

Politecnico di Torino

Corso di Laurea in Ingegneria Gestionale



Relatore
Prof. Fulvio Corno

Studentessa
Concettina Prestanicola
S174280

Material Requirements Planning: implementazione in Java

Copia dell'Allegato 1

Studente proponente : s174280 Prestanicola Concettina

Titolo della proposta: "Material Requirements Planning:applicazione in Java"

Descrizione del problema proposto:

Semplicemente detto MRP,Material Requirements Planning è stato sviluppato da Joseph Orlicky negli anni Sessanta e gradualmente adottato da tutte le aziende americane.

Si tratta di una tecnica che calcola i fabbisogni netti dei materiali e pianifica gli ordini di produzione e di acquisto. A tale scopo bisogna tenere in considerazione la domanda del mercato,detta "domanda indipendente",la distinta base dei prodotti,i lead time di produzione e di acquisto e infine le giacenze dei magazzini.

Per ogni prodotto considerato, tale algoritmo suggerirà quali ordini di acquisto emettere e quando emetterli in modo che tutti i componenti per la produzione (la "domanda dipendente") giungano quando sono necessari.

Descrizione della rilevanza gestionale del problema:

MRP è stato utilizzato nelle aziende manifatturiere per oltre 30 anni e ancora oggi è un buon punto di partenza per molte aziende nonostante lavori con l'ipotesi di capacità produttiva infinita.

Di fatto ci permette di rispondere a queste domande:

- Che cosa produrre e acquistare?
- Quanto produrre?
- Quando produrre e acquistare?

permettendo così all'azienda di minimizzare le scorte,nonché i costi di stoccaggio, e massimizzare il livello di servizio.

Nell'ottica di domanda indipendente e dipendente,l'MRP funge da ponte: i fabbisogni dei prodotti finiti vengono trasformati nei fabbisogni dei componenti e delle materie prime.

L'algoritmo opera essenzialmente in tre fasi:

1. La **nettificazione** delle scorte: quanto bisogna produrre al netto delle giacenze.
2. La **precessione temporale**: anticipo della produzione e degli acquisti rispetto al momento del fabbisogno.
3. L'**esplosione dei fabbisogni**: trasmette ai componenti ed alle materie prime i fabbisogni netti.

Al termine dell'esecuzione dell'algoritmo, il sistema propone gli ordini di produzione e gli ordini di acquisti pianificati.

Descrizione dei data-set per la valutazione:

Il database che utilizzerò per testare le funzionalità della mia applicazione Java è costituito da 24 prodotti suddivisi su tre linee di produzione per un totale di 300 materiali.

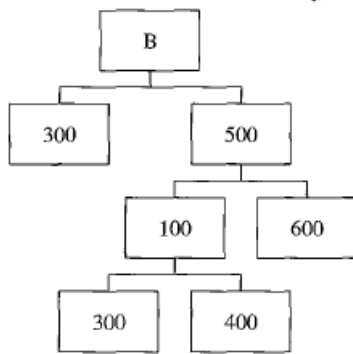
Ogni materiale è caratterizzato da:

- una descrizione testuale
- la quantità di lottizzazione
- la domanda mensile
- il magazzino iniziale
- il lead time di produzione per i materiali interni o il lead time di consegna per materiali esterni
- il valore monetario.

Descrizione preliminare degli algoritmi coinvolti:

L'idea principale è quella di utilizzare un algoritmo ricorsivo: sarà infatti necessario eseguire lo stesso insieme di operazioni man mano che scendiamo di livello nella distinta base del prodotto.

Per maggior chiarezza consideriamo un esempio:



Data la distinta base di B considero prima il fabbisogno della domanda indipendente, cioè quante unità di B produrre e in base al lead time e alle scorte di magazzino troverò il momento di inizio della produzione.

Successivamente sulla base delle quantità da produrre “esploderò” i fabbisogni delle domande dipendenti del primo livello, cioè i prodotti 300 e 500. Quest’ultimo prodotto presenta un secondo livello per cui dovrò cercare quante unità di 100 e 600 produrre. Itererò nell’esplosione dei fabbisogni fino a trovare la terminazione dell’albero.

Verrà utilizzato un algoritmo di ordinamento per la rappresentazione dei prodotti all’utente.

Descrizione preliminare delle funzionalità previste per l'applicazione software:

L'applicazione dovrà permettere all'utente di:

- Inserire un nuovo prodotto e contestualmente la sua distinta base, il lead time e la tecnica di lottizzazione utilizzata;
- Trovare le informazioni relative ad un prodotto tra quelli presenti;
- Selezionare un prodotto e determinare quindi l'MRP di quest'ultimo. I risultati saranno presentati nella consueta forma tabellare dell'MRP (Tabella 1).

Tabella 1

Part A	1	2	3	4	5	6	7	8
Gross requirements								
Scheduled receipts								
Adjusted SRs								
Projected on-hand								
Net requirements								
Planned order receipts								
Planned order releases								

Data di invio della proposta : 24 Novembre 2013

Descrizione del problema affrontato

Il successo di una società dipende dall'equilibrio delle funzioni gestionali tra operazioni interne ed esterne. Una società che riesce ad equilibrare queste funzioni riesce ad acquistare un certo rilievo nel mercato. Questo equilibrio viene raggiunto tramite la gestione della supply chain, che comprende le seguenti funzioni:

- Acquisto o produzione
- Pianificazione della previsione e della domanda
- Pianificazione della produzione e del magazzino
- Gestione magazzino logistico
- Distribuzione e trasporti
- Contabilità
- Assistenza clienti

Ognuna di queste funzioni ha una sua logica ed una sua importanza, tuttavia la pianificazione aziendale copre un ruolo cruciale all'interno dei settori di produzione e di distribuzione. Per tale ragione essa è stata ed è tuttora oggetto di numerosi studi intenti a semplificarne le operazioni e, viste la complessità e la noia di pianificare e gestire l'inventario servendosi solo di carta e penna, è stato del tutto naturale cercare di estenderla al calcolo automatico.

In questo contesto si inserisce Joseph Orlicky il quale insieme con altri all'IBM ha sviluppato il Material Requirements Planning. Semplicemente detto anche MRP, tale tecnica ha riscosso un notevole successo a partire dagli anni Settanta grazie alla campagna "MRP Crusade" dell'APICS (American Production and Inventory Control Society).

Prima dell' MRP la maggior parte dei sistemi di controllo della produzione si basava su varianti dei punti di riordino: in sostanza la produzione di qualsiasi parte, che fosse un prodotto finito o un componente, veniva richiesta quando il magazzino calava sotto un livello specifico. Orlicky e gli altri creatori di MRP si accorsero che questo approccio molto bene si confà ai prodotti finali in quanto la loro domanda è esterna al processo produttivo ed è quindi soggetta a incertezza, mentre per i componenti bisognava tener conto che essi vengono utilizzati per la produzione di prodotti finali e pertanto la loro domanda è

funzione della domanda dei prodotti finiti. Alla luce di ciò trattare i due tipi di domanda in modo equivalente significherebbe ignorare la dipendenza della domanda del componente su richiesta del prodotto finale e quindi porterebbe a inefficienze nella programmazione della produzione.

Il cuore dell'MRP risiede in queste due definizioni:

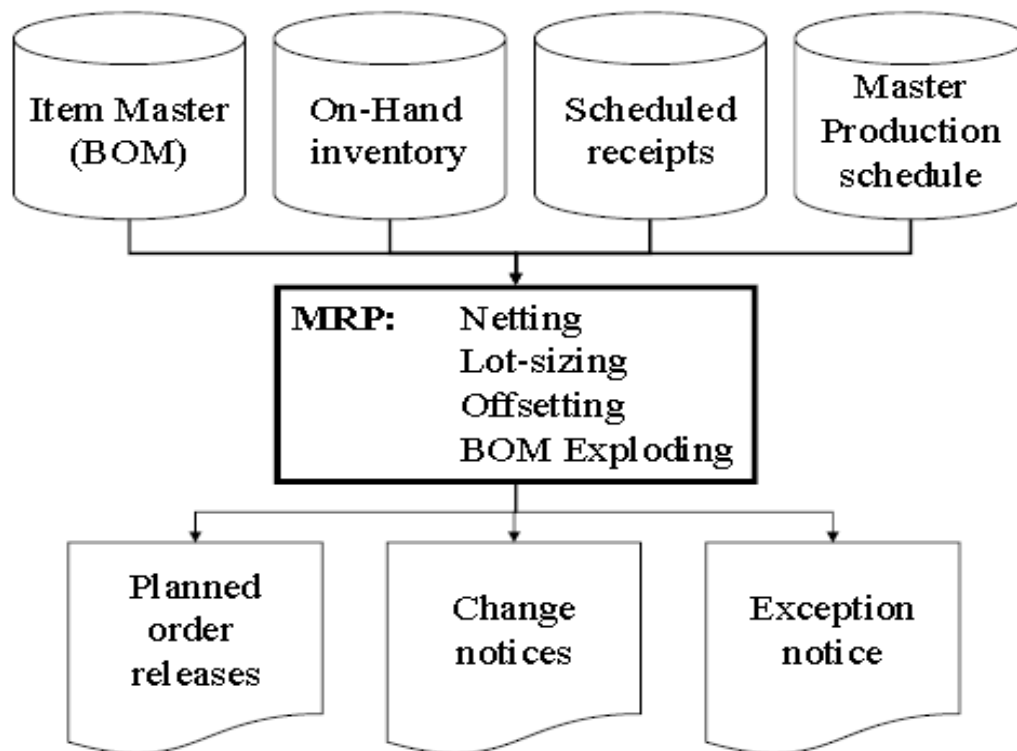
- **domanda indipendente:** qualsiasi domanda che nasce al di fuori del sistema produttivo. Si noti, a tal proposito, che l'esempio più intuitivo è la domanda di prodotti finiti ma ponendoci in un'ampia prospettiva anche la domanda di componenti potrebbe costituire una domanda indipendente qualora i suddetti componenti possano essere venduti separatamente (ad esempio pezzi di ricambio).
- **domanda dipendente:** è la domanda dei componenti che costituiscono il prodotto della domanda dipendente.

Tale sistema di programmazione è anche detto sistema "push" in quanto la produzione dei materiali viene "spinta" sulla base di un piano dettato da un'autorità centrale. In contrapposizione ricordiamo il sistema "pull" il cui tipo esempio è rappresentato dal sistema Kanban della Toyota: in questo caso la produzione viene attivata dal prelievo fisico di materiali piuttosto che da un piano prestabilito.

Come si evince dal termine, la funzione di base di MRP è pianificare i fabbisogni di materiali coordinando gli aspetti quantitativi e temporali: da un lato il sistema deve determinare le opportune quantità di produzione di tutti i tipi di articoli, dai prodotti destinati alla vendita ai componenti utilizzati per costruirli, nonché anche le opportune quantità di acquisto di materie prime e parti in outsourcing; d'altra parte deve anche determinare i tempi di produzione in modo da rispettare eventuali scadenze.

In molti sistemi MRP, il tempo è suddiviso in "bucket" così da discretizzare la rispettiva domanda. La dimensione di questi bucket è a discrezione aziendale ma è ovvio che minore è tale dimensione, maggiori sono l'accuratezza e la precisione della pianificazione.

Material Requirements Planning



Inputs e output

Gli inputs necessari al funzionamento del sistema MRP sono:

1. Master Production Schedule(MPS): fornisce quantità e date di scadenza per tutte le parti che costituiscono la domanda dipendente. Tali dati vengono utilizzati come richieste lorde per avviare la procedura di pianificazione.
2. Master File Item: contiene le informazioni di base su ogni articolo, quali codice, descrizione, distinta base, lot-sizing, lead-time, politiche di riordino e così via. Molto interessante da comprendere è la distinta base, detta anche bill of material: questa contiene tutti i materiali di cui un prodotto è composto e tutte le rispettive quantità necessarie alla realizzazione del prodotto stesso. Le informazioni sul lead time sono necessarie per risalire alla tempistica di produzione, mentre quelle di lot sizing sono imputate alla politica di riordino dell'azienda.
3. On -Hand Inventory: sono i dati sulla quantità delle parti attualmente in magazzino, incluse le scorte.
4. Scheduled Receipts: contiene tutti gli ordini rilasciati in precedenza, sia d'acquisto sia di produzione , che sono stati confermati e sono quindi in attesa di essere consegnati. È possibile che le tempistiche di tali richieste non vengano rispettate per cui è necessario operare degli aggiustamenti nella pianificazione a monte.

Gli outputs prodotti dall'MRP sono i seguenti:

1. Planner order release: rappresentano gli job che dovranno essere eseguiti nel caso di componenti interne e gli ordini di acquisti che dovranno essere rilasciati per componenti outsourcing. In questo contesto vengono rilasciate per ogni materiale informazioni su quantità richieste e date di scadenza. Sono quindi il risultato operativo e desiderato di tutta la tecnica di pianificazione.

2. Change notices: si tratta di notifiche vengono rilasciate nel caso di cambiamenti delle date di consegna, quindi notifiche per anticipi o ritardi.
3. Exception reports: il sistema informa l'utente che vi sono discrepanze tra ciò che si è progettato e ciò che è possibile realizzare.

Algoritmo di MRP

Prima di procedere all'esposizione delle operazioni eseguite per il calcolo dei fabbisogni diamo alcune notazioni, così da semplificare la comprensione. Per ogni materiale si definisce:

- $D(t)$ = fabbisogno lordo (domanda) per il periodo t
- $S(t)$ = quantità che sarà disponibile al tempo t
- $I(t)$ = magazzino disponibile alla fine del tempo t
- $N(t)$ = fabbisogno netto del periodo t

A questo punto possiamo descrivere le fasi principali che interessano l'MRP.

- **Nettificazione** consiste in due operazioni fondamentali. La prima è modulare gli scheduled receipts accelerando quelli che sono attualmente in programma di arrivare troppo tardi e rimandando quelli attualmente in programma di arrivare troppo presto; la seconda è calcolare la domanda netta. Per calcolare quando il primo $S(t)$ dovrebbe arrivare, bisogna determinare per quanti bucket, cioè periodi, l'on hand inventory coprirà la domanda, ovvero: $I(t) = I(t-1) - D(t)$. Iteriamo in questo calcolo finché $I(t) < 0$, a questo punto t è il periodo in cui dobbiamo far arrivare il primo ordine. Si noti che se l'arrivo di tale ordine era stato programmato per un periodo anteriore a t bisognerà posticiparlo invece per un periodo posteriore bisognerà anticiparlo. Tali eventi daranno luogo ad una change notice.

Notificata questa modifica, l'inventario deve essere aggiornato computando le quantità programmate in arrivo: $I(\text{dopo la modifica di } S) = I(\text{prima la modifica di } S) + S(t)$.

L'anticipo/il posticipo degli ordini per il periodo t si ripete fino a quando $I(t) > 0$.

Tale operazione di aggiustamento degli ordini va avanti finché gli scheduled receipts sono esauriti o finché non si è raggiunta la fine dell'orizzonte temporale.

Terminate queste modifiche il calcolo dei fabbisogni netti è semplice: sia t^* il primo periodo in cui $I(t) < 0$ dopo che gli scheduled receipts sono stati sistemati, allora:

$$N(t) = \begin{cases} 0 & \text{se } t < t^* \\ -I(t) & \text{se } t = t^* \\ D(t) & \text{se } t > t^* \end{cases}$$

Ovvero i fabbisogni netti saranno nulli per tutti i periodi precedenti t^* , pari al valore negativo del magazzino periodo t^* e pari alla domanda lorda per i periodi successivi a t^* .

- **Lottizzazione** programma le quantità necessarie a soddisfare i fabbisogni netti che sono stati calcolati nella fase precedente. A tal proposito si assume I una volta

determinati fabbisogni netti è necessario. Occorre a questo punto ricordare che esistono molte tecniche di lottizzazione(fixed order quantity,lot for lot,fixed order period,ecc.) ma la nostra attenzione si focalizza solo su una di esse:la fixed order quantity in base alla quale la quantità di riordino è pari ad una quantità predeterminata. Occorre dire che questa semplice tecnica è spesso anche la più utilizzata realmente in quanto l'azienda conosce a priori i costi di trasporto e di setup sostenuti e presumibilmente tali costi saranno costanti nel tempo per una fissata quantità.

- **Precessione temporale** verte sull'anticipo della produzione e degli acquisti rispetto al momento del fabbisogno,cioè determina la data di rilascio dell'ordine traslando la data di consegna. Generalmente si presuppone che i lead time siano fissi.
- **Esplosione dei fabbisogni** trasmette a tutti i componenti ed a tutte le materie prime le quantità di parti ordinate necessarie per l'orizzonte temporale considerato.

Tali passi vengono iterati per tutti i livelli della distinta base,cioè per ogni componente del prodotto finito. Al termine dell'esecuzione dell'algoritmo, il sistema propone gli ordini di produzione e gli ordini di acquisti pianificati.

Per una migliore comprensione si riporta di seguito un esempio applicativo molto semplice ma completo relativo alla produzione di uno sgabello:esso è costituito da un sedile,due bulloni e da una base che a sua volta è costituita da quattro gambe e quattro bulloni.Nell'esempio in esame le tecniche di lottizzazione sono diverse tuttavia questa comporta una variazione solo a livello quantitativo e non qualitativo.

Week	1	2	3	4	5	6
Domanda	0	0	0	0	120	0

LLC	Liv	Nome	Qty	LT	IO	LS
0	0	Sgabello	1	1	20	FOQ 100
1	.1	Sedile	1	1	0	L4L
1	.1	Base	1	1	0	L4L
2	.2	Gamba	4	2	0	POQ 2
2	.2	Bullone	4	2	0	POQ 3
2	.1	Bullone	2	2	0	POQ 3

Pianificazione livello 0

Item	Sgabello	Sgabello	t=0	1	2	3	4	5
I(0)	20	Gross requirement - $D(t)$		0	0	0	0	120
SS	0	Scheduled receipt						
		Adjusted Scheduled receipt - $SR(t)$						
Lot type	FOQ	Projected on-hand inventory - $I(t)$	20	20	20	20	20	0
Lot value	100	Net requirement - $N(t)$		0	0	0	0	100
		Planned order receipt - $PR(t)$						100
Lead time	1	Planned order release - $PO(t)$	0	0	0	0	100	0

Pianificazione livello 1

Item	Gamba	Gamba	t=0	1	2	3	4	5	6
I(0)	0	Gross requirement - $D(t)$		0	0	400	0	0	0
SS	0	Scheduled receipt							
		Adjusted Scheduled receipt - $SR(t)$							
Lot type	POQ	Projected on-hand inventory - $I(t)$	0	0	0	0	0	0	0
Lot value	2	Net requirement - $N(t)$		0	0	400	0	0	0
		Planned order receipt - $PR(t)$				400			
Lead time	2	Planned order release - $PO(t)$	0	400	0	0	0	0	0

Item	Bullone	Bullone	t=0	1	2	3	4	5	6
I(0)	0	Gross requirement - $D(t)$		0	0	400	200	0	0
SS	0	Scheduled receipt							
		Adjusted Scheduled receipt - $SR(t)$							
Lot type	POQ	Projected on-hand inventory - $I(t)$	0	0	0	200	0	0	0
Lot value	3	Net requirement - $N(t)$		0	0	400	0	0	0
		Planned order receipt - $PR(t)$				600			
Lead time	2	Planned order release - $PO(t)$	0	600	0	0	0	0	0

Pianificazione livello 2

Item	Gamba	Gamba	t=0	1	2	3	4	5	6
I(0)	0	Gross requirement - $D(t)$		0	0	400	0	0	0
SS	0	Scheduled receipt							
		Adjusted Scheduled receipt - $SR(t)$							
Lot type	POQ	Projected on-hand inventory - $I(t)$	0	0	0	0	0	0	0
Lot value	2	Net requirement - $N(t)$		0	0	400	0	0	0
		Planned order receipt - $PR(t)$				400			
Lead time	2	Planned order release - $PO(t)$	0	400	0	0	0	0	0

Item	Bullone	Bullone	t=0	1	2	3	4	5	6
I(0)	0	Gross requirement - $D(t)$		0	0	400	200	0	0
SS	0	Scheduled receipt							
		Adjusted Scheduled receipt - $SR(t)$							
Lot type	POQ	Projected on-hand inventory - $I(t)$	0	0	0	200	0	0	0
Lot value	3	Net requirement - $N(t)$		0	0	400	0	0	0
		Planned order receipt - $PR(t)$				600			
Lead time	2	Planned order release - $PO(t)$	0	600	0	0	0	0	0

Limiti del sistema MRP

Sebbene abbia riscosso notevole successo, il sistema MRP presenta dei problemi che non possono essere ignorati. I principali sono essenzialmente:

- La capacità produttiva è ignorata e ciò può portare a piani di produzione non ammissibili. Nel sistema infatti non viene considerato quanto lavoro l'impianto di produzione debba svolgere in quanto si assume implicitamente che la capacità delle linee sia infinita.
- I lead time devono essere noti a priori e sono costanti nel tempo, indipendentemente dalla quantità di produzione. È un'assunzione del tutto irrealistica considerando che le tempistiche hanno una distribuzione casuale. Ciò ha grande impatto sulla politica di gestione delle scorte: un'azienda per soddisfare la domanda del cliente adotterà

lead time pessimistici, ragion per cui si troverà con un ampio magazzino di scorte: ciò impone alti costi di stoccaggio.

- L'instabilità del sistema MRP è alta: non è un caso infatti che si parli di "nervosismo". Una piccola variazione nel piano di produzione può infatti produrre strani effetti sugli ordini pianificati: un piano non ammissibile diventa fattibile solo perché la domanda diminuisce o in seguito all'adozione di una diversa tecnica di lottizzazione.

Questi limiti, insieme con un sistema di previsione e di integrazione con la contabilità, hanno portato allo sviluppo di una nuova generazione di MRP: il Manufacturing Resources Planning noto come MRP II. Quest'ultimo a sua volta si è evoluto nell'ERP: Enterprise Resources Planning.

Descrizione del data-set utilizzato per l'analisi

Il dataset utilizzato per testare le funzionalità dell'applicazione Java realizzata è composto da circa 200 materiali.

Esso è strutturato in due tabelle:

- Prodotto: qui sono definiti tutti i materiali, ognuno dei quali è caratterizzato da ID, descrizione testuale, domanda mensile, lead time, quantità di lottizzazione e valore monetario.
Sulla base delle descrizioni e dei codici è possibile distinguere i materiali in sei categorie:
 - PRODOTTO FINITO: il cui codice inizia per PS. Nel database sono presenti 24 prodotti finiti e sono quelli destinati alla vendita diretta e per i quali si calcola l'MRP.
 - COMPONENTE ACQUISTATO: il cui codice inizia per CA. Si contano oltre 40 componenti acquistati: sono quindi materiali che vengono acquisiti all'esterno dell'azienda.
 - COMPONENTE FABBRICATO: il cui codice inizia per CM. Sono presenti circa 90 materiali di questo tipo. Si tratta di componenti che vengono realizzati internamente all'azienda.
 - BOX: il cui codice inizia per B. Appartengono a tale categoria 12 materiali.
 - VARIE: il cui codice inizia per V. Sono presenti solo 2 elementi.
 - MATERIA PRIMA: il cui codice inizia per R. Si contano 22 voci.
- Bom: contiene le relazioni esistenti tra due materiali. Tale relazione è definita mediante i due identificativi, cioè il codice, dei prodotti: uno rappresenta il genitore e l'altro il figlio. A tale coppia è associata la quantità di materiale-figlio necessaria per la produzione di un materiale-padre.

Il database è stato fornito dal prof. Paolo Chiabert, docente di "Programmazione e Gestione della Produzione" al Politecnico di Torino.

I dati sono gli stessi che vengono utilizzati per la simulazione della gestione di un'azienda con SAP.

4.Descrizione ad alto livello delle strutture dati e degli algoritmi utilizzati

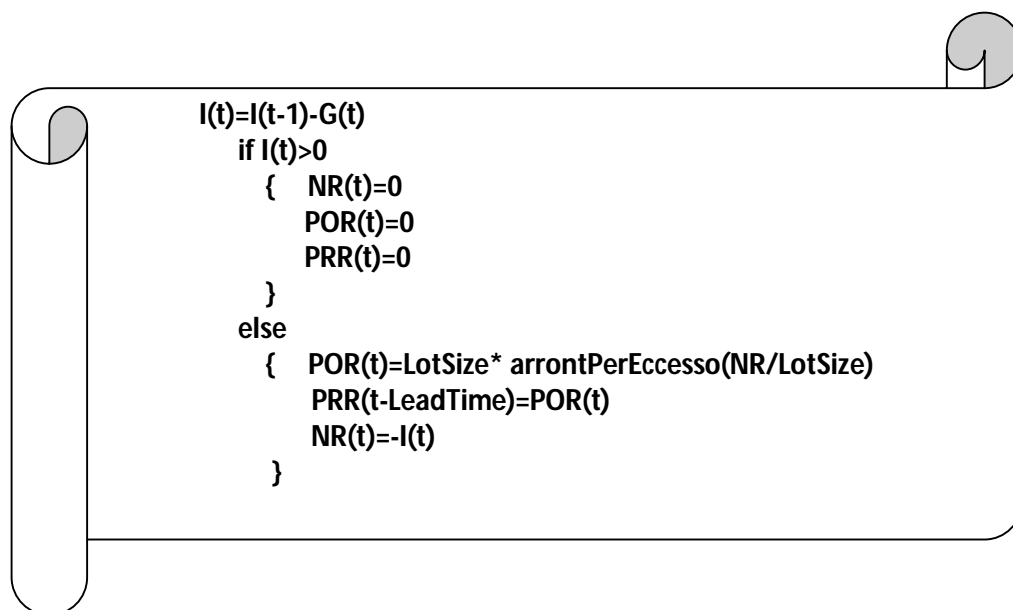
Le classi utilizzate nell'applicazione realizzata sono le seguenti:

- ❖ **ProdottoDAO:** è una semplice classe DAO per l'estrazione dei dataset dal database.
- ❖ **DBConnect:** è la classe per la connessione al database ed è quindi usata nella classe DAO.
- ❖ **Model, ControllerGui e Main:** sono le classi per la gestione dell'interfaccia grafica.
- ❖ **Prodotto:** è un bean necessario per la gestione dei diversi materiali. Ogni prodotto è definito da codice, descrizione testuale, quantità di lottizzazione, domanda mensile, lead time(di produzione per i materiali interni o di consegna per materiali esterni), valore monetario e magazzino iniziale. Quest'ultimo si pone pari ad un valore costante per tutti i prodotti. Degno di nota è l'attributo <<parents>>: si tratta di una mappa in cui vengono memorizzati i genitori del prodotto in esame e le rispettive quantità necessarie. In sostanza il prodotto è un costituente dei padri. Si parla di padri perché un prodotto può fungere da componente per più tipi di prodotti ma in quantità diverse. Il suddetto attributo sarà particolarmente utile nella creazione della distinta base.
- ❖ **BillOfMaterials:** è una delle principali classi del progetto in quanto permette la creazione della distinta base di un prodotto finito mediante un multigrafo orientato. Si è scelto questo tipo di struttura perché sembra meglio rappresentare l'effettiva struttura di una distinta base: ogni vertice costituisce un materiale e ogni arco permette di conservare la connessione tra due materiali. Tale connessione è orientata: dato un prodotto p, esso è costituito dai prodotti m ed n ma non vale il contrario: non posso percorrere il grafo a ritroso. Poiché uno stesso prodotto può essere il risultato dell'assemblaggio di più prodotti si è optato per un multigrafo.
Il grafo risultante quindi parte da un vertice (il prodotto finito) per snodarsi in un insieme di vertici (tutti i componenti) il cui collegamento viene effettuato sulla base dell'attributo <<parents>> della classe Prodotto.
La creazione del grafo avviene mediante l'invocazione del metodo << creaBom(Prodotto p, List<Prodotto> listaComponenti)>> i cui parametri sono rispettivamente il prodotto finito e la lista di tutte le relazioni tra i diversi prodotti. Inizializzato il grafo e aggiunto il vertice sorgente, tutta l'implementazione ruota intorno al metodo << cercaComponenti(Prodotto p, List<Prodotto> listaComponenti)>>. Si tratta di un algoritmo ricorsivo che crea le varie connessioni tra i vertici nel seguente modo: dato un prodotto p e la consueta listaComponenti che, ricordiamo, contiene tutte le relazioni tra i prodotti, si cercano tutti i componenti il cui padre è proprio il prodotto p e si memorizzano in una lista. Si creano quindi i diversi archi tra il prodotto p e gli elementi della lista appena costruita mediante la chiamata al metodo <<aggiungiComponente(Prodotto padre, Prodotto figlio) >>. A questo punto la procedura viene ripetuta ricorsivamente per ogni elemento della lista creata fino all'esaurimento.

❖ **AlgoritmoMRP**: è la classe che racchiude le funzionalità dell'intera applicazione. Quando si invoca il costruttore di questa classe tutti gli elementi della tabella che conterrà i risultati della pianificazione vengono inizializzati a zero e una mappa creata in modo tale da poter mantenere memoria della tabella di pianificazione di ogni prodotto. Il codice di quest'ultimo è infatti la chiave della mappa. Prima di procedere nell'esposizione dell'algoritmo occorre fermarsi sulla composizione di questa tabella. La struttura utilizzata è una matrice:

- * Le colonne indicano l'orizzonte temporale che si assume coprire sei bucket($t=1 \dots 6$);
- * Le righe, rifacendoci alla notazione vista precedentemente, indicano :
 1. La domanda del materiale $D(t)$
 2. L'inventario disponibile $I(t)$
 3. Il fabbisogno netto $N(t)$
 4. L'ordine pianificato $POR(t)$
 5. Il rilascio dell'ordine pianificato $PRR(t)$

Detto ciò l'implementazione della tecnica di MRP passa per il metodo `<< generaMRP(Prodotto p, BillOfMaterials bom) >>` il quale riceve in input il prodotto finito e la relativa distinta base. Si tratta in realtà di un metodo di comodo in quanto dopo aver riempito un vettore con la domanda (ipotizzata costante per ogni bucket) del prodotto finito, viene chiamato un metodo avente stesso nome ma firma diversa: oltre al prodotto e alla distinta base è necessario passare anche il vettore inizializzato poco prima. Si passa dunque al calcolo della tabella MRP per il prodotto finito (ricordiamo che l'ultima riga rappresenta il rilascio dell'ordine pianificato nonché quindi i fabbisogni necessari a soddisfare la domanda indipendente) usando l'algoritmo che precedentemente è stato illustrato sopra e di cui segue una notazione più formale:



```
I(t)=I(t-1)-G(t)
if I(t)>0
{  NR(t)=0
   POR(t)=0
   PRR(t)=0
}
else
{  POR(t)=LotSize* arrontPerEccesso(NR/LotSize)
   PRR(t-LeadTime)=POR(t)
   NR(t)=-I(t)
}
```

Tale operazione viene eseguita per tutti gli t bucket e al suo termine la tabella risultante viene memorizzata nella mappa.

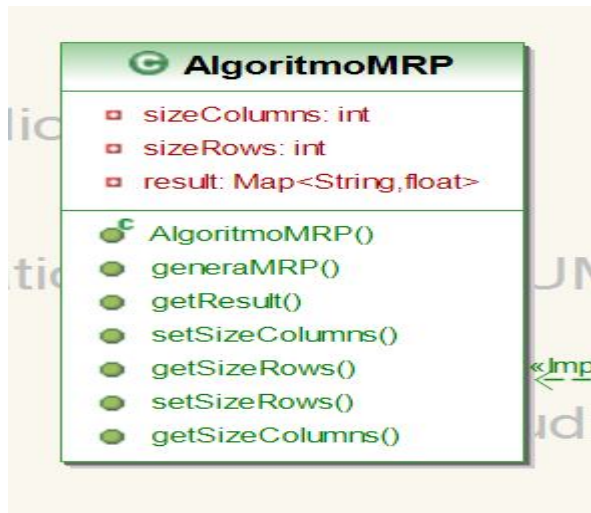
Bisogna a questo punto prepararsi a far esplodere l'MRP collegando i fabbisogni del prodotto p ai fabbisogni dei suoi componenti. A tale scopo si cercano tutti i nodi successori di p (che non sono altro che i suoi componenti per come è stata costruita la bill of material) e per ognuno di essi si chiama il metodo `<< (Prodotto p, float richieste[], BillOfMaterials bom, Prodotto parent >>` che opera allo stesso modo del metodo `<< generaMRP(Prodotto p, float domandaIndipendente[], BillOfMaterials bom) >>` con l'unica differenza che nel calcolo del fabbisogno dei componenti occorre considerare le quantità necessarie di questi per la produzione del prodotto. L'algoritmo termina nel momento in cui non le domande siano nulle oppure una volta esauriti i nodi.

Nel caso in cui la pianificazione non sia possibile perché è prevista per un tempo anteriore al tempo 0, l'applicazione lo segnala al fondo delle altre tabelle.

Diagramma delle classi principali

Si riportano i diagrammi delle classi `Algorithm oMRP`, `BillOfMaterials` e `Prodotto`





L'applicazione realizzata e i risultati ottenuti

The image shows a screenshot of a web application titled "Pianificazione MRP". The main heading is "Material Requirements Planning".

The interface includes the following elements:

- Prodotto:** A dropdown menu and a "Cerca" button.
- Prodotto Finito:** A dropdown menu and a "Genera MRP" button.
- NUOVO PRODOTTO:** A section with input fields for "Codice", "Descrizione", "LotSize", "LeadTime", "Domanda", and "Valore".
- Distinta base del nuovo prodotto:** A section with input fields for "Prodotto", "Componente", and "Quantità".
- Buttons:** "Inserisci" and "Aggiungi".
- Results Area:** A large yellow rectangular area at the bottom of the page, likely intended for displaying the results of the MRP calculation.

Grazie all'applicazione realizzata è possibile gestire la pianificazione degli ordini di produzione e degli acquisti secondo la tecnica MRP. Indipendentemente dai limiti precedentemente descritti di cui soffre tale tecnica, si può dire che l'MRP è molto utilizzata in quanto i procedimenti sono standard e semplici da comprendere. Analizziamo le funzionalità dell'applicazione singolarmente così da poter evidenziarne punti di forza e punti di debolezza.

- *Ricerca dei prodotti:* l'utente può cercare le informazioni relative ad un prodotto selezionandolo dalla combo box. È un'operazione semplice che non richiede alcuna particolare competenza dell'utente. Ciò che potrebbe apparire tedioso è di per sé l'ispezione del menu visto l'elevato numero di prodotti. Per ridurre in parte tale fastidio i prodotti sono presentati in ordine alfanumerica secondo l'attributo "ProdottoID".

A lavoro terminato la scelta più congeniale sarebbe stata quella associare alla combo box un campo testuale.

Selezionato un prodotto, nell'area testuale vengono fornite quindi le informazioni ad esso associate.

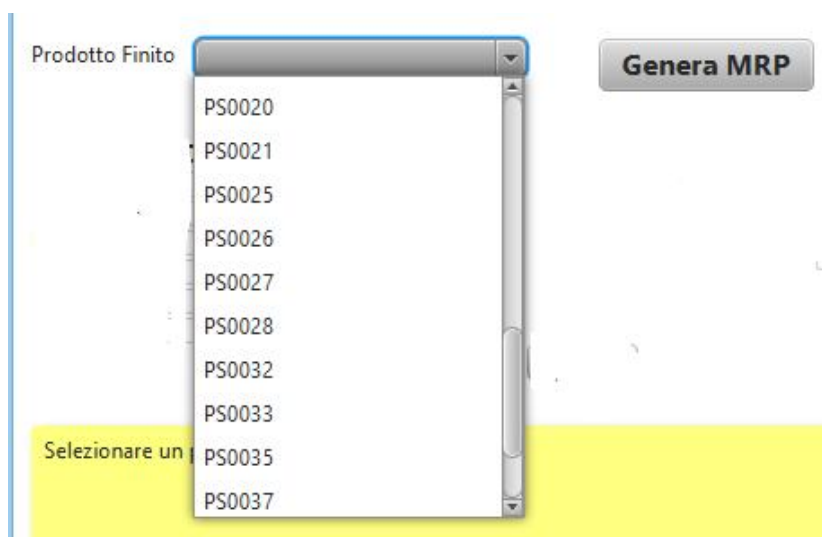
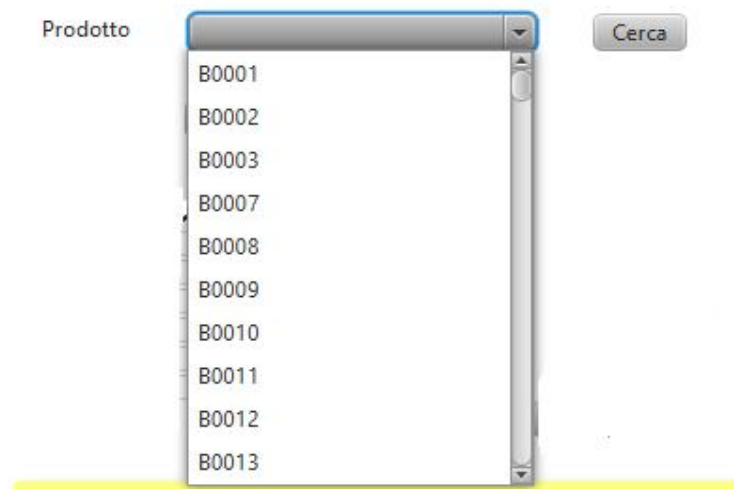
- *Generazione MRP:* l'insieme di prodotti che l'utente può selezionare in questo caso è ridotto rispetto a quello precedente in quanto l'algoritmo per sua stessa definizione deve essere eseguito a partire da prodotti finiti.

L'area testuale presenta i risultati della pianificazione che possono essere compresi e interpretati da un utente che già conosce la notazione tipica di questa tecnica.

Si fa presente che rispetto al classico algoritmo non sono stati considerati eventuali ordini in

arrivo e quindi relativi aggiustamenti a causa di mancanza di dati.

Ciò non ha comunque alcun impatto sull'efficienza dell'operazione giacché si tratta di aggiungere o sottrarre al magazzino disponibile delle quantità.



Si richiede quindi una minima "istruzione" dell'utente sulle tabelle di MRP: è sicuramente un limite dell'applicazione in quanto non è "accessibile" a tutti. Le tabelle riportate nell'area testuale sono tutte quelle relative ai prodotti per cui la pianificazione era necessaria. Laddove infatti non emergono fabbisogni l'algoritmo si arresta e non procede alla stampa di tabelle vuote.

- Inserimento di un nuovo prodotto:
l'utente può inserire dei nuovi prodotti riportandone le informazioni richieste. Può aggiungere indistintamente prodotti finiti, componenti acquistati, fabbricati e altro.

NUOVO PRODOTTO

Codice	<input type="text"/>
Descrizione	<input type="text"/>
LotSize	<input type="text"/>
LeadTime	<input type="text"/>
Domanda	<input type="text"/>
Valore	<input type="text"/>

Inserisci

- Creazione distinta base: l'utente può inserire le relazioni tra i diversi materiali purché questi siano già presenti nel database.

Distinta base del nuovo prodotto

Prodotto	<input type="text"/>
Componente	<input type="text"/>
Quantità	<input type="text"/>

Aggiungi

L'applicazione realizzata offre quindi la possibilità di calcolare i fabbisogni in modo abbastanza semplice tuttavia bisogna tener presente che la pianificazione di ogni prodotto è indipendente dalla pianificazione degli altri prodotti. Chiariamo il concetto con un esempio: siano P ed S due prodotti, sia P costituito da B e C, sia S costituito da C e D; dalla pianificazione di P trovo i fabbisogni di C necessari alla sola produzione di P, analogamente vale per la pianificazione di S. Volendo trovare il fabbisogno totale di B dovrei creare delle dipendenze anche tra S e P e ripianificare il tutto. Si riportano di seguito gli estratti di alcune pianificazioni.

PS0011						CA0039					
0.0	200.0	200.0	300.0	200.0	0.0	0.0	200.0	200.0	300.0	200.0	0.0
40.0	20.0	0.0	0.0	40.0	40.0	40.0	20.0	0.0	0.0	40.0	40.0
0.0	160.0	180.0	300.0	200.0	0.0	0.0	160.0	180.0	300.0	200.0	0.0
0.0	180.0	180.0	300.0	240.0	0.0	0.0	180.0	180.0	300.0	240.0	0.0
0.0	180.0	300.0	240.0	0.0	0.0	0.0	180.0	300.0	240.0	0.0	0.0
R0014						CA0043					
0.0	200.0	200.0	300.0	200.0	0.0	0.0	200.0	200.0	300.0	200.0	0.0
40.0	20.0	0.0	0.0	40.0	40.0	40.0	20.0	0.0	0.0	40.0	40.0
0.0	160.0	180.0	300.0	200.0	0.0	0.0	160.0	180.0	300.0	200.0	0.0
0.0	180.0	180.0	300.0	240.0	0.0	0.0	180.0	180.0	300.0	240.0	0.0
0.0	180.0	300.0	240.0	0.0	0.0	0.0	180.0	300.0	240.0	0.0	0.0

Pianificazione impossibile per Prodotto finito 7 al tempo 0
Pianificazione impossibile per Prodotto finito 7 al tempo 1
Pianificazione impossibile per Componente acquistato 12 al tempo 1
Pianificazione impossibile per Componente acquistato 10 al tempo 1
Pianificazione impossibile per Componente fabbricato 13 al tempo 1
Pianificazione impossibile per Componente fabbricato 16 al tempo 1
Pianificazione impossibile per Materia prima 5 al tempo 1
Pianificazione impossibile per Componente acquistato 11 al tempo 1
Pianificazione impossibile per Componente fabbricato 25 al tempo 1

Da un'ampia analisi si evince che vi sono molti fabbisogni che non possono essere soddisfatti tra il tempo 0 e il tempo 1. Causa principale di tale esito è il fatto che la quantità di magazzino sia uguale per tutti i materiali: è infatti un'assunzione non realistica poiché vi possono essere grandi consumi di un materiale rispetto ad un altro e l'azienda cerca sempre di avere abbastanza stock da sopperire alle richieste.

Tuttavia l'impossibilità delle pianificazioni è un evento che può tranquillamente verificarsi in un'azienda, soprattutto quando si ha un'enorme crescita delle vendite inattesa.

Per far fronte alla situazione l'azienda può sollecitare la produzione e gli acquisti con straordinari, acquisire il prodotto finito in outsourcing oppure rinunciare all'incremento di vendita.

Un video dimostrativo delle funzionalità dell'applicazione è disponibile al link:

<http://youtu.be/aWabk2vQpHA>



Licenza d'uso

Quest'opera è stata rilasciata con licenza Creative Commons Attribuzione - Non commerciale - Condividi allo stesso modo 4.0 Internazionale. Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-nc-sa/4.0/deed.it>.

Tu sei libero di:

- **Condividere** — riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare questo materiale con qualsiasi mezzo e formato
- **Modificare** — remixare, trasformare il materiale e basarti su di esso per le tue opere
- Il licenziante non può revocare questi diritti fintanto che tu rispetti i termini della licenza.

Ai seguenti termini:

- **Attribuzione** — Devi attribuire adeguatamente la paternità sul materiale, fornire un link alla licenza e indicare se sono state effettuate modifiche. Puoi realizzare questi termini in qualsiasi maniera ragionevolmente possibile, ma non in modo tale da suggerire che il licenziante avalli te o il modo in cui usi il materiale.
- **NonCommerciale** — Non puoi usare il materiale per scopi commerciali.
- **StessaLicenza** — Se remixi, trasformi il materiale o ti basi su di esso, devi distribuire i tuoi contributi con la stessa licenza del materiale originario.
- **Divieto di restrizioni aggiuntive** — Non puoi applicare termini legali o misure tecnologiche che impongano ad altri soggetti dei vincoli giuridici su quanto la licenza consente loro di fare.

