



Università Politecnica delle Marche

Facoltà di Ingegneria

Corso di Laurea in Ingegneria Informatica e dell'Automazione

Progettazione database per software di produzione

Database project for production software

Relatore

Prof. ssa Diamantini Claudia

Candidato

Tigano Costantino

A.A. 2018/2019

Indice

Introduzione.....	1
Capitolo 1: Analisi dei requisiti	
1.1 Requisiti espressi in linguaggio naturale.....	2
1.2 Modulistica e documentazione	4
1.3 Glossario dei termini	6
1.4 Strutturazione dei requisiti	7
1.5 Specifica delle operazioni	8
Capitolo 2: Progettazione concettuale	
2.1 Spiegazione delle modalità con cui si intende procedere... 11	
2.2 Entità principali	11
2.3 Schema scheletro	13
2.4 Sviluppo schema scheletro.....	14
2.5 Dizionario dei dati	18
2.6 Regole aziendali	20
Capitolo 3: Progettazione logica	
3.1 Tavola dei volumi	22
3.2 Tavola delle operazioni	22
3.3 Ristrutturazione schema concettuale	25
3.3.1 Analisi delle ridondanze.....	25
3.3.2 Scelta identificatori principali.....	28
3.4 Traduzione nel modello relazionale.....	28
Capitolo 4: Codifica e testing	
4.1 Definizione dello schema e screen delle tabelle.....	30
4.2 Codifica delle operazioni e screen dell'esecuzione	35
4.3 Operazioni di analisi	47
Capitolo 5: Conclusioni	
5.1 Considerazioni su operazioni di analisi	49
5.2 Considerazioni sulle operazioni	50
5.3 Sviluppi futuri	51
Bibliografia e Sitografia.....	53

Introduzione

Il mio elaborato riguarda la progettazione di un database per la gestione di dati operativi di un'azienda. Nello specifico l'azienda produce dei contenitori automatici per la raccolta differenziata dei rifiuti, tuttavia questo database è applicabile per qualsiasi azienda che produce prodotti di tipo modulato. La progettazione è stata eseguita seguendo il modello di progettazione canonico: Raccolta informazioni e analisi dei requisiti, Progettazione concettuale, Progettazione logica, codifica e testing. Infine è stato eseguito uno studio sulla flessibilità del database riguardo eventuali operazioni aggiuntive e sono state fatte delle considerazioni sulle operazioni di analisi.

Raccolta informazioni e analisi dei requisiti:

Le informazioni sono state raccolte mediante interviste al titolare dell'azienda e il responsabile della produzione. Inoltre sono state fatte delle osservazioni direttamente in fabbrica per meglio comprendere il processo produttivo ed è stata fornita della modulistica come esempio per la rappresentazione dei dati. Infine è stata stilata una strutturazione dei requisiti e la specifica delle operazioni da dover eseguire sul database.

Progettazione concettuale:

La progettazione concettuale è stata eseguita in maniera procedurale, attraverso i costrutti del modello Entity-Relationship è stato costruito uno schema scheletro in cui sono state raccolte le entità e relazioni principali per la descrizione dell'applicazione. Successivamente attraverso una strategia mista (botton-up e top-down) operando per raffinamenti è stato prodotto uno schema E-R che rappresenta totalmente la realtà di interesse.

Progettazione logica:

Nella ristrutturazione dello schema concettuale è stato eseguito lo studio delle ridondanze (in quanto lo schema E-R prodotto è stato reso minimale). Nella traduzione del modello relazionale sono stati scelti gli identificatori principali e successivamente si è elaborata la traduzione delle entità e delle relazioni.

Codifica e testing:

Attraverso l'uso di xampp è stato creato il database in localhost in sql nella piattaforma phpmyadmin. Le operazioni sono state implementate in php creando delle classi i cui metodi richiamano le operazioni stilate nell'analisi dei requisiti.

Conclusione:

E' stata valutata la flessibilità del database su operazioni che eventualmente potranno essere realizzate e altre invece che non sono state racchiuse nel progetto. Infine è stata fatta un'ulteriore

analisi sulla possibilità di implementare un'architettura aggiuntiva per le operazioni di analisi dei dati.

Capitolo 1

Analisi dei requisiti

La raccolta delle informazioni è avvenuta tramite interviste ai protagonisti della base di dati, chiamate telefoniche e osservazioni fatte nell'esecuzione del processo lavorativo. L'analisi dei requisiti che la base di dati deve rispettare parte dall'esposizione di essi espressi in linguaggio naturale.

1.1 Requisiti espressi in linguaggio naturale

La base di dati progettata riguarda la parte operativa dell'azienda per la produzione di contenitori automatici per la raccolta di rifiuti, composti da più componenti da assemblare. Il processo di produzione del prodotto è visto come una serie di lavorazioni eseguite secondo un preciso ordine (stabilendo dunque una precedenza tra di esse). Ai dipendenti è stato concesso assegnarsi più lavorazioni (attraverso l'uso di un tablet) e più dipendenti possono partecipare a una di esse. Le lavorazioni da fare sono dettate dal prodotto in produzione che a sua volta dipende dall'ordine a cui è associato. Dunque la base di dati gestisce i dati dei fornitori, del magazzino (che contiene strumenti e risorse usate nelle lavorazioni) dei clienti, degli ordini, dei prodotti, dei dipendenti e delle lavorazioni.

Riguardo i fornitori è stata indicata la partita iva, denominazione e una modalità di rintracciabilità. Il magazzino invece è composto da tutto ciò che serve per il processo produttivo. In particolare per ogni materia prima, semilavorato, accessorio e attrezzatura sono a disposizione la quantità rimasta e la quantità presente per valutare una nuova fornitura o una nuova produzione.

Riguardo i clienti invece sono state ripercorse la partita iva, nome, e-mail e numero telefonico.

Gli ordini sono stati registrati dal coordinatore della produzione e ad essi sono stati associati i prodotti selezionati dal cliente. Inoltre è possibile visionare lo stato di produzione di ciascun ordine di un determinato cliente.

I prodotti dunque sono legati all'ordine e sono individuati da una matricola in modo da poter distinguere due prodotti uguali nello stesso o differente ordine.

I dipendenti invece hanno a disposizione username e password per identificarsi, a essi sono associate le competenze e le lavorazioni a cui partecipano. Si distinguono in operai e amministratore.

Le lavorazioni sono alla base del processo produttivo, a ciascuna di esse sono state specificate le materie prime, i semilavorati, accessori e attrezzature; osservando che una lavorazione può anche produrre un semilavorato o un accessorio. Inoltre per ciascuna lavorazione si conosce un codice di priorità (che detta la sequenza per la produzione), il numero di dipendenti necessari e la categoria di appartenenza. Infine sotto richiesta del titolare ad per ogni lavorazione si deve conoscere il tempo impiegato per eseguirla in modo da poterla confrontare con una durata di lavorazione scelta di default ai fini di una valutazione dell'efficienza dell'operato. Infine è stato previsto che il ciclo di vita della base di dati è di un anno.

1.2 Modulistica e documentazione

Sotto richiesta il coordinatore della produzione ha fornito un file per vedere la strutturazione della produzione. Vedi figura 1.

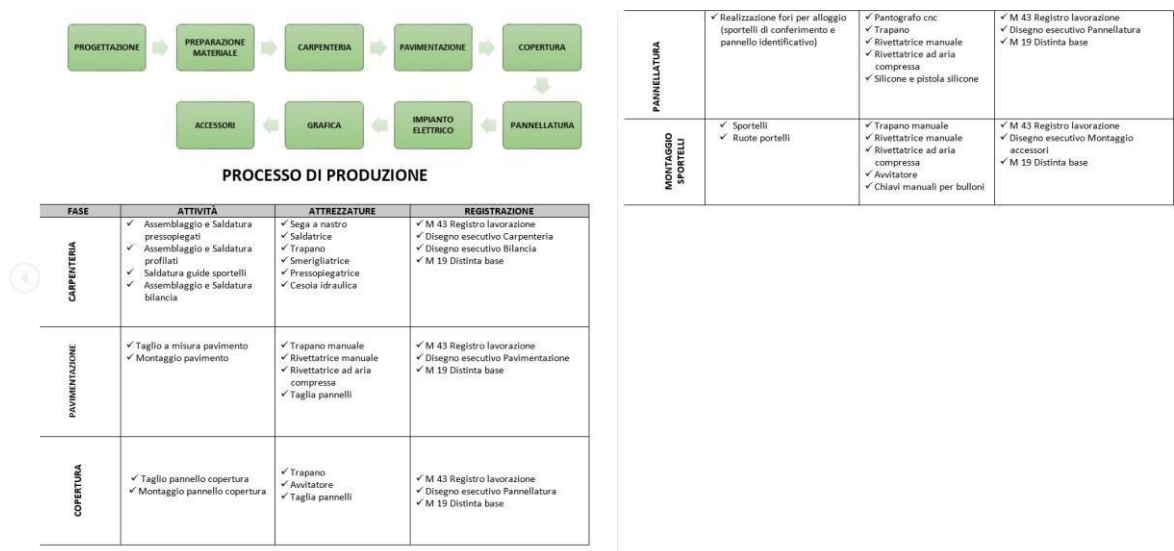


Fig. 1.1

Inoltre è stato fornito un esempio di preventivo per meglio notare la composizione del prodotto.

Ecoisola a 6. moduli monolato configurata per :

n° 6 conferimenti per contenitori da 1100 lt

Accessori:

- Sistema di gestione
- Lettore ottico qr code
- Bocche di conferimento con apertura simultanea
- Sensori rilevamento livello riempimento contenitori
- Impianto di sanificazione
- Grafica personalizzata
- Portellone di apertura elettromeccanica frontale
- Sensori di identificazione apertura bocche di conferimento per singolo sportello
- Sensori si pressione su portelli di apertura bocche di conferimento
- N° 4 punti di aggancio in copertura per spostamento ecoisola mediante gru e/o muletto

cod.	componenti	configurazione				
		q.tà				
M6100F	6 moduli monolato mod. 1100 con portellone frontale	1				
AEMF	Portellone frontale aggiuntivo 1100	1				
SP2	Sportello rifiuto light con apertura simultanea da due sportelli in poi	6				

Fig. 1.2

GEST1	sistema di gestione		1				
PI2	Display - lettore ottico qr code		1				
A99	Impianto di alimentazione elettrica a rete 220		1				
A104D	Finitura con inserimento di foto fornite dal committente		1				
A106	impianto di sanificazione		1				
			totale netto				18.300,00
	nella fornitura sono esclusi i contenitori per rifiuti interni alle ecoisole						

OPTIONAL:

cod.	componenti	configurazione				
		q.tà				netto
A100	alimentazione con fotovoltaico da 250V e n°2 batterie di accumulo	1				1.250,00
A102	videosorveglianza	1				1.000,00
A108	illuminazione a led per singolo sportello	6				330,00

Sono esclusi:

trasporto – incidenza 4% per isole 6%

messa in uso incidenza 2%

iva in base alle aliquote in vigore alla data della fatturazione

Validità offerta : 60 gg.

Fig. 1.3

1.3 Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Ecoisola	E' il prodotto effettivo da vendere. Esso è composto da più componenti e dunque per produrlo sono richieste più lavorazioni.	Prodotto	Ordine, lavorazione.
Lavorazione	E' il passo base per la produzione del prodotto. I dipendenti possono assegnarsele per poterle eseguire e mandare avanti il processo produttivo.	Operazione	Materie prime, semilavorato, dipendenti, prodotto.
Dipendente	Sono sia gli operai che l'amministratore. Entrambi eseguono lavorazioni e andranno valutati riguardo l'efficienza dell'operato.	Lavoratori	Lavorazione, composizione prodotto.
Magazzino	Per l'applicazione il magazzino è una semplice raccolta di strumenti utili per la produzione.	Deposito	Fornitore, produzione
Stato	Lo stato di produzione serve per stabilire in che fase della produzione si trova un determinato prodotto.		Cliente, prodotto.
Competenza	Le competenze sono delle capacità degli operai usate come prerequisito per potersi assegnare una lavorazione.	Capacità	Dipendente, lavorazione.
Materia prima	E' in grosso modo il significato comune, serve per produrre semilavorati.		Magazzino, lavorazione.
Semilavorato	Sono tutti quei pezzi che poi andranno ad assemblare il prodotto finale.		Lavorazione.

Accessorio	Particolari pezzi per la composizione del prodotto. Sono quei pezzi che possono differire da prodotto a prodotto.	Optional	Lavorazione.
Categoria	Le lavorazioni sono raggruppate per tipo di lavorazione, carpenteria, pavimentazione ecc...		Lavorazione.

1.4 Strutturazione dei requisiti

Frase relative ai fornitori

Riguardo i fornitori si deve conoscere la partita iva, denominazione e una modalità di rintracciabilità.

Frase relative al magazzino

Il magazzino invece è composto da tutto ciò che serve per il processo produttivo. In particolare per ogni materia prima e per ogni semilavorato bisogna sapere la quantità rimasta per valutare una nuova fornitura.

Frase relative ai clienti

Riguardo i clienti invece si reperiscono la partita iva, nome, e-mail e numero telefonico.

Frase relative agli ordini

Gli ordini saranno registrati dal coordinatore della produzione e ad essi saranno associati i prodotti selezionati dal cliente. Inoltre è possibile visionare lo stato di produzione di ciascun ordine di un determinato cliente.

Frase relative al prodotto

I prodotti dunque sono legati all'ordine e sono individuati da una matricola in modo da poter distinguere due prodotti uguali nello stesso o differente ordine.

Frase relative ai dipendenti

I dipendenti invece hanno a disposizione username e password per identificarsi, a essi sono associate le competenze e le lavorazioni a cui partecipano. Si distinguono in operai o amministratori.

Frase relative alle lavorazioni

Le lavorazioni dunque sono alla base del processo produttivo, a ciascuna di esse sono specificate le materie prime, i semilavorati, attrezzature e accessori; osservando che una lavorazione può anche produrre un semilavorato o un accessorio. Inoltre per ciascuna lavorazione si conosce un codice di priorità (che detta la sequenza per la produzione), il numero di dipendenti necessari e la categoria di appartenenza. Infine sotto richiesta del titolare ad per ogni lavorazione si deve conoscere il tempo impiegato per eseguirla in modo da poterla confrontare con una durata di lavorazione scelta di default; ai fini di una sintesi dei dati per valutare l'efficienza dell'operato.

1.5 Specifica delle operazioni

1. Registra accessorio (Circa 20 volte l'anno)
2. Registra semilavorato (Circa 1000 volte l'anno)
3. Registra modello prodotto (Circa 200 volte l'anno)
4. Registra materia prima (Circa 300 volte l'anno)
5. Registra fornitore (Circa 40 volte l'anno)
6. Registra lavorazione (Circa 100 volte l'anno)
7. Registra attrezzatura (Circa 60 volte l'anno)
8. Registra prodotto (Circa 300 volte l'anno)
9. Registra ordine (Circa 100 volte l'anno)
10. Registra competenza (Circa 25 volte l'anno)
11. Registra dipendente (Circa 1300 volte l'anno)
12. Registra cliente (Circa 400 volte l'anno)
13. Registra work (Circa 10 volte l'anno)
14. Registra fornitura (Circa 15 volte l'anno)
15. Registra uso materie prime (Circa 20 volte l'anno)
16. Registra uso attrezzatura (Circa 10 volte l'anno)
17. Registra ingresso (Circa 2 volte l'anno)
18. Registra precedenza (Circa 10 volte l'anno)
19. Registra composizione (Circa 2 volte l'anno)
20. Registra assegnamento competenze (Circa 10 volte l'anno)
21. Registra possidenza di competenze (Circa 15 volte l'anno)
22. Registra partecipazione a work (Circa 2 volte l'anno)
23. Registra aggiunta accessori (Circa 2 volte l'anno)
24. Modifica accessorio (Circa 2 volte l'anno)
25. Modifica semilavorato (Circa 2000 volte l'anno)
26. Modifica modello prodotto (Circa 20 volte l'anno)
27. Modifica materia prima (Circa 2 volte l'anno)
28. Modifica fornitore (Circa 2 volte l'anno)
29. Modifica lavorazione (Circa 10 volte l'anno)
30. Modifica attrezzatura (Circa 2 volte l'anno)

31. Modifica prodotto (Circa 10 volte l'anno)
32. Modifica ordine (Circa 15 volte l'anno)
33. Modifica competenza (Circa 2 volte l'anno)
34. Modifica dipendente (Circa 2 volte l'anno)
35. Modifica cliente (Circa 2 volte l'anno)
36. Modifica work (Circa 2000 volte l'anno)
37. Modifica fornitura (Circa 20 volte l'anno)
38. Modifica uso di materie prime (Circa 20 volte l'anno)
39. Modifica uso di attrezzature (Circa 2 volte l'anno)
40. Modifica ingresso (Circa 10 volte l'anno)
41. Modifica precedenza (Circa 5 volte l'anno)
42. Modifica composizione (Circa 10 volte l'anno)
43. Modifica assegnazione di competenze (Circa 5 volte l'anno)
44. Modifica possidenza di competenze (Circa 5 volte l'anno)
45. Modifica partecipazione a work (Circa 1200 volte l'anno)
46. Modifica aggiunta di accessori (Circa 100 volte l'anno)
47. Elimina accessorio (Circa 5 volte l'anno)
48. Elimina semilavorato (Circa 5 volte l'anno)
49. Elimina modello prodotto (Circa 5 volte l'anno)
50. Elimina materia prima (Circa 2 volte l'anno)
51. Elimina fornitore (Circa 2 volte l'anno)
52. Elimina lavorazione (Circa 5 volte l'anno)
53. Elimina attrezzatura (Circa 2 volte l'anno)
54. Elimina prodotto (Circa 900 volte l'anno)
55. Elimina ordine (Circa 1700 volte l'anno)
56. Elimina competenza (Circa 2 volte l'anno)
57. Elimina dipendente (Circa 2 volte l'anno)
58. Elimina cliente (Circa 2 volte l'anno)
59. Elimina work (Circa 800 volte l'anno)
60. Elimina fornitura (Circa 100 volte l'anno)
61. Elimina uso materia prima (Circa 5 volte l'anno)
62. Elimina uso attrezzatura (Circa 5 volte l'anno)
63. Elimina ingresso (Circa 5 volte l'anno)
64. Elimina precedenza (Circa 5 volte l'anno)
65. Elimina composizione (Circa 5 volte l'anno)
66. Elimina assegnazione di competenze (Circa 4 volte l'anno)
67. Elimina possidenza di competenze (Circa 2 volte l'anno)
68. Elimina partecipazione a work (Circa 1000 volte l'anno)
69. Elimina aggiunta di accessori (Circa 800 volte l'anno)
70. Visualizza accessori (Circa 800 volte l'anno)
71. Visualizza aggiunta di accessori (Circa 800 volte l'anno)
72. Visualizza attrezzature (Circa 30 volte l'anno)
73. Visualizza clienti (Circa 50 volte l'anno)
74. Visualizza competenze (Circa 50 volte l'anno)
75. Visualizza composizioni (Circa 150 volte l'anno)
76. Visualizza dipendenti (Circa 50 volte l'anno)
77. Visualizza fornitori (Circa 20 volte l'anno)

78. Visualizza forniture (Circa 50 volte l'anno)
79. Visualizza ingressi (Circa 20 volte l'anno)
80. Visualizza lavorazioni (Circa 100 volte l'anno)
81. Visualizza materie prime (Circa 20 volte l'anno)
82. Visualizza modelli prodotti (Circa 20 volte l'anno)
83. Visualizza ordini (Circa 50 volte l'anno)
84. Visualizza partecipazioni a work (Circa 50 volte l'anno)
85. Visualizza possidenze di competenze (Circa 50 volte l'anno)
86. Visualizza precedenze (Circa 20 volte l'anno)
87. Visualizza prodotti (Circa 50 volte l'anno)
88. Visualizza semilavorati (Circa 50 volte l'anno)
89. Visualizza usi di attrezzature (Circa 50 volte l'anno)
90. Visualizza usi di materie prime (Circa 50 volte l'anno)
91. Visualizza works (Circa 100 volte l'anno)
92. Visualizza forniture di un fornitore (Circa 20 volte l'anno)
93. Visualizza forniture di una materia prima (Circa 20 volte l'anno)
94. Visualizza gli attrezzi usati da una lavorazione (Circa 20 volte l'anno)
95. Visualizza le materie prime usate in lavorazione (Circa 50 volte l'anno)
96. Visualizza semilavorati per una lavorazione (Circa 50 volte l'anno)
97. Visualizza le lavorazioni di un modello prodotto (Circa 50 volte l'anno)
98. Visualizza gli accessori di un prodotto (Circa 80 volte l'anno)
99. Visualizza i prodotti in base alla data di produzione (Circa 60 volte l'anno)
100. Visualizza gli ordini in base alla data di scadenza (Circa 80 volte l'anno)
101. Visualizza gli ordini di un cliente (Circa 60 volte l'anno)
102. Visualizza le competenze necessarie per una lavorazione (Circa 80 volte l'anno)
103. Visualizza i dipendenti che hanno una competenza (Circa 50 volte l'anno)
104. Visualizza i dipendenti che partecipano ad un work (Circa 200 volte l'anno)
105. Dettaglio fornitore (Circa 20 volte l'anno)
106. Dettaglio forniture (Circa 50 volte l'anno)
107. Dettaglio materia prima (Circa 20 volte l'anno)
108. Dettaglio attrezzature (Circa 20 volte l'anno)
109. Dettaglio uso di attrezzi (Circa 20 volte l'anno)
110. Dettaglio semilavorato (Circa 20 volte l'anno)
111. Controllo di registro work (Circa 2500 volte l'anno)
112. Controllo numero dipendenti (Circa 2500 volte l'anno)
113. Calcola durata effettiva (Circa 1000 volte l'anno)
114. Visualizzazione lavorazioni da eseguire a schermo (100 al giorno)
115. Visualizzazione lavorazioni consigliate a schermo (100 al giorno)

Capitolo 2

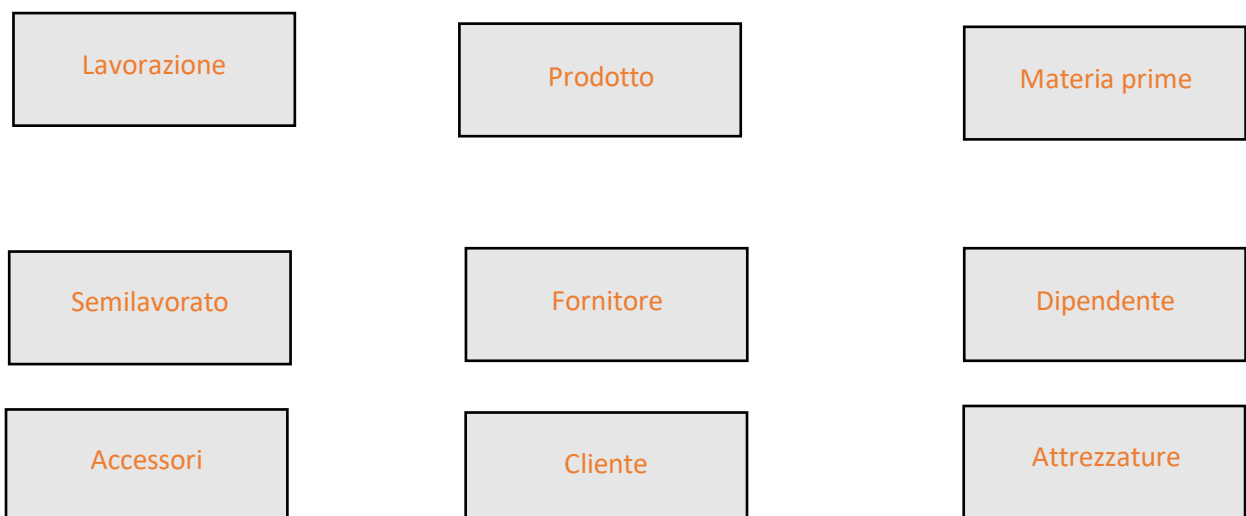
Progettazione Concettuale

2.1 Spiegazione della modalità con cui si intende procedere

Dall'analisi dei requisiti, dalla strutturazione degli stessi e dalla specifica delle operazioni sono state comprese le dinamiche che la base di dati deve avere. Data la complessità della progettazione è stata usata una tecnica mista di che sfrutta sia la strategia top-down che la botton-up. Di seguito sono mostrate le fasi come descrizione della progettazione.

1. Grazie all'analisi dei requisiti è stata ottenuta una visione globale del contesto e ciò ha permesso di individuare le entità principali.
2. Successivamente è stato costruito uno schema scheletro per individuare le prime relazioni principali.
3. Lo schema scheletro è stato diviso in più sottoschemi e utilizzando primitive di raffinamento si è arrivati ad una descrizione dettagliata di ciascun sottoschema.
4. Infine i vari sottoschemi sono stati integrati ottenendo così uno schema finale.

2.2 Entità principali



Con **lavorazione** si intende l'unità base del processo produttivo. Una sequenza di esse compone il prodotto finale.

Il **prodotto** è il risultato delle lavorazioni ed è esso stesso che indica le lavorazioni da fare per realizzarlo. La produzione del prodotto dipende dalla scadenza dell'ordine e dalla possibilità in termini di disponibilità di risorse per poterlo realizzare.

Con **materie prime** si indica tutto ciò che occorre per ciascuna lavorazione. In base alla loro quantità si prevede se dover ordinare una fornitura.

Le **attrezzature** sono tutti gli strumenti che vengono usati nel processo produttivo.

I **semilavorati** sono delle parti di prodotto da realizzare o realizzati da lavorazioni. Difatti un semilavorato può essere sia una risorsa che un'uscita di lavorazione.

Gli **accessori** costituiscono anch'essi delle parti del prodotto che differiscono da prodotto in prodotto.

I **fornitori** sono coloro a cui ci si rivolge per rifornirsi di materie prime e/o accessori.

I **dipendenti** sono coloro che si assegnano lavorazioni per la composizione del prodotto.

I **clienti** sono coloro che confermano l'ordine che manda in produzione un determinato prodotto.

Individuate le entità principali è stato possibile costruire lo schema scheletro come base da cui partire per poi raffinare secondo una strategia mista.

2.3 Schema scheletro

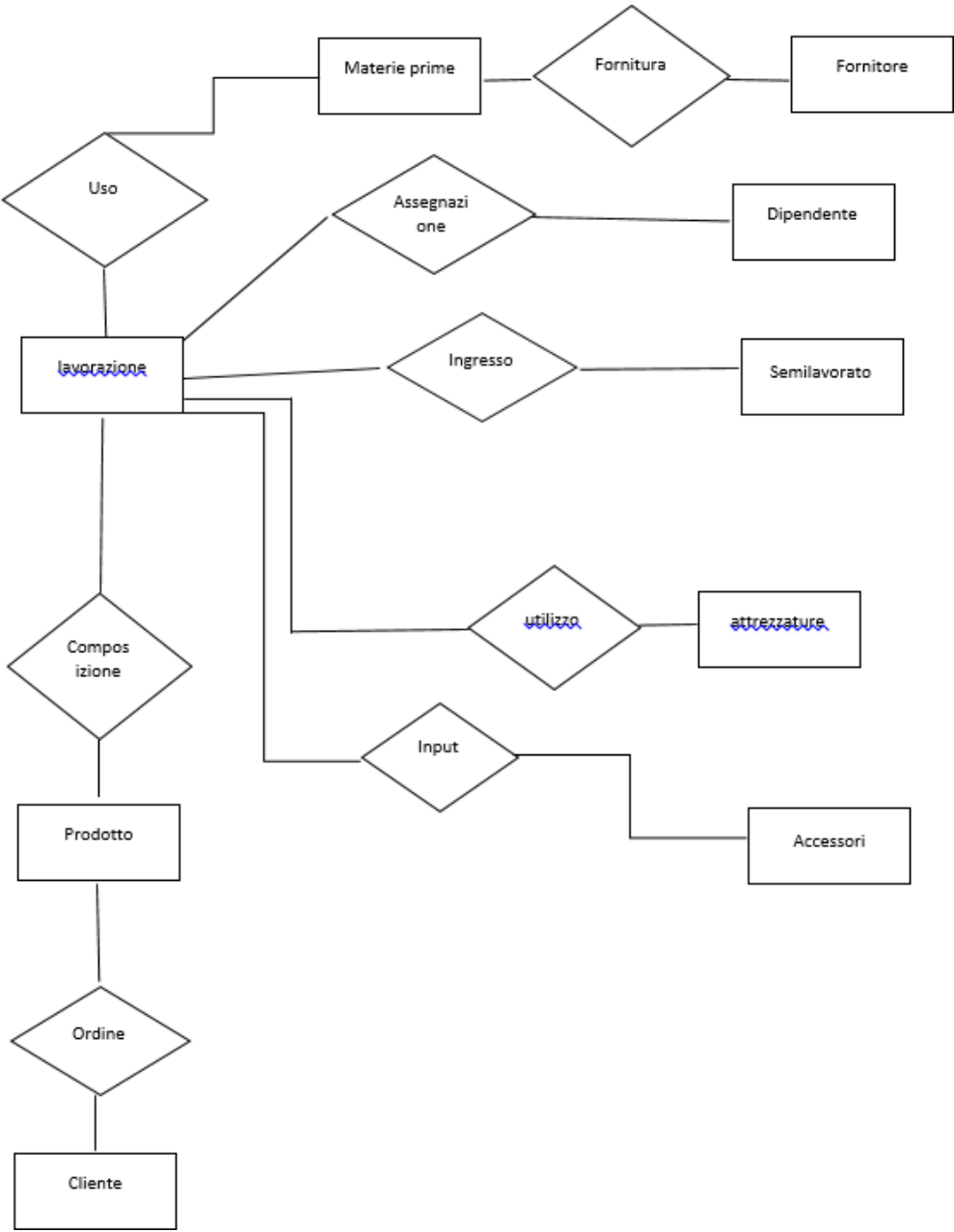


Fig 2.1

2.4 Sviluppo dello schema scheletro

Qui inizia la strategia mista tra top-down e bottom-up. Sviluppo dunque parti di schema partendo dallo schema scheletro.

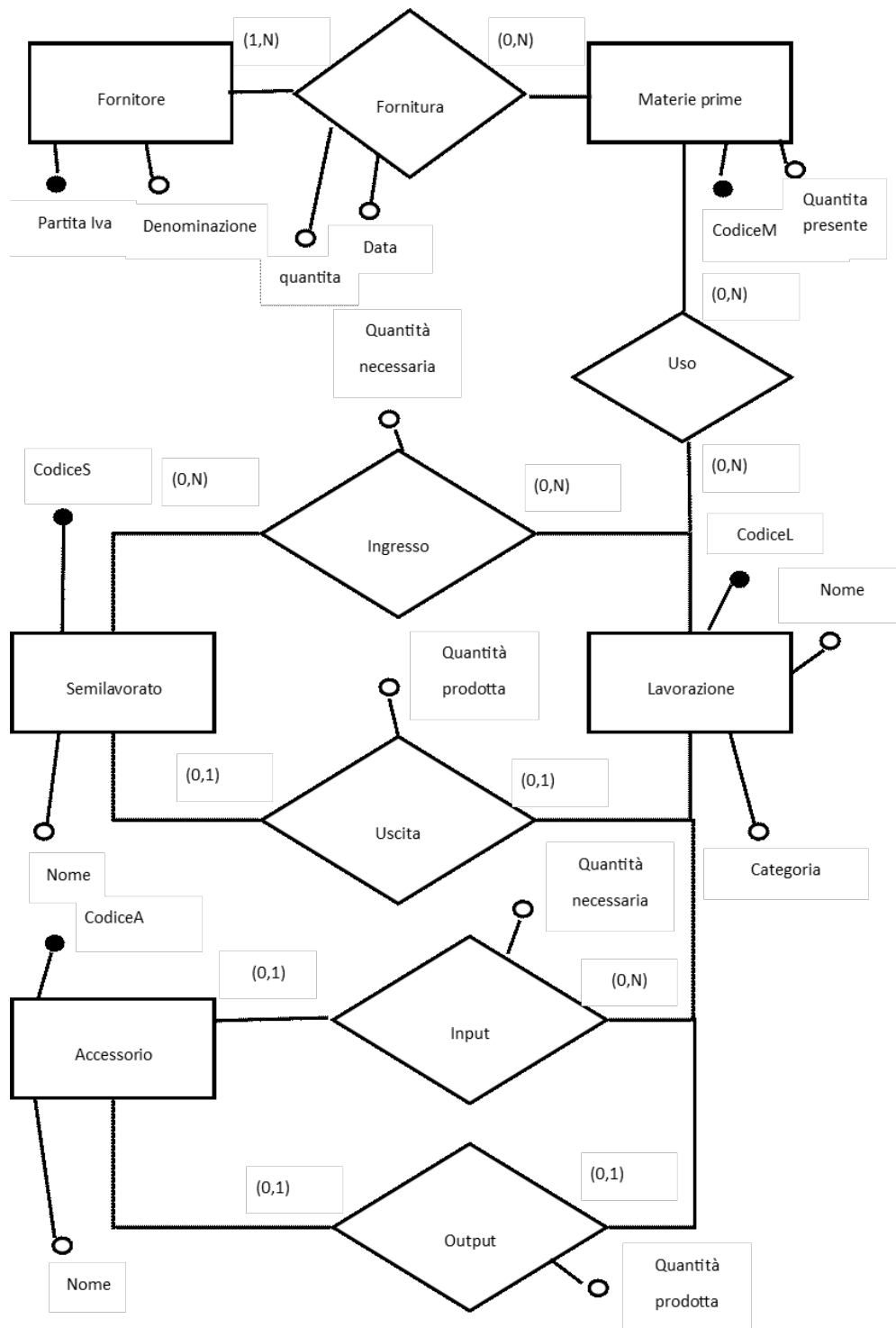


Fig. 2.2

Entità lavorazione

L'entità lavorazione ricopre un ruolo centrale nella base di dati e i suoi attributi sono: Codice, nome, tipo, categoria, durata effettiva. L'attributo categoria serve per inquadrare a quale reparto fa riferimento la lavorazione. Inoltre le lavorazioni possono o meno usare materie prime, attrezzature, semilavorati e accessori. Infine ci sono delle lavorazioni che producono semilavorati che vengono usate per lavorazioni successive. Stesso discorso per gli accessori. Per maggiori informazioni riguardo agli attributi delle entità vedere (2.7 dizionario dei dati).

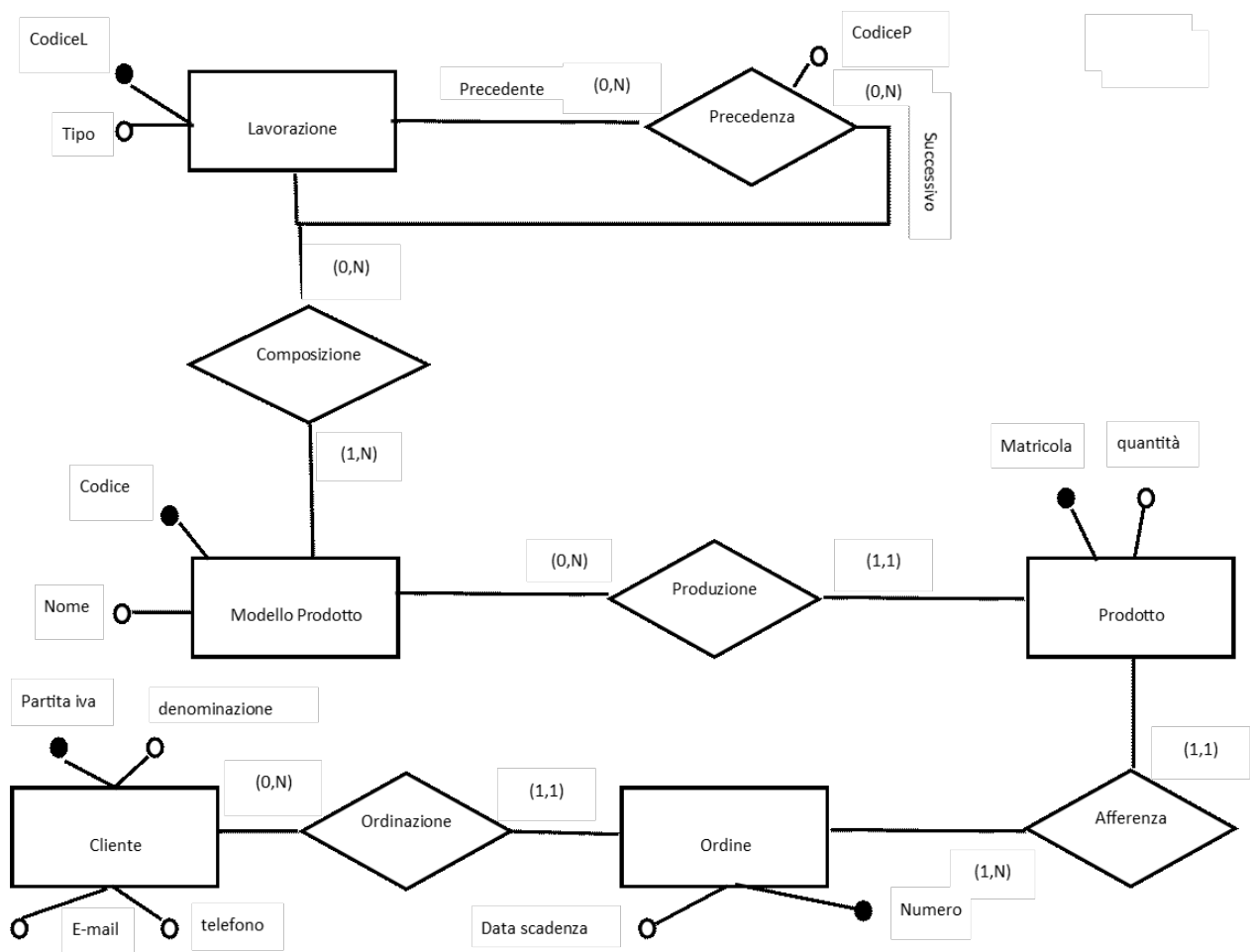


Fig. 2.3

In questo schema si aggiunge l'attributo tipo all'entità lavorazione in quanto ce ne sono alcune che non partecipano a precedenza.

Entità modello prodotto e prodotto

L'entità modello prodotto individua i tipi di prodotti che l'azienda può produrre. Ad ogni modello prodotto si associano le lavorazioni che lo compongono, mentre non si associano quelle lavorazioni

che comprendono gli accessori in quanto essi dipendono dall'ordine. Questa è la sostanziale differenza con l'entità prodotto che invece è legata all'ordine e alla messa in produzione dello stesso.

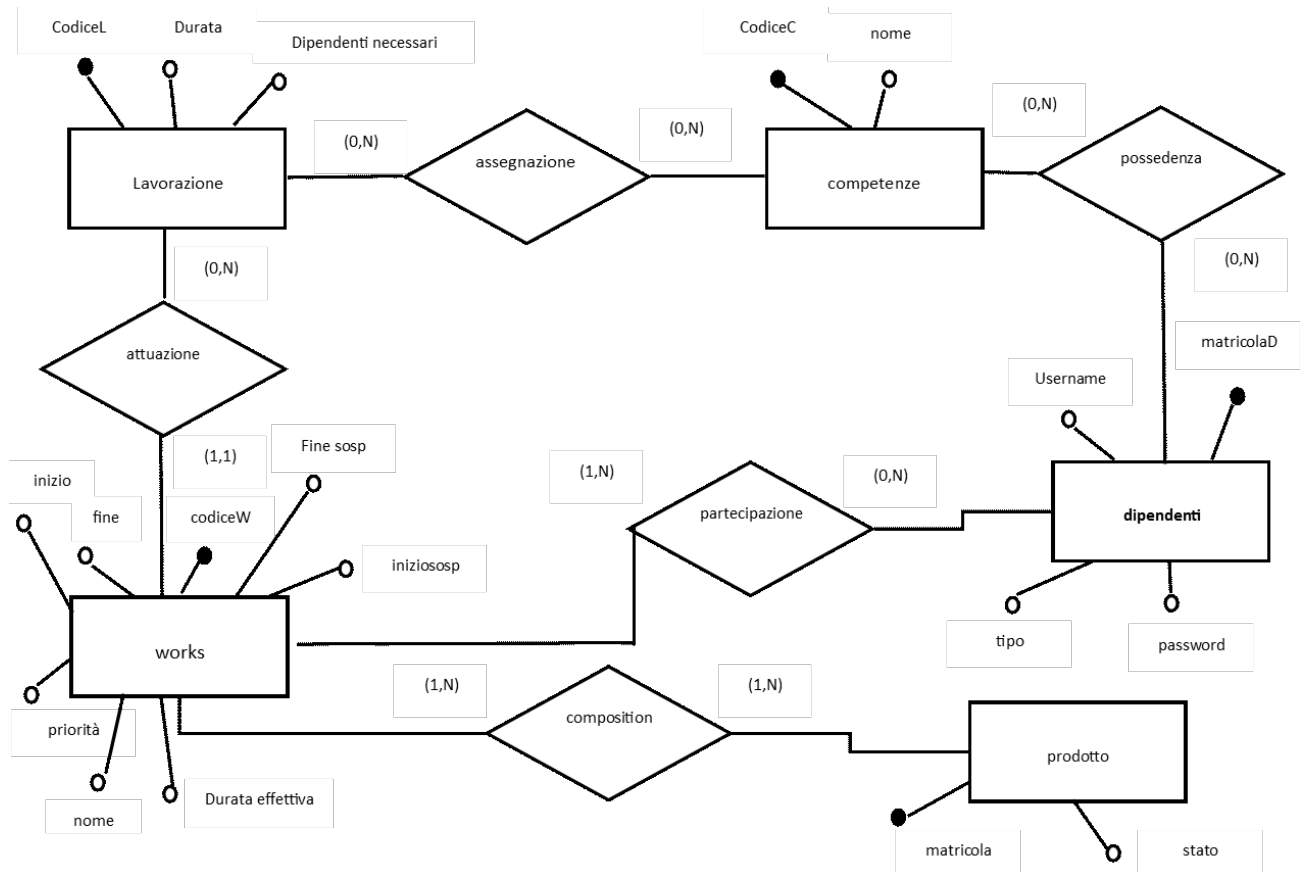


Fig. 2.4

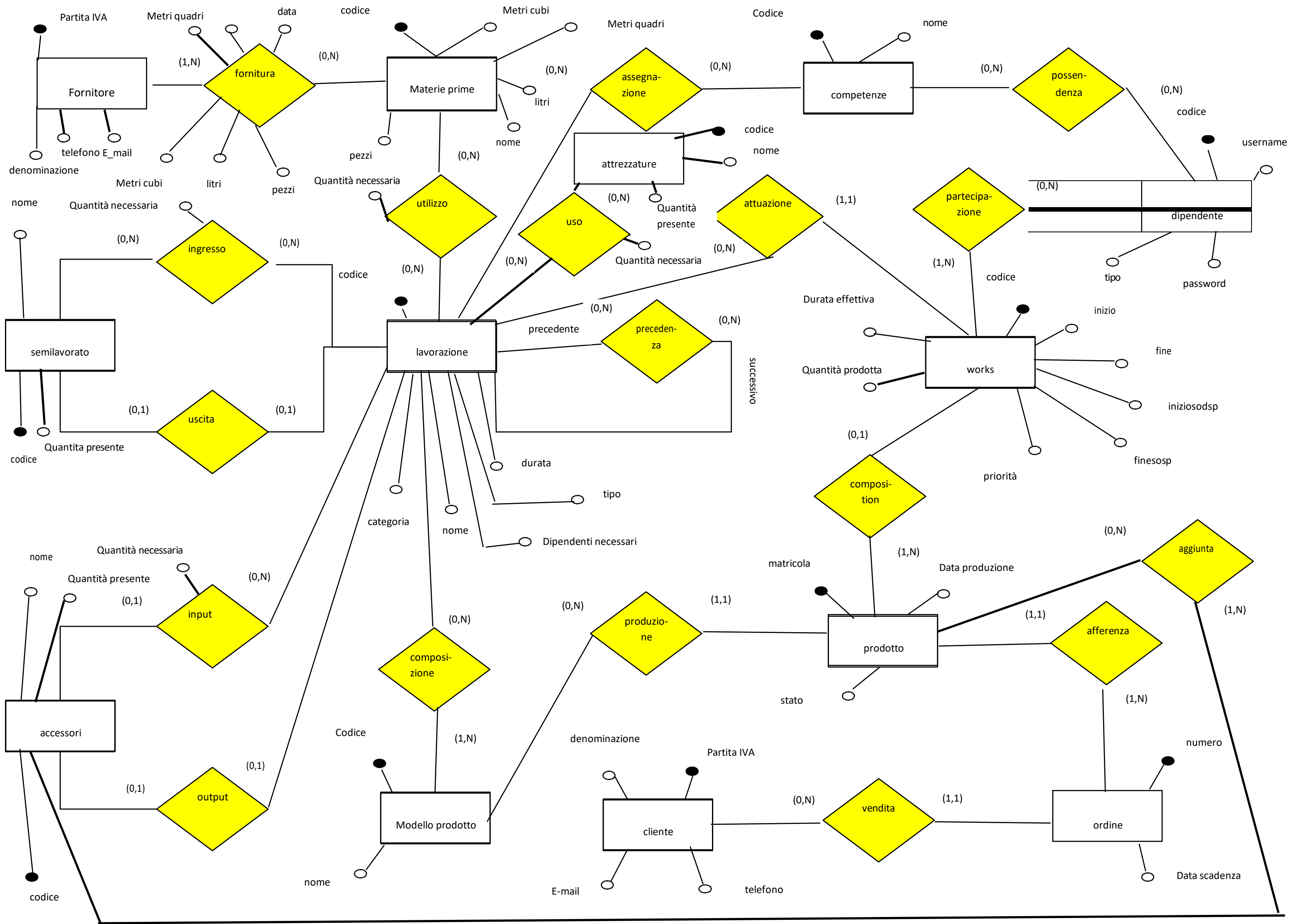
Entità works

I works sono le lavorazioni da fare. Essi sono individuati in base a quali sono i prodotti mandati in produzione. La cardinalità minima da parte di lavorazione è 0 in quanto ci sono lavorazioni che non fanno parte strettamente al processo produttivo. Notiamo inoltre che negli attributi di works compare durata effettiva. Attributo che va confrontato con durata prevista di lavorazione per stimare la qualità dell'operato. Per maggiori informazioni riguardo agli attributi delle entità vedere (2.5 dizionario dei dati).

Questo è il modello E-R ottenuto mediante la strategia botton-up integrando gli schemi analizzati in precedenza.

Non ci sono differenze tra questo schema e i suoi componenti se non l'integrazione di attributi.

Inoltre non sono state inserite né generalizzazioni né attributi multivalore cercando di avere già in questa fase uno schema che sia minimale e più vicino possibile al modello logico dei dati.



2.5 Dizionario dei dati

Entità

Nome entità	Descrizione	Attributi	Identificatore
Fornitore	Ditta che rifornisce le materie prime.	Partita iva, denominazione.	Partita iva.
Materie prime	Tutto ciò che occorre per una lavorazione.	Codice, nome, quantità_presente.	Codice
Lavorazione	Concetto base del processo produttivo.	Codice, nome, durata, tipo, categoria, dipendenti necessari.	Codice.
Semilavorato	Parte di prodotto da assemblare tramite lavorazioni.	Codice, nome.	Codice.
Accessorio	Parte di prodotto da assemblare non collegata al modello.	Codice, nome.	Codice.
Modello prodotto	Prodotto base che identifica le lavorazioni base di un prodotto.	Codice, nome.	Codice.
Attrezzature	Gli attrezzi, i macchinari che sono a disposizione per le lavorazioni e dunque per il processo produttivo.	Codice, nome.	Codice
Prodotto	Prodotto da mandare in produzione, comprende sia lavorazioni di base che aggiuntive legate ad esempio agli accessori.	Matricola, nome.	Matricola, nome, stato.
Ordine	Descrivo ciò che deve entrare in produzione. Stabilisce un primo ordine di lavorazioni in base alla scadenza.	Numero, data scadenza.	Numero.
Cliente	Ditta destinataria dell'ordine e dunque del prodotto in produzione.	Partita iva, denominazione, telefono, e-mail.	Partita iva.
Competenza	Descrivono i requisiti necessari per una singola lavorazione.	Codice, nome.	Codice.
Dipendente	Colui che può assegnarsi lavorazioni,	Codice, tipo, username, password.	Codice.

	è l'utente della base di dati.		
Works	Sono le lavorazioni da fare, le lavorazioni legate al prodotto che è mandato in produzione.	Codice, nome, priorità, inizio, fine, iniziososp, finesosp, durata effettiva.	Codice.

Relazioni

Nome relazione	Descrizione	Entità coinvolte	Attributi
Fornitura	E' sostanzialmente un ordine al fornitore per la gestione delle materie prime.	Fornitore, materie prime.	Quantità presente.
Utilizzo	Esprime l'uso di materie prime da parte delle lavorazioni.	Lavorazione, materie prime.	
Uso	Esprime l'uso di attrezzature associate ad una lavorazione	Attrezzature, lavorazione.	
Precedenza	Esprime la priorità in termini di catena produttiva tra una lavorazione e un'altra.	Lavorazione.	
Input	Collega la lavorazione con gli accessori coinvolti.	Lavorazione, accessorio.	Quantità necessaria.
Output	Esprime il concetto di accessorio prodotto da una lavorazione o assemblato ad esso.	Lavorazione, accessorio.	
Ingresso	Collega la lavorazione con i semilavorati coinvolti.	Lavorazione, semilavorato.	Quantità necessaria.
Uscita	Denota il semilavorato assemblato dalla lavorazione.	Lavorazione, semilavorato.	Quantità prodotta.
Composizione	Individua il modello prodotto come catena di lavorazioni.	Lavorazione, modello prodotto.	
Produzione	Esprime come il prodