1,1) dal lato di Utente e eredita il suo volume
Documento	Entità	900	L'entità partecipa alla relazione Identificazione con (1,1) e eredita il suo volume
Ordine	Entità	2.000	Ipotesi

Inoltro	Relazione	2.000	La relazione è (1,1) dal lato di Ordine e eredita il suo volume
Rimessa	Relazione	2.100	Le spedizioni sono poco più degli ordini perché comprendono prodotti malfunzionanti, restituiti e rispediti
Spedizione	Entità	2.100	Le spedizioni sono poco più degli ordini perché comprendono prodotti malfunzionanti, restituiti e rispediti
Acquisto	Relazione	4.000	Si ipotizza che in media in un Ordine vengano acquistati 2 prodotti
Pubblicazione	Relazione	500	Si ipotizza che in media venga recensito un prodotto ogni 6 acquistati
Recensione	Entità	500	Si ipotizza che in media venga recensito un prodotto ogni 6 acquistati
Valutazione	Relazione	500	La relazione è (1,1) dal lato di Recensione e eredita il suo volume
Richiesta	Relazione	200	Si ipotizza che delle 4.000 unità acquistate, 200 vengano rese a prescindere dal motivo
Reso	Entità	200	L'entità partecipa alla relazione Richiesta con (1,1) e eredita il suo volume
Giustificato	Relazione	200	La relazione è (1,1) dal lato di Reso
Motivazione	Entità	20	Ipotesi
Estensione	Relazione	400	Si ipotizza che per 1 prodotto su 10 venga scelta una formula di Estensione di garanzia

Formula	Relazione	1.000	Si ipotizza che in media un Prodotto abbia 2 formule di estensione/alternative di garanzia
Garanzia	Entità	700	Ipotesi
Validità	Relazione	7.000	Si ipotizza che in media una formula di estensione/alternativa di garanzia copra 10 guasti

AREA ASSISTENZA

NOME	TIPO	VOLUME	MOTIVAZIONE
Guasto	Entità	2.000	Ipotesi
Rottura	Relazione	2.000	La relazione è (1,1) dal lato di Guasto e eredita il suo volume
Insieme	Relazione	4.000	Si ipotizza che un guasto presenti in media 2 codici di errore
Errore	Entità	4.000	Si ipotizza che un guasto presenti in media 2 codici di errore
Troubleshoot	Relazione	2.000	Ad ogni Guasto corrisponde una domanda di partenza, la radice dell'albero
Domanda	Entità	10.000	Ipotesi; si ipotizza che un albero abbia in media 5 domande
Albero	Relazione	8.000	Le occorrenze di Albero sono i rami, che sono n-1 nodi, si hanno quindi $2000 \cdot (5-1)$ rami
Proposta	Relazione	3.000	Si ipotizza che in media un albero proponga 1,5 rimedi
Rimedio	Entità	6.000	Ipotesi

Tentativo	Relazione	6.000	La relazione è (1,1) dal lato di Rimedio e eredita il suo volume
AssistenzaFisica	Entità	200	Ipotesi
Diagnosi	Relazione	250	Si ipotizza che in pochi casi l'unità per cui è chiesta assistenza abbia più di un guasto
Controllo	Relazione	200	La relazione è (1,1) dal lato di AssistenzaFisica e eredita il suo volume
Prenotazione	Relazione	400	Si ipotizza che una riparazione richieda in media 2 interventi
Intervento	Entità	400	L'entità partecipa alla relazione Prenotazione con (1,1) e eredita il suo volume
Eseguito	Relazione	400	La relazione è (1,1) dal lato di Intervento e eredita il suo volume
Tecnico	Entità	500	Ipotesi
Locazione	Relazione	500	La relazione è (1,1) dal lato di Tecnico e eredita il suo volume
Sede	Entità	230	Ci sono circa due tecnici per sede sempre attivi, più un extra per coprire ferie, malattie o imprevisti
Disponibilità	Relazione	5000	Si ipotizza che un tecnico lavori 5 giorni a settimana e mattina e pomeriggio (due distinte fasce orarie); si avranno quindi $500 \cdot 5 \cdot 2$ occorrenze
Orario	Entità	10	I giorni vanno dal lunedì al venerdì e ogni giorno ha 2 fasce orarie
Necessità	Relazione	50	Si ipotizza che 1 su 4 Assistenze Fisiche

			necessiti di parti di ricambio
OrdineParti	Entità	50	L'entità partecipa alla relazione Necessità con (1,1) e eredita il suo volume
Lista	Relazione	100	Si ipotizza che un ordine comprenda in media 2 parti

AREA SMONTAGGIO

NOME	TIPO	VOLUME	MOTIVAZIONE
PrimoTest	Relazione	500	Rappresenta la radice dell'albero dei Test per ogni Prodotto, quindi il volume sarà quello di Prodotto
Test	Entità	5.000	Si ipotizza che un albero comprenda in media 10 Test
TestTree	Relazione	4.500	Questa relazione rappresenta i rami dell'Albero, che sono (nodi-1); quindi si ha $500 \cdot (10-1)$
Funzionamento	Relazione	10.000	Supponiamo che in media un Test verifichi il funzionamento di 2 parti
Esito	Relazione	10	Si suppone che solo 10 unità abbiano fallito un test (si salvano solo i test falliti)
Conservato	Relazione	100	Si ipotizza che 100 Prodotti resi siano stati raggruppati in lotti e siano stati nuovamente stoccati nei Magazzini

OPERAZIONI E AGGIUNTA RIDONDANZE

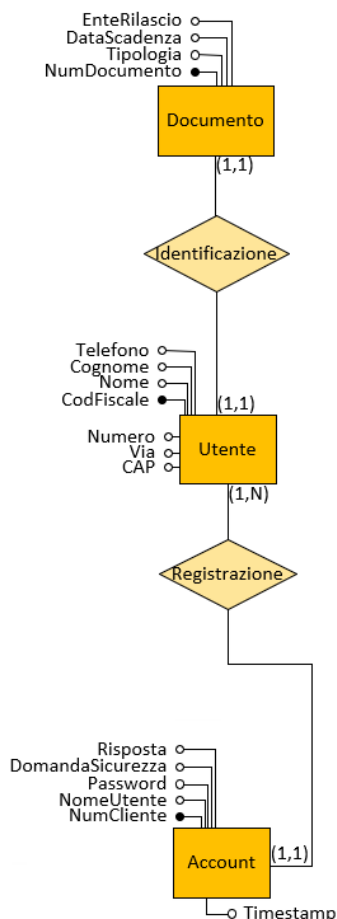
In questa sezione verranno elencate le 8 operazioni che sono state scelte per l'implementazione, e verrà anticipato per quali operazioni è stato scelto di introdurre ridondanze. Più in basso si troverà l'analisi delle operazioni con le relative Tavole Degli Accessi.

LISTA OPERAZIONI

- 1) Creazione di un Account
- 2) Trovare la spesa complessiva di un Ordine (**aggiunta ridondanza "Spesa" in Ordine**)
- 3) Trovare i Prodotti che sono stati più soggetti a riparazioni
- 4) Dato un Codice di Errore, mostrare i possibili Rimedi e il Guasto che indica
- 5) Dato un Prodotto, trovare la media dei voti delle Recensioni per il Prodotto
- 6) Per ogni Prodotto, trovare quale sua Variante è la più venduta
- 7) Stilare una classifica dei prodotti più insoddisfacenti (cioè quelli resi per diritto di recesso)
- 8) Trovare la quantità di Unità vendute per ogni Lotto e dividerle per le Unità totali del Lotto, per analizzare come stanno andando le vendite (**aggiunta ridondanza del numero di unità che compone il lotto in Lotto**)

ANALISI OPERAZIONI

1) Analizziamo la 1° operazione; la creazione di un Account. Per questa operazione sono necessarie 3 scritture; una per ognuna delle 3 entità in cui sono partizionate le informazioni del cliente:



-DATI IN INPUT: NomeUtente, Password, DomandaSicurezza, Risposta, CodFiscale, Nome, Cognome, Telefono, Via, Numero, CAP, NumDocumento, Tipologia, DataScadenza, EnteRilascio

-DATI IN OUTPUT: nessuno

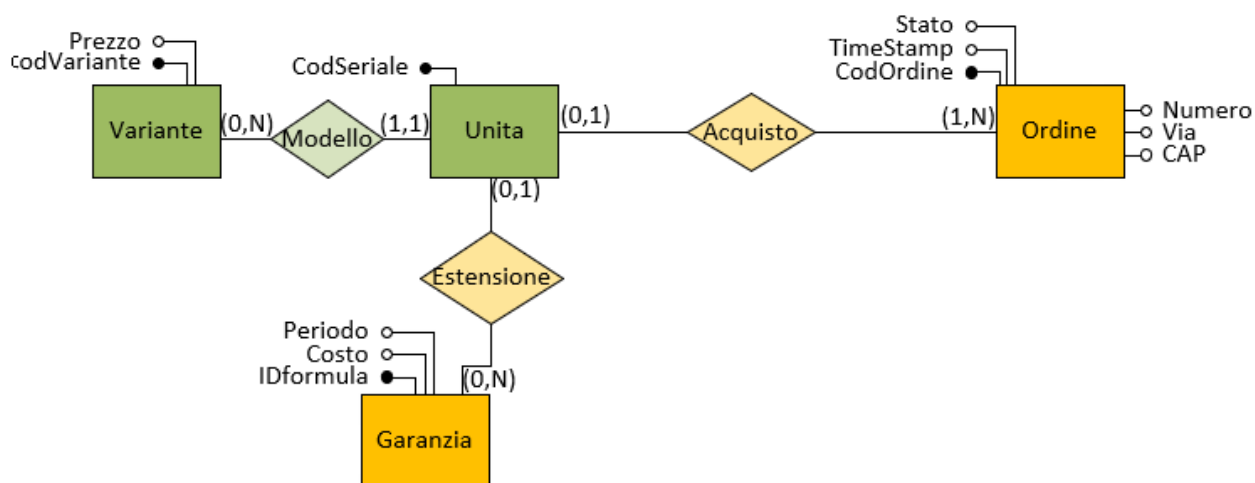
-FREQUENZA SUPPOSTA: 3/gg

NOME	TIPO	VOLUME
Account	Entità	1.000
Registrazione	Relazione	1.000
Utente	Entità	900
Identificazione	Relazione	900
Documento	Entità	900

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Documento	Entità	1	Scrittura	Inserisco il Documento
Account	Entità	1	Scrittura	Inserisco i dati
Registrazione	Relazione	1	Scrittura	Creo il collegamento tra l'Utente e l'Account

Il costo giornaliero dell'operazione sarà $3 \cdot (2+2+2)=16$

2) Analizziamo la seconda operazione, ovvero dato l'ID di un ordine troviamone la spesa complessiva.



Per farlo bisogna passare dalla relazione acquisto, che ci indica dato un ordine quali unità sono state acquistate. Poi da unità si ricava il prezzo dell'unità sulla base della variante che quell'unità rappresenta. Inoltre devo vedere se erano state scelte formule di estensione garanzia e leggerne il costo

-DATI IN INPUT: CodOrdine

-DATI IN OUTPUT: Spesa

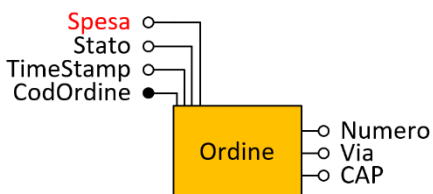
-FREQUENZA SUPPOSTA: 200/gg

NOME	TIPO	VOLUME
Ordine	Entità	2.000
Acquisto	Relazione	3.000
Unità	Entità	10.000
Modello	Relazione	10.000
Variante	Entità	1.000

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Acquisto	Relazione	2	Lettura	Leggo le unità acquistate
Variante	Entità	2	Lettura	Leggo i prezzi dei vari prodotti per sommarli
Estensione	Relazione	2	Lettura	Guardo se le unità hanno formule di estensione garanzia
Garanzia	Entità	2	Lettura	Guardo quando costa la suddetta formula

Questa operazione ha un costo di $200 * (2+2+2+2) = 1600$, che non è ignorabile; questo costo infatti aumenterebbe ancora se aumentassero le richieste di visualizzazione della spesa o il numero medio di articoli acquistati; si pensi che un utente che visualizza i suoi ordini, ne visualizza più di uno contemporaneamente e per ognuno visualizzerà la spesa totale, quindi questo costo potrebbe essere ancora più alto e viene introdotta una ridondanza in Ordine

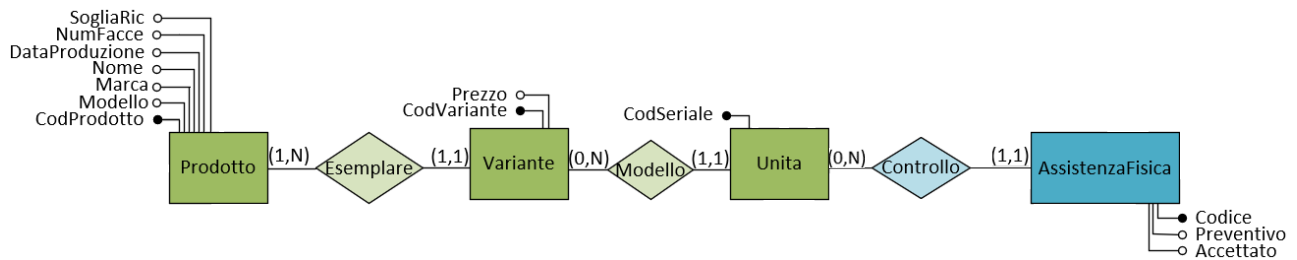
-Vediamo introducendo l'attributo Spesa in Ordine quale sarebbe il costo totale:



CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Ordine	Entità	1	Lettura	Leggo la spesa

In questo caso invece il costo dell'operazione è pari alla frequenza, cioè 200. Si nota che l'operazione precedente deve essere comunque fatta per inserire in Spesa il valore che deve avere, ma questa viene fatta una sola volta al momento della creazione dell'ordine e non dovrà mai essere aggiornata; la ridondanza quindi viene mantenuta.

3) La terza operazione che verrà implementata prevede di trovare i Prodotti che sono stati più soggetti a riparazioni, quelli che possono essere definiti i più difettosi.



Da Controllo vanno ricavate le Unità che sono state soggette a riparazioni; poi si legge da Unità in cui si può ricavare il CodVariante (poiché la relazione è (1,1) dal lato di Unità). Analogamente leggo in Variante e ricavo il CodProdotto, e infine leggo in Prodotto e ricavo il nome del prodotto.

-DATI IN INPUT: nessuno

-DATI IN OUTPUT: Lista dei Prodotti difettosi, a partire dal più difettoso

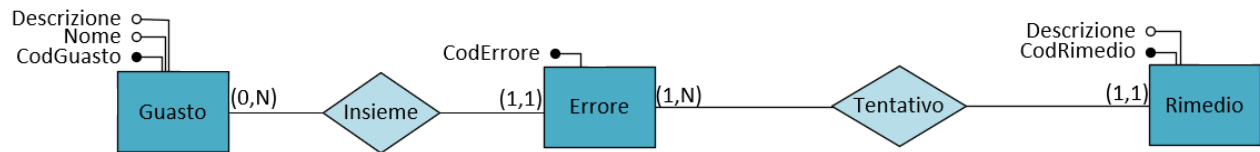
-FREQUENZA SUPPOSTA: 1/mese

NOME	TIPO	VOLUME
Prodotto	Entità	500
Variante	Entità	1.000
Unità	Entità	10.000
AssistenzaFisica	Entità	200
Controllo	Relazione	200

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Controllo	Relazione	200	Lettura	Trovo i codici seriali delle unità riparate
Unità	Entità	200	Lettura	Trovo i codici Varianti delle Unità riparate
Variante	Entità	200	Lettura	Trovo i codici Prodotto delle unità riparate
Prodotto	Entità	200	Lettura	Trovo il nome dei Prodotti

Il costo totale dell'operazione è di 800 (mensile)

4) Per la quarta operazione, dato un codice di errore verrà mostrato il guasto che indica e la descrizione dei possibili rimedi che potrebbero risolvere il problema.



Per prima cosa da Guasto si legge il nome del guasto relativo al codice di errore (la relazione è (1,1) dal lato di Errore), poi va effettuata una lettura alla relazione Tentativo per trovare i codici dei possibili rimedi e infine in Rimedio se ne legge la descrizione.

-DATI IN INPUT: CodErrore

-DATI IN OUTPUT: Nome, Descrizione

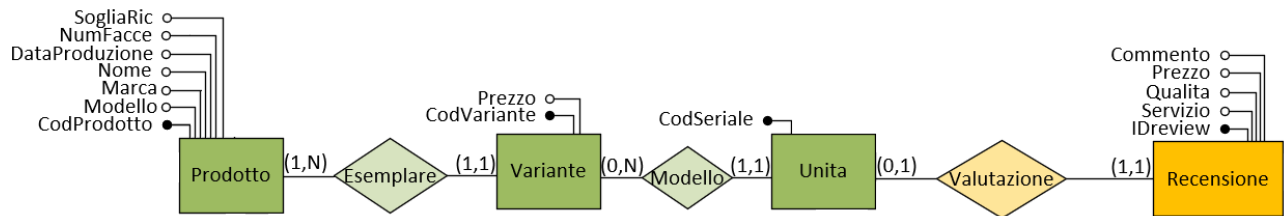
-FREQUENZA SUPPOSTA: 1/settimana

NOME	TIPO	VOLUME
Guasto	Entità	2.000
Errore	Entità	4.000
Tentativo	Relazione	6.000
Rimedio	Entità	6.000

CONCETTO	CoSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Guasto	Entità	1	Lettura	Leggo il nome del Guasto
Tentativo	Relazione	1,5	Lettura	Si trovano i codici dei possibili Rimedi
Rimedio	Entità	1,5	Lettura	Si legge la descrizione del Rimedio

Il costo totale dell'operazione è $1+1,5+1,5=4$ (settimanale).

5) La quinta operazione prevede di trovare la media dei voti delle recensioni di un Prodotto. Viene in sostanza fatta una media dei punteggi numerici delle categorie per cui è possibile assegnare un punteggio per ogni recensione, dopodiché viene nuovamente effettuata una media tra i punteggi medi di ogni recensione.



Per questa operazione sarà necessario partire dalla relazione Esemplare, per leggere tutte le Varianti che possiede il Prodotto di interesse; dopo verranno effettuate letture in Unità per trovare le unità fisiche esistenti per tutte le Varianti del prodotto; dovrà essere letta Valutazione per trovare le recensioni in cui corrispondono i Codici Seriali e infine verranno lette in Recensione

-DATI IN INPUT: CodProdotto

-DATI IN OUTPUT: MediaRecensioni

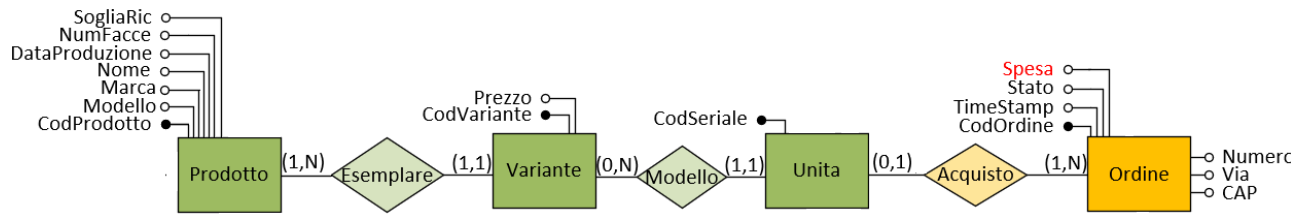
-FREQUENZA SUPPOSTA: 10/gg

NOME	TIPO	VOLUME
Prodotto	Entità	500
Esemplare	Relazione	1.000
Variante	Entità	1.000
Modello	Relazione	10.000
Unità	Entità	10.000
Valutazione	Relazione	500
Recensione	Entità	500

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Esemplare	Relazione	2	Lettura	Leggo le Varianti che il prodotto ha (in media 2)
Modello	Relazione	20	Lettura	Leggo i Codici Seriali delle Varianti che ho (in media 10 per ogni Variante)
Valutazione	Relazione	1	Lettura	Per ogni Unità ne ricavo l'IDreview
Recensione	Entità	1	Lettura	Leggo le informazioni delle varie recensioni

Il costo totale è di $10(2+20+1+1)=240$ (giornaliero).

6) La sesta operazione è di analisi, e prevede di trovare, per ogni prodotto acquistato almeno una volta, le sue Varianti più vendute.



Per realizzare l'operazione si parte da Acquisto, dove si leggono tutte le Unità acquistate, in seguito si leggono da unità le Varianti che rappresentano e da Variante i Prodotti a cui appartengono. Una volta ricavati questi dati si possono contare le occorrenze;

-DATI IN INPUT: Nessuno

-DATI IN OUTPUT: Per ogni Prodotto venduto la sua Variante più venduta

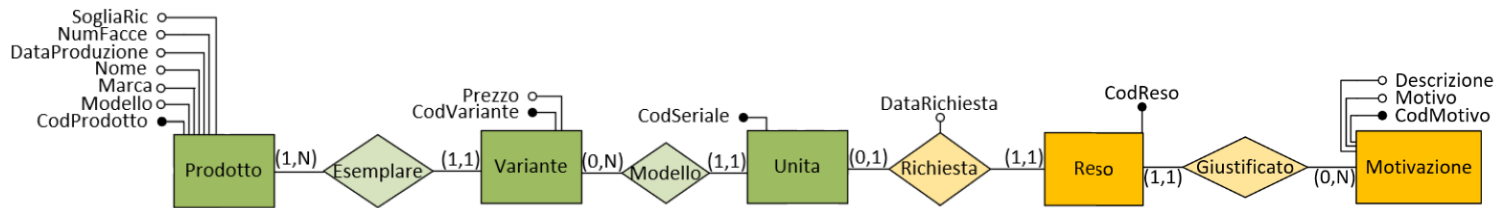
-FREQUENZA SUPPOSTA: 3/mese

NOME	TIPO	VOLUME
Prodotto	Entità	500
Esemplare	Relazione	1.000
Variante	Entità	1.000
Modello	Relazione	10.000
Unità	Entità	10.000
Acquisto	Relazione	4.000
Ordine	Entità	2.000

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Acquisto	Relazione	4.000	Lettura	Si leggono i Codici Seriali delle Unità acquistate
Unità	Entità	4.000	Lettura	Per ogni Unità si legge la sua Variante
Variante	Entità	1.000	Lettura	Per ogni Variante si legge il suo Prodotto e si possono così eseguire le operazioni sui dati

Il costo dell'operazione è $3 \cdot (4.000 + 4.000 + 1.000) = 27.000$ (mensile).

7) Anche la settima operazione è analitica: consiste nel fare una classifica dei Prodotti resi per diritto di recesso, che possono essere considerati i più insoddisfacenti.



Per eseguire questa operazione è necessario leggere da Reso tutti i resi effettuati per diritto di recesso e le Unità a cui si riferiscono; da Unità si leggono le relative varianti, da Variante si leggono i relativi codici dei prodotti e da Prodotto infine si ricava marca, nome e modello. A questo punto è possibile stilare la classifica.

-DATI IN INPUT: nessuno

-DATI IN OUTPUT: Classifica dei Prodotti insoddisfacenti

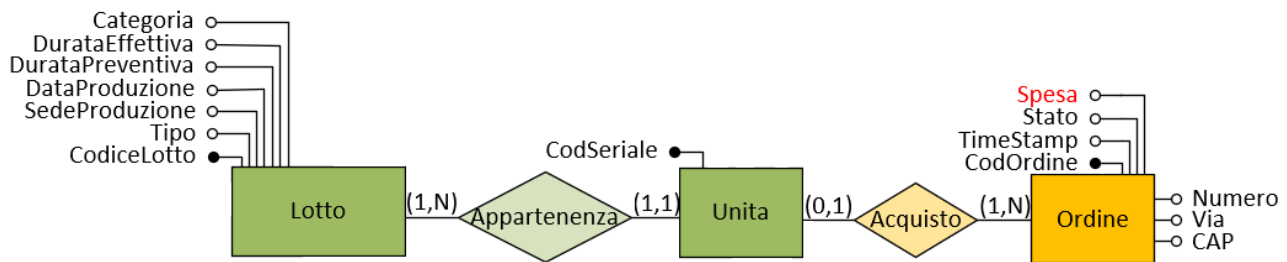
-FREQUENZA SUPPOSTA: 1/mese

NOME	TIPO	VOLUME
Prodotto	Entità	500
Esemplare	Relazione	1.000
Variante	Entità	1.000
Modello	Relazione	10.000
Unità	Entità	10.000
Richiesta	Relazione	200
Reso	Entità	200
Giustificato	Relazione	200
Motivazione	Entità	20

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Reso	Entità	200	Lettura	Si leggono tutti i resi, le relative unità e quali sono stati effettuati per diritto di recesso
Unità	Entità	200	Lettura	Si legge per ogni Unità la sua Variante
Variante	Entità	200	Lettura	Dalla Variante ricavo il Prodotto
Prodotto	Entità	200	Lettura	Si effettuano le operazioni necessarie (serve passare dal Prodotto per ricavare il nome)

Il costo dell'operazione è $200 \times 4 = 800$ (mensile).

8) L'ultima operazione prevede di ricavare, per ogni Lotto, quante delle sue Unità sono state vendute e di fare un rapporto tra le Unità vendute e quelle totali del Lotto, per avere una percentuale.



Per prima cosa va ricavato il numero di Unità in ogni Lotto, quindi saranno fatte tante letture in Appartenenza (per ogni Lotto) quante le Unità che compongono il Lotto. Dopo da Acquisto si ricava quante Unità sono state vendute per ogni Lotto e si può fare la divisione.

-DATI IN INPUT: nessuno

-DATI IN OUTPUT: Percentuale Vendite/Unità per ogni Lotto

-FREQUENZA SUPPOSTA: 4/mese

NOME	TIPO	VOLUME
Unità	Entità	10.000
Appartenenza	Relazione	10.000
Lotto	Entità	200
Acquisto	Relazione	4.000
Ordine	Entità	2.000

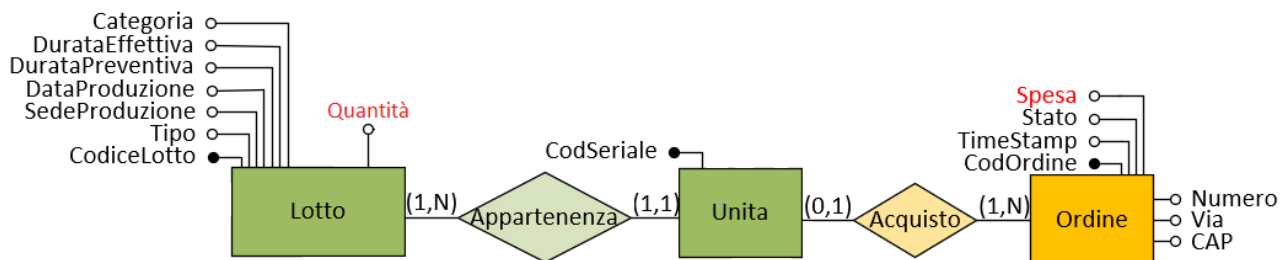
CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Appartenenza	Relazione	9.500	Lettura	Per ogni Lotto dobbiamo sapere quante Unità possiede. Le letture sono 9.500 e non 10.000 perché si suppone che 500 Unità appartengano a Lotti Ricondizionati, e non ci interessano
Acquisto	Relazione	4.000	Lettura	Per ogni Lotto si guarda quante delle sue Unità sono state acquistate, e cioè si contano quelle che compaiono nella relazione Acquisto

Il costo medio di questa operazione è $4 \cdot (9.500 + 4.000) = 40.500$ (mensile).

E' possibile ridurre il costo?

Si noti un dettaglio importante; quest'operazione viene eseguita 4 volte al mese (stima) e ogni volta deve essere nuovamente ricavato tramite 9.500 letture, il numero di Unità che compongono ogni Lotto; questa quantità di letture non è ottimale se si considera che potrebbero interessare all'azienda anche altre operazioni che prevedono di sapere quante Unità compongono i vari Lotti.

Per questo motivo vediamo se conviene inserire una ridondanza in Lotto che indica da quante Unità è composto; un dato che tra l'altro non cambia mai e quindi andrebbe fatta l'operazione per ricavare il dato solo al momento dell'inserimento del Lotto nel DataBase:



E avremmo i seguenti accessi:

CONCETTO	COSTRUTTO	ACCESSI	TIPO	DESCRIZIONE
Lotto	Entità	190	Lettura	Per ogni Lotto dobbiamo sapere quante Unità possiede. E' sufficiente leggere l'entità. Le letture sono 190 e non 200 perché si suppone che 10 Lotti siano di prodotti ricondizionati, e non ci interessano
Acquisto	Relazione	4.000	Lettura	Per ogni Lotto si guarda quante delle sue Unità sono state acquistate, e cioè si contano quelle che compaiono nella relazione Acquisto

Il costo adesso è $4 \cdot (190 + 4.000) = 16.760$ (mensile); la ridondanza si mantiene.

MODELLO LOGICO

Adesso segue il Modello Logico; i nomi delle tabelle saranno colorate in base all'Area di interesse a cui appartengono (verde, rosso, blu, giallo). Inoltre verranno rappresentate in viola le chiavi esterne e in generale gli attributi con vincoli di integrità referenziali, che saranno poi dichiarati esplicitamente subito dopo il Modello Logico.

Prodotto (CodProdotto, Modello, Marca, Nome, DataProduzione, NumFacce, SogliaRic, **PrimoTest**)

Variante (CodVariante, Prezzo, **Prodotto**)

Unità (CodSeriale, **Variante**, **Garanzia**, **Lotto**, **Acquisto**)

Parte (CodiceParte, NomeParte, PrezzoParte, Peso, **Prodotto**)

Costituzione (CodiceParte, NomeMateriale, Quantita)

Materiale (NomeMateriale, TassoSvalutazione, Valore)

Fissaggio (PrimaParte, SecondaParte, Giunzione)

Giunzione (Tipo)

InfoGiunzione (Giunzione, Caratteristica, Valore)

Caratteristica (NomeCaratteristica, UnitàDiMisura)

Operazione (IDoperazione, NomeOperazione, Faccia, TempoMedio, Varianza, **Parte**)

Vincoli (Precedente, Successiva)

Utilizzo (Operazione, Giunzione)

StrumentiOp (Utensile, Operazione)

Utensile (NomeUtensile)

Esecuzione (Stazione, Operazione, Ordine)

Stazione (CodStazione, Operatore, Sequenza)

Operatore (IDoperatore)

Sequenza (IDsequenza, TempoT)

Lotto (CodiceLotto, Tipo, SedeProduzione, DataProduzione, DurataPreventiva, DurataEffettiva, Categoria, Sequenza, Magazzino, Settore, DataStock, DataRilascio, **Prodotto**, Quantita)

Perdite (Lotto, Operazione, Stazione, Numero)

Magazzino (CodMagazzino, Capienza, Predisposizione)

Account (NumCliente, NomeUtente, Password, DomandaSicurezza, Risposta, Timestamp, CodFiscale)

Utente (CodFiscale, Nome, Cognome, Telefono, Numero, Via, Cap, **Documento**)

Documento (NumDocumento, Tipologia, DataScadenza, EnteRilascio)

Ordine (CodOrdine, Timestamp, Stato, Spesa, Numero, Via, CAP, **Cliente**)

Spedizione (IDspedizione, Stato, DataPrevista, Hub, DataUpdate, **Ordine**)

Recensione (IDreview, Servizio, Qualita, Prezzo, Commento, **Cliente**, **Unita**)

Garanzia (IDformula, Costo, Periodo)

Formula (Prodotto, **Garanzia**)

Validita (**Garanzia**, **Guasto**)

Reso (CodReso, DataRichiesta, **Unita**, **Motivo**)

Motivazione (CodMotivo, Motivo, Descrizione)

Guasto (CodGuasto, Nome, Descrizione, **Prodotto**, **PrimaDomanda**)

Errore (CodErrore, **Guasto**)

Rimedio (CodRimedio, Descrizione, **Errore**)

Domanda (Richiesta, **DomandaPrecedente**, RispostaPrecedente, **Rimedio**)

Diagnosi (CodAssistenza, **Guasto**)

AssistenzaFisica (Codice, Preventivo, Accettato, **Unita**)

Intervento (Ticket, DataIntervento, Tipo, Numero, Via, CAP, **Tecnico**, **CodAssistenza**)

Tecnico (Matricola, **CAPsede**)

Sede (CAP)

Disponibilita (**Tecnico**, **Giorno**, **Fascia**)

Orario (Giorno, **Fascia**)

OrdineParti (CodRicambio, DataOrdine, StimaConseg, DataConseg, **CodAssistenza**)

Lista (**Parte**, CodRicambio)

Ricevuta (Codice, Prezzo, Metodo, **CodAssistenza**)

Test (CodTest, Nome, Politica, **Padre**)

Funzionamento (**Test**, **Parte**)

Esito (**Unita**, **Test**, Fallito)

Conservato (Magazzino, **Reso**)



TABELLA DEI VINCOLI DI INTEGRITA' REFERENZIALE

Formattazione *Entità* ⇔ *Attributo*

OGGETTO	REFERENZA
Prodotto ⇔ PrimoTest	Test ⇔ CodTest
Variante ⇔ Prodotto	Prodotto ⇔ CodProdotto
Unita ⇔ Variante	Variante ⇔ CodVariante
Unita ⇔ Garanzia	Garanzia ⇔ IDformula
Unita ⇔ Lotto	Lotto ⇔ CodiceLotto
Unita ⇔ Acquisto	Ordine ⇔ CodOrdine
Parte ⇔ Prodotto	Prodotto ⇔ CodProdotto
Costituzione ⇔ CodiceParte	Parte ⇔ CodiceParte
Costituzione ⇔ NomeMateriale	Materiale ⇔ NomeMateriale
Fissaggio ⇔ PrimaParte	Parte ⇔ CodiceParte
Fissaggio ⇔ SecondaParte	Parte ⇔ CodiceParte
Fissaggio ⇔ Giunzione	Giunzione ⇔ Tipo
InfoGiunzione ⇔ Giunzione	Giunzione ⇔ Tipo
InfoGiunzione ⇔ Caratteristica	Caratteristica ⇔ NomeCaratteristica
Operazione ⇔ Parte	Parte ⇔ CodiceParte
Vincoli ⇔ Precedente	Operazione ⇔ IDoperazione
Vincoli ⇔ Successiva	Operazione ⇔ IDoperazione
Utilizzo ⇔ Operazione	Operazione ⇔ IDoperazione
Utilizzo ⇔ Giunzione	Giunzione ⇔ Tipo
StrumentiOp ⇔ Utensile	Utensile ⇔ NomeUtensile
StrumentiOp ⇔ Operazione	Operazione ⇔ IDoperazione
Esecuzione ⇔ Stazione	Stazione ⇔ CodStazione
Esecuzione ⇔ Operazione	Operazione ⇔ IDoperazione
Stazione ⇔ Operatore	Operatore ⇔ IDoperatore
Stazione ⇔ Sequenza	Sequenza ⇔ IDsequenza
Lotto ⇔ Sequenza	Sequenza ⇔ IDsequenza
Lotto ⇔ Magazzino	Magazzino ⇔ CodMagazzino
Lotto ⇔ Prodotto	Prodotto ⇔ CodProdotto
Perdite ⇔ Lotto	Lotto ⇔ CodiceLotto
Perdite ⇔ Operazione	Operazione ⇔ IDoperazione
Perdite ⇔ Stazione	Stazione ⇔ CodStazione
Account ⇔ CodFiscale	Utente ⇔ CodFiscale
Utente ⇔ Documento	Documento ⇔ NumDocumento
Ordine ⇔ Cliente	Account ⇔ NumCliente
Spedizione ⇔ Ordine	Ordine ⇔ CodOrdine
Recensione ⇔ Cliente	Account ⇔ NumCliente
Recensione ⇔ Unita	Unita ⇔ CodSeriale
Formula ⇔ Prodotto	Prodotto ⇔ CodProdotto

Formula ⇔ Garanzia	Garanzia ⇔ IDformula
Validita ⇔ Garanzia	Garanzia ⇔ IDformula
Validita ⇔ Guasto	Guasto ⇔ CodGuasto
Reso ⇔ Unita	Unita ⇔ CodSeriale
Reso ⇔ Motivo	Motivazione ⇔ CodMotivo
Guasto ⇔ Prodotto	Prodotto ⇔ CodProdotto
Guasto ⇔ PrimaDomanda	Domanda ⇔ Richiesta
Errore ⇔ Guasto	Guasto ⇔ CodGuasto
Rimedio ⇔ Errore	Errore ⇔ CodErrore
Domanda ⇔ DomandaPrecedente	Domanda ⇔ Richiesta
Domanda ⇔ Rimedio	Rimedio ⇔ CodRimedio
Diagnosi ⇔ CodAssistenza	AssistenzaFisica ⇔ Codice
Diagnosi ⇔ Guasto	Guasto ⇔ CodGuasto
AssistenzaFisica ⇔ Unita	Unita ⇔ CodSeriale
Intervento ⇔ Tecnico	Tecnico ⇔ Matricola
Intervento ⇔ CodAssistenza	AssistenzaFisica ⇔ Codice
Tecnico ⇔ CAPsede	Sede ⇔ CAP
Disponibilita ⇔ Tecnico	Tecnico ⇔ Matricola
Disponibilita ⇔ Giorno	Orario ⇔ Giorno
Disponibilita ⇔ Fascia	Orario ⇔ Fascia
OrdineParti ⇔ CodAssistenza	AssistenzaFisica ⇔ Codice
Lista ⇔ Parte	Parte ⇔ CodiceParte
Lista ⇔ CodRicambio	OrdineParti ⇔ CodRicambio
Ricevuta ⇔ CodAssistenza	AssistenzaFisica ⇔ Codice
Test ⇔ Padre	Test ⇔ CodTest
Funzionamento ⇔ Test	Test ⇔ CodTest
Funzionamento ⇔ Parte	Parte ⇔ CodiceParte
Esito ⇔ Unita	Unita ⇔ CodSeriale
Esito ⇔ Test	Test ⇔ CodTest
Conservato ⇔ Magazzino	Magazzino ⇔ CodMagazzino
Conservato ⇔ Reso	Reso ⇔ CodReso

NORMALIZZAZIONE BCNF

Di seguito è riportata ogni Relazione che è stata controllata per verificarne la normalizzazione, e ogni Relazione risulta già in Forma Normale di Boyce Codd

-Prodotto (CodProdotto, Modello, Marca, Nome, DataProduzione, NumFacce, SogliaRic, PrimoTest)

CodProdotto → Modello, Marca, Nome, DataProduzione, NumFacce, SogliaRic, PrimoTest

È già in **BCNF**

-Variante (CodVariante, Prezzo, Prodotto)

CodVariante → Prezzo, Prodotto

È già in **BCNF**

-Unità (CodSeriale, Variante, Garanzia, Lotto, Acquisto)

CodSeriale → Variante, Garanzia, Lotto, Acquisto

È già in **BCNF**

-Parte (CodiceParte, NomeParte, PrezzoParte, Peso, Prodotto)

CodiceParte → NomeParte, PrezzoParte, Peso, Prodotto

È già in **BCNF**

-Costituzione (CodiceParte, NomeMateriale, Quantita)

CodiceParte, NomeMateriale → Quantita

È già in **BCNF**

-Materiale (NomeMateriale, TassoSvalutazione, Valore)

NomeMateriale → TassoSvalutazione, Valore

È già in **BCNF**

-**Fissaggio** (PrimaParte, SecondaParte, Giunzione)

È già in **BCNF**

-**Giunzione** (Tipo)

È già in **BCNF**

-**InfoGiunzione** (Giunzione, Caratteristica, Valore)

Giunzione, Caratteristica → Valore

È già in **BCNF**

-**Caratteristica** (NomeCaratteristica, UnitàDiMisura)

NomeCaratteristica → UnitàDiMisura

È già in **BCNF**

-**Operazione** (IDoperazione, NomeOperazione, Faccia, TempoMedio, Varianza, Parte)

IDoperazione → NomeOperazione, Faccia, TempoMedio, Varianza, Parte

È già in **BCNF**

-**Vincoli** (Precedente, Successiva)

È già in **BCNF**

-**Utilizzo** (Operazione, Giunzione)

È già in **BCNF**

-**StrumentiOp** (Utensile, Operazione)

È già in **BCNF**

-**Utensile** (NomeUtensile)

È già in **BCNF**

-Esecuzione (Stazione, Operazione, Ordine)

Stazione, Operazione → Ordine

È già in **BCNF**

-Stazione (CodStazione, Operatore, Sequenza)

CodStazione → Operatore, Sequenza

È già in **BCNF**

-Operatore (IDoperatore)

È già in **BCNF**

-Sequenza (IDsequenza, TempoT)

IDsequenza → TempoT

È già in **BCNF**

-Lotto (CodiceLotto, Tipo, SedeProduzione, DataProduzione, DurataPreventiva, DurataEffettiva, Categoria, Sequenza, Magazzino, Settore, DataStock, DataRilascio, Prodotto, Quantita)

CodiceLotto → Tipo, SedeProduzione, DataProduzione, DurataPreventiva, DurataEffettiva, Categoria, Sequenza, Magazzino, Settore, DataStock, DataRilascio, Prodotto

È già in **BCNF**

-Perdite (Lotto, Operazione, Stazione, Numero)

Lotto, Operazione, Stazione → Numero

È già in **BCNF**

-Magazzino (CodMagazzino, Capienza, Predisposizione)

CodMagazzino → Capienza, Predisposizione

È già in **BCNF**

-Account (NumCliente, NomeUtente, Password, DomandaSicurezza, Risposta, Timestamp, CodFiscale)

NumCliente → NomeUtente, Password, DomandaSicurezza, Risposta, Timestamp, CodFiscale

È già in **BCNF**

-Utente (CodFiscale, Nome, Cognome, Telefono, Numero, Via, Cap, Documento)

CodFiscale → Nome, Cognome, Telefono, Numero, Via, Cap, Documento

È già in **BCNF**

-Documento (NumDocumento, Tipologia, DataScadenza, EnteRilascio)

NumDocumento → Tipologia, DataScadenza, EnteRilascio

È già in **BCNF**

-Ordine (CodOrdine, Timestamp, Stato, Spesa, Numero, Via, CAP, Cliente)

CodOrdine → Timestamp, Stato, Spesa, Numero, Via, CAP, Cliente

È già in **BCNF**

-Spedizione (IDspedizione, Stato, DataPrevista, Hub, DataUpdate, Ordine)

IDspedizione → Stato, DataPrevista, Hub, DataUpdate, Ordine

È già in **BCNF**

-Recensione (IDreview, Servizio, Qualita, Prezzo, Commento, Cliente, Unita)

IDreview → Servizio, Qualita, Prezzo, Commento, Cliente, Unita

È già in **BCNF**

-Garanzia (IDformula, Costo, Periodo)

IDformula → Costo, Periodo

È già in **BCNF**

-Formula (Prodotto, Garanzia)

È già in **BCNF**

-Validita (Garanzia, Guasto)

È già in **BCNF**

-Reso (CodReso, Motivo, CodMotivo, Descrizione, DataRichiesta, Unita)

CodReso → Motivo, CodMotivo, Descrizione, DataRichiesta, Unita

È già in **BCNF**

-Motivazione (CodMotivo, Motivo, Descrizione)

CodMotivo → Motivo, Descrizione

È già in **BCNF**

-Guasto (CodGuasto, Nome, Descrizione, Prodotto, PrimaDomanda)

CodGuasto → Nome, Descrizione, Prodotto, PrimaDomanda

È già in **BCNF**

-Errore (CodErrore, Guasto)

CodErrore → Guasto

È già in **BCNF**

-Rimedio (CodRimedio, Descrizione, Errore)

CodRimedio → Descrizione, Errore

È già in **BCNF**

-Domanda (Richiesta, DomandaPrecedente, RispostaPrecedente, Rimedio)

Richiesta → DomandaPrecedente, RispostaPrecedente, Rimedio

È già in **BCNF**

-Diagnosi (CodAssistenza, Guasto)

È già in **BCNF**

-AssistenzaFisica (Codice, Preventivo, Accettato, Unita)

Codice → Preventivo, Accettato, Unita

È già in **BCNF**

-Intervento (Ticket, DataIntervento, Tipo, Numero, Via, CAP, Tecnico, CodAssistenza)

Ticket → DataIntervento, Tipo, Numero, Via, CAP, Tecnico, CodAssistenza

È già in **BCNF**

-Tecnico (Matricola, CAPsede)

Matricola → CAPsede

È già in **BCNF**

-Sede (CAP)

È già in **BCNF**

-Disponibilita (Tecnico, Giorno, Fascia)

È già in **BCNF**

-Orario (Giorno, Fascia)

È già in **BCNF**

-OrdineParti (CodRicambio, DataOrdine, StimaConseg, DataConseg, CodAssistenza)

CodRicambio → DataOrdine, StimaConseg, DataConseg, CodAssistenza

È già in **BCNF**

-Lista (Parte, CodRicambio)

È già in **BCNF**

-Ricevuta (Codice, Prezzo, Metodo, CodAssistenza)

Codice → Prezzo, Metodo, CodAssistenza

È già in **BCNF**

-Test (CodTest, Nome, Politica, Padre)

CodTest → Nome, Politica, Padre

È già in **BCNF**

-Funzionamento (Test, Parte)

È già in **BCNF**

-Esito (Unita, Test, Fallito)

Unita, Test → Fallito

È già in **BCNF**

-Conservato (Magazzino, Reso)

È già in **BCNF**

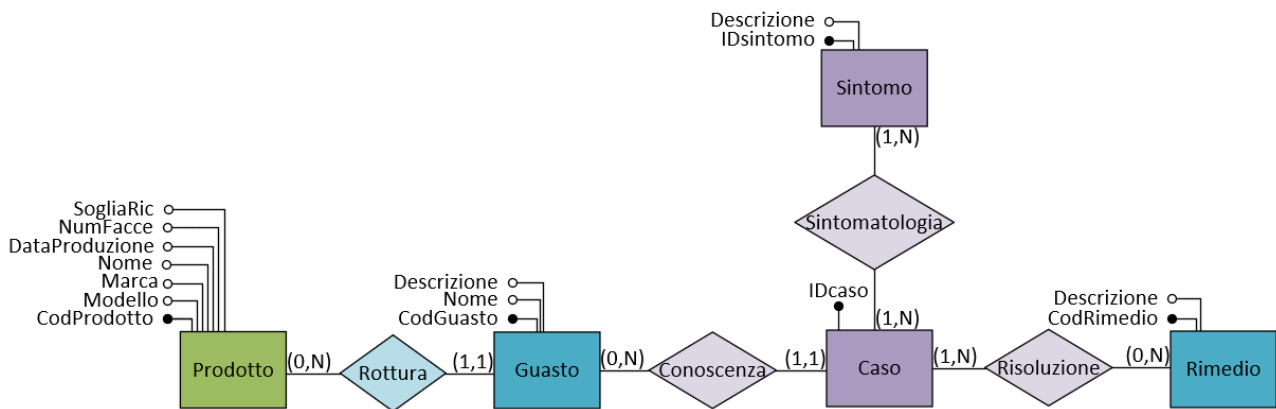
VINCOLI GENERICI

Segue la lista dei Vincoli Generici; sono riportati in **grassetto** quelli implementati.

- Le Sequenze sono composte da operazioni ordinate e atte a montare un particolare prodotto; è necessario che l'ordine rispetti i vincoli di precedenza tecnologica
- Il tipo di un Lotto può essere solo 'Prodotto' o 'Ricondizionato'
- La categoria di un Lotto può essere NULL, A, B, C e D**
- La Password di un Account deve essere composta da almeno 8 caratteri**
- In Recensione i punteggi di Servizio, Qualità, Prezzo sono interi compresi tra 1 e 5**
- Lo Stato di un Ordine può essere unicamente in processazione, in preparazione, spedito, evaso**
- Lo Stato di una Spedizione può essere unicamente spedito, in transito, in consegna, consegnato
- Un cliente non può recensire una stessa Variante di Prodotto più volte indipendentemente da quante unità della data Variante ha acquistato**
- Le Operazioni di ogni Stazione sono svolte sulla stessa Faccia
- In ogni Sequenza per un dato Prodotto, ogni Operazione compare solo una volta
- TempoMedio, Varianza, Ordine, TempoT, Numero, Quantita, hanno valore positivo
- Un Ordine appena aggiunto ha lo stato 'In Processazione'**
- Perché lo stato di un Ordine passi a 'In Processazione', devono esserci sufficienti Unità non ancora acquistate da evadere per l'ordine
- Controllo quando un Cliente esegue un Ordine che il suo Documento non sia scaduto
- Un Ordine può essere annullato solo se non ha superato lo stato 'In Processazione'
- Lo stato di un ordine deve seguire una sequenza definita (in Processazione, Preparazione, Spedito, Evaso)**
- Un'Unità può essere recensita solo se la sua Spedizione è in stato 'Consegnato'
- Un Reso effettuato per diritto di recesso può essere effettuato entro 30 giorni dalla consegna
- L'Albero di Domanda ha l'attributo Risposta che può assumere solo valori 'Sì' e 'No'**
- Intervento è dotato di un Indirizzo solo se il suo Tipo è 'Domicilio'
- Se l'Assistenza Fisica è coperta da Garanzia, non ha Preventivo, è automaticamente accettata e non ha Ricevuta
- OrdineParti contiene solo Parti relative all'Unità soggetta ad Assistenza
- L'attributo Fallito in Esito può valere solo 0 o 1**

DATA ANALYTICS – Diagnosi Intelligente dei Guasti CBR

Per prima cosa, affinché possano essere implementate tutte le funzionalità richieste, è necessario aggiungere alcune tabelle che verranno mostrate attraverso un **complemento di Diagramma ER**:



Si capisce facilmente che ogni Caso della base di conoscenza sia correlato ad un Guasto, e poiché sono memorizzati all'interno della base solo Casi risolti, ogni Caso è associato ad almeno un Rimedio.

Infine vanno memorizzati anche i Sintomi che si sono presentati per ognuno dei Casi.

Verranno analizzate adesso le varie fasi una alla volta.

1) FASE DI RETRIEVE

La Fase di RETRIEVE è realizzata tramite una procedura che viene chiamata una volta per ogni sintomo che l'utente inserisce, infatti **la procedura prende in ingresso un sintomo**. L'obiettivo di questa procedura, oltre a **salvare tutti i sintomi** e renderli utilizzabili da altre procedure all'interno della stessa sessione, è quello di **generare una tabella in cui sono memorizzati i casi simili e il numero di sintomi in comune tra ogni caso simile e il problema che ha l'utente**.

Nella prossima pagina verrà descritto nel dettaglio il codice.

Ecco il codice della Stored Procedure:

```
1  DROP PROCEDURE IF EXISTS Retrieve;
2  DELIMITER $$
3  CREATE PROCEDURE Retrieve (IN _IDSintomo INT)
4  BEGIN
5      DECLARE _max INT;
6
7      -- raccolta dei sintomi dell'utente
8      CREATE TEMPORARY TABLE IF NOT EXISTS Sintomi(
9          IDSintomo INT
10     )ENGINE = InnoDB DEFAULT CHARSET = latin1;
11
12     INSERT INTO Sintomi (IDSintomo)
13     VALUES (_IDSintomo);
14
15     -- costruzione di 2 temporary table, una per contenere i sintomi in
16     -- comune coi vari casi esistenti, una per i casi affini che verranno individuati
17     CREATE TEMPORARY TABLE IF NOT EXISTS SintomiInComune(
18         IDcaso INT,
19         Numero INT
20     )ENGINE = InnoDB DEFAULT CHARSET = latin1;
21
22     CREATE TEMPORARY TABLE IF NOT EXISTS CasiProbabili(
23         IDcaso INT,
24         Numero INT
25     )ENGINE = InnoDB DEFAULT CHARSET = latin1;
26
27     -- individuazione dei sintomi in comune coi vari casi
28     INSERT INTO SintomiInComune (IDcaso, Numero)
29     (SELECT SA.IDcaso, COUNT(SI.IDSintomo) AS Numero
30      FROM Sintomatologia SA INNER JOIN Sintomi SI ON SA.IDSintomo=SI.IDSintomo
31      GROUP BY SA.IDcaso);
32
33     -- salvo in una variabile il numero massimo di sintomi in comune individuato tra i vari casi
34     SET _max = (SELECT MAX(Numero) FROM SintomiInComune);
35
36     -- individuazione del caso/i più probabile e inserzione nella temporary table
37     TRUNCATE CasiProbabili;
38     INSERT INTO CasiProbabili (IDcaso, Numero)
39     (SELECT IDcaso, Numero
40      FROM SintomiInComune
41      WHERE Numero = _max);
42
43     TRUNCATE SintomiInComune;
44
45 END $$
46 DELIMITER ;
```


Questa Stored Procedure costruisce 3 Temporary Table:

- Sintomi**: contiene i sintomi inseriti (uno per chiamata) dall'utente
- SintomiInComune**: contiene il numero di sintomi in comune con ogni caso conosciuto
- CasiProbabili**: contiene i casi con il più alto numero di sintomi in comune, e il numero stesso

Tranne la Temporary Table Sintomi, ogni singola Table va troncata (svuotata) opportunamente all'interno della procedura stessa; infatti i dati corretti vengono inseriti solo all'ultima chiamata della RETRIEVE, quella in cui è stato inserito l'ultimo sintomo dall'utente; per questo i vecchi dati intermedi, risultato delle prime chiamate vanno sempre rimossi.

Va notato che all'interno della Table CasiProbabili, è contenuto l'IDcaso dei casi che presentano il massimo numero di sintomi in comune con l'utente, e il numero stesso; per questo motivo il numero sarà uguale per ogni singolo caso qualora se ne trovi più di uno. Questa ripetizione è necessaria perché servirà portare nelle prossime procedure questa informazione.

In conclusione dopo un numero di CALL Retrieve (IDSintomo) pari al numero di sintomi che si vuole inserire, si può passare alla fase di REUSE. **Vediamo in ogni caso un esempio di chiamate, in modo da poter proseguire con l'esempio nelle fasi successive:**



1	CALL Retrieve (1);
2	CALL Retrieve (2);
3	CALL Retrieve (4);
4	CALL Retrieve (5);

	IDSintomo	Descrizione
▶	1	Lo smartphone si comporta in modo irregolare bloccandosi senza un criterio
	2	Lo smartphone ci mette più del solito ad accendersi
	4	Lo smartphone presenta casualmente pop-up pubblicitari
	5	Lo smartphone si spegne irregolarmente senza preavviso

Visualizzazione dei sintomi inseriti tramite Retrieve

2) FASE DI REUSE

La fase di REUSE ha l'obiettivo di **stampare i rimedi suggeriti all'utente con un relativo punteggio di affidabilità**.

Per realizzare la fase di REUSE va definita la modalità in cui viene assegnato il punteggio a ogni rimedio che verrà suggerito all'utente.

 **La modalità per assegnare i punteggi ai rimedi è la seguente:**

- Di base, il punteggio di ogni rimedio parte da 10
- Per ogni sintomo presentato dall'utente ma che un caso associato non presenta, il rimedio associato al caso perde 1 punto
- Per ogni sintomo presentato dal caso associato ma che l'utente non presenta, il rimedio associato al caso perde 1 punto

La procedura di REUSE si occupa quindi di utilizzare gli IDcaso conservati nella Temporary Table CasiConsigliati (costruita nella fase di RETRIEVE) per fornire in proiezione ognuno dei rimedi associati ai vari casi simili con il proprio punteggio di affidabilità, calcolato secondo le regole sopra definite.

Segue ora l'analisi del codice della procedura:

```
1 DROP PROCEDURE IF EXISTS `Reuse`;
2 DELIMITER $$
3 CREATE PROCEDURE `Reuse` ()
4 BEGIN
5 -- costruzione di alcuni dati utili a calcolare il punteggio (i sintomi che si hanno
6 -- in piu rispetto a quelli dei casi trovati, e i sintomi mancanti rispetto ai casi trovati)
7 DECLARE _sintomiextra INT;
8 DECLARE _sintomiincomune INT;
9 DECLARE _sintomiposseduti INT;
10
11 SET _sintomiposseduti =
12 (SELECT COUNT(*)
13 FROM Sintomi);
14
15 SET _sintomiincomune =
16 (SELECT MAX(Numero)
17 FROM CasiProbabili);
18
19 SET _sintomiextra = _sintomiposseduti - _sintomiincomune;
20
21 CREATE TEMPORARY TABLE IF NOT EXISTS SintomiMancanti(
22 IDcaso INT,
23 Numero INT
24 )ENGINE = InnoDB DEFAULT CHARSET = latin1;
25
26 TRUNCATE SintomiMancanti;
27 INSERT INTO SintomiMancanti
28 (SELECT C.IDcaso, (COUNT(*) - _sintomiincomune)
29 FROM CasiProbabili C INNER JOIN Sintomatologia S ON C.IDcaso=S.IDcaso
30 GROUP BY C.IDcaso);
31
32 -- costruzione e popolamento della temporary table target, cioè quella che contiene i rimedi e i giusti punteggi
33 CREATE TEMPORARY TABLE IF NOT EXISTS RimediSuggeriti(
34 CodRimedio INT,
35 Rimedio VARCHAR(300),
36 Punteggio INT
37 )ENGINE = InnoDB DEFAULT CHARSET = latin1;
38
39 TRUNCATE RimediSuggeriti;
40 INSERT INTO RimediSuggeriti (CodRimedio, Rimedio, Punteggio)
41 (SELECT R.CodRimedio, R.Descrizione, (10 - S.Numero - _sintomiextra)
42 FROM SintomiMancanti S INNER JOIN Risoluzione Re ON S.IDcaso=Re.IDcaso
43 INNER JOIN Rimedio R ON Re.CodRimedio=R.CodRimedio);
44
45 SELECT Rimedio, Punteggio
46 FROM RimediSuggeriti
47 ORDER BY Punteggio DESC;
48 END $$
49 DELIMITER ;
```

Come si ricavi per ogni caso (consigliato) la sua lista di rimedi è banale, **il grosso di questa procedura è dedicata al calcolo del punteggio di ogni rimedio**; per capire le operazioni effettuate è necessario partire dalla **formula per il calcolo del punteggio**:

$10 - (\text{N}^\circ \text{ di sintomi che presenta il caso ma non l'utente}) - (\text{N}^\circ \text{ di sintomi che presenta l'utente ma non il caso})$

-Il primo dato, cioè il numero di sintomi che presenta il caso ma non l'utente, potrebbe variare per ognuno dei casi suggeriti (cioè quelli nella Temporary Table CasiProbabili), pertanto è necessaria una Temporary Table, chiamata SintomiMancanti che per ogni caso indica il numero di sintomi mancanti all'utente.

Vediamo nello specifico il codice relativo a questa Temporary Table:

```
21 CREATE TEMPORARY TABLE IF NOT EXISTS SintomiMancanti(  
22     IDcaso INT,  
23     Numero INT  
24 )ENGINE = InnoDB DEFAULT CHARSET = latin1;  
25  
26 TRUNCATE SintomiMancanti;  
27 INSERT INTO SintomiMancanti  
28     (SELECT C.IDcaso, (COUNT(*) - _sintomiincomune)  
29     FROM CasiProbabili C INNER JOIN Sintomatologia S ON C.IDcaso=S.IDcaso  
30     GROUP BY C.IDcaso);
```

In sintesi, si prende una tabella che contiene i casi consigliati e il numero dei suoi rimedi, ma al numero dei suoi rimedi viene sottratto il numero di sintomi in comune tra il caso e l'utente, che è salvato nella variabile `_sintomiincomune`.

(il valore di questa variabile si ricava facilmente, in quanto contenuto nella Temporary Table creata nella fase di RETRIEVE; nel codice scritto si prende il massimo numero di sintomi in comune anche se quel numero non cambia, come modo veloce per poter selezionare un solo valore).

-Il secondo dato, cioè il numero di sintomi che presenta l'utente ma non il caso, è uguale per ogni caso suggerito, quindi si ricava facilmente sottraendo al numero dei sintomi che inserisce l'utente il numero dei sintomi in comune:

```
11 SET _sintomiposseduti =  
12     (SELECT COUNT(*)  
13     FROM Sintomi);  
14  
15 SET _sintomiincomune =  
16     (SELECT MAX(Numero)  
17     FROM CasiProbabili);  
18  
19 SET _sintomiextra = _sintomiposseduti - _sintomiincomune;
```

Costruite queste due informazioni, è finalmente possibile proiettare i rimedi con i loro punteggi:

```
33 CREATE TEMPORARY TABLE IF NOT EXISTS RimediSuggestiti(  
34     CodRimedio INT,  
35     Rimedio VARCHAR(300),  
36     Punteggio INT  
37 )ENGINE = InnoDB DEFAULT CHARSET = latin1;  
38  
39 TRUNCATE RimediSuggestiti;  
40 INSERT INTO RimediSuggestiti (CodRimedio, Rimedio, Punteggio)  
41     (SELECT R.CodRimedio, R.Descrizione, (10 - S.Numero - _sintomiextra)  
42     FROM SintomiMancanti S INNER JOIN Risoluzione Re ON S.IDcaso=Re.IDcaso  
43     INNER JOIN Rimedio R ON Re.CodRimedio=R.CodRimedio);
```

La tabella contiene il codice dei rimedi suggeriti (che non viene proiettato), la descrizione del rimedio e il suo punteggio.

Il punteggio, aggiunto come terzo attributo, segue esattamente la formula definita sopra.

Anche in questo caso sono necessarie delle opportune TRUNCATE, questa volta in caso venga chiamata più volte di fila la Reuse; se così fosse è necessario che non si inseriscano nuovamente dati già presenti, quindi vengono svuotate le tabelle e riempite con gli stessi dati.

Proseguendo l'esempio iniziato sopra:

1 CALL Reuse;

Rimedio	Punteggio
▶ Ripristina lo Smartphone	8
Installa un antivirus sullo smartphone	8
Sostituisci la batteria	7
Svuotare la RAM chiudendo le Applicazioni in background	7

Come è possibile notare, anche se (per come è stata definita la Retrieve) i casi suggeriti hanno sempre lo stesso numero di sintomi in comune con l'utente (che è il più alto possibile), il punteggio dei rimedi dei vari casi suggeriti non è però necessariamente uguale con la formula adottata.

3) FASE DI REVISE

La fase di REVISE prende in ingresso i rimedi con i quali l'utente è stato in grado di risolvere il suo problema, e se tali rimedi sono reputati abbastanza diversi da quelli proposti da ognuno dei casi suggeriti presi singolarmente, allora verrà chiamata la procedura di RETAIN che si occuperà di salvare il problema risolto dell'utente come nuovo caso della base di conoscenza.

A ogni chiamata, la procedura di REVISE prende in ingresso un rimedio utilizzato dall'utente e un booleano, che sarà sempre FALSE tranne che all'inserimento dell'ultimo rimedio (questo serve a non valutare se andare o no nella fase di RETAIN prematuramente).



Come viene deciso se tenere o no il problema dell'utente?

-Il problema risolto dell'utente è reputato sufficientemente diverso da quelli proposti se i rimedi con i quali ha risolto il problema hanno nel loro insieme un **Grado di Diversità maggiore di 2**;

-Il Grado di Diversità è ottenuto confrontando i rimedi dell'utente con i rimedi di ogni caso associato, uno per uno;

Le regole sono:

-Dato un caso associato e i suoi rimedi, ogni rimedio sfruttato dall'utente che tale caso non suggeriva aumenta di 1 il Grado di Differenza con il caso

-Dato un caso associato e i suoi rimedi, ogni rimedio suggerito da tale caso che l'utente non ha sfruttato aumenta di 1 il Grado di Differenza con il caso

- ✦ **IMPORTANTE:** Esisterà quindi un Grado di Differenza per ogni caso di cui sono stati proposti i rimedi! **Il problema dell'utente viene aggiunto alla base di conoscenza solo quando TUTTI i Gradi di Differenza sono maggiori di 2**, infatti se questo non succede anche solo una volta il problema dell'utente è riconducibile al relativo caso.

L'obiettivo della procedura è quindi quello di costruire una Temporary Table che contiene tutti i casi associati (quelli di cui erano stati proposti i rimedi) e per ognuno di essi il numero di rimedi extra dell'utente e il numero di rimedi inutilizzati dall'utente.

Proseguendo lo stesso esempio considerato per le altre fasi, la tabella è così fatta:

	IDcaso	RimediExtra	RimediInutilizzati
▶	1	2	1
	3	2	1

Visione della Temporary Table ResocontoCaso

In sostanza, il Grado di Differenza per entrambi i casi reputati simili è di $2 + 1 = 3$

L'esempio considerato chiamerà quindi la procedura di RETAIN, ma prima segue l'analisi del codice.

```

30 CREATE TEMPORARY TABLE IF NOT EXISTS ResocontoCaso(
31     IDcaso INT,
32     RimediExtra INT,
33     RimediInutilizzati INT
34 )ENGINE = InnoDB DEFAULT CHARSET = latin1;
35
36 TRUNCATE ResocontoCaso;
37 INSERT INTO ResocontoCaso
38     (SELECT
39         R.IDcaso,
40         (_rimediutilizzati - COUNT(*)),
41         ((SELECT COUNT(*) FROM CasiProbabili C INNER JOIN Risoluzione R1 ON C.IDcaso=R1.IDcaso WHERE R1.IDcaso=R.IDcaso GROUP BY C.IDcaso) - COUNT(*)) AS RimediInutilizzati
42     FROM Soluzioni S INNER JOIN RimediCaso R ON S.IDrimedio=R.CodRimedio
43     GROUP BY R.IDcaso);
44
45
46 SET _gradodifferenza =
47     (SELECT MIN(RimediExtra + RimediInutilizzati)
48     FROM ResocontoCaso);
49
50 IF _gradodifferenza > 2 AND _checkretain = TRUE
51 THEN
52     CALL `Retain`;
53 END IF;

```

La logica di questo segmento è la seguente:

- Creazione della Temporary Table ResocontoCaso
- Inserimento dei dati visti a pagina precedente
- Salvataggio del più piccolo Grado di Differenza in una variabile
- Controllo: se tale variabile è maggiore di 2 viene chiamata la RETAIN



Ma come funziona l'inserimento dei dati in ResocontoCaso?

```

(SELECT
    R.IDcaso,
    (_rimediutilizzati - COUNT(*)),
    ((SELECT COUNT(*) FROM CasiProbabili C INNER JOIN Risoluzione R1 ON C.IDcaso=R1.IDcaso WHERE R1.IDcaso=R.IDcaso GROUP BY C.IDcaso) - COUNT(*)) AS RimediInutilizzati
FROM Soluzioni S INNER JOIN RimediCaso R ON S.IDrimedio=R.CodRimedio
GROUP BY R.IDcaso);

```

Questo SELECT estrae i dati che vengono inseriti in ResocontoCaso e può non essere di facile lettura, quindi **verranno analizzati i tre attributi inseriti uno ad uno.**

1) Come prima cosa viene inserito l'IDcaso

2) Il secondo attributo è il seguente:

(_rimediutilizzati - COUNT(*))

Dove _rimediutilizzati è il numero di rimedi utilizzati dall'utente e COUNT(*) risulta essere il numero di rimedi in comune tra quelli associati al caso e quelli sfruttati dall'utente;

Si ottengono così i rimedi extra utilizzati dall'utente e non relativi al caso

3) Il terzo attributo è ricavato tramite una Subquery:

```

((SELECT COUNT(*)
FROM CasiProbabili C INNER JOIN Risoluzione R1 ON C.IDcaso=R1.IDcaso
WHERE R1.IDcaso=R.IDcaso GROUP BY C.IDcaso)
- COUNT(*)) AS RimediInutilizzati

```

La Subquery seleziona il numero di rimedi proposti dal caso, una volta chiusa la Subquery viene sottratto a tale numero COUNT(*), che è ancora una volta il numero di rimedi in comune.

Si ottengono così i rimedi associati al caso inutilizzati dall'utente.

Come visto, nell'esempio iniziato il Grado di Differenza è 3 per ognuno dei casi simili, quindi verrà chiamata la procedura di RETAIN.

```
IF _gradodifferenza > 2 AND _checkretain = TRUE
THEN
    CALL `Retain`;
END IF;
```

4) FASE DI RETAIN

Se aggiungere o no il problema dell'utente alla base di conoscenza come nuovo caso è stato già deciso nella fase di REVISE, quindi **l'unico scopo di questa procedura è quello di aggiungere i dati posseduti e conservati nelle varie Temporary Table al Database**; la procedura viene infatti chiamata direttamente dalla REVISE in caso venga reputato opportuno.

Uno sguardo al codice:

```
1  DROP PROCEDURE IF EXISTS `Retain`;
2  DELIMITER $$
3  CREATE PROCEDURE `Retain` ()
4  BEGIN
5      DECLARE _lastid INT;
6
7      INSERT INTO Caso (Guasto)
8      (SELECT C2.Guasto
9      FROM CasiProbabili C1 INNER JOIN Caso C2 ON C1.IDcaso=C2.IDcaso
10     LIMIT 1);
11
12     SET _lastid = LAST_INSERT_ID();
13
14     INSERT INTO Sintomatologia (IDcaso, IDSintomo)
15     SELECT _lastid, IDSintomo
16     FROM Sintomi;
17
18     INSERT INTO Risoluzione (IDcaso, CodRimedio)
19     SELECT _lastid, IDrimedio
20     FROM Soluzioni;
21
22 END $$
23 DELIMITER ;
```

L'unica cosa di cui prendere nota per questa procedura è come viene scelto il Guasto da associare al nuovo caso che sta venendo creato.

La scelta fatta è banalmente quella di prendere il guasto associato ad uno dei casi a suo tempo reputati simili, e di catalogare il nuovo caso come inerente allo stesso guasto.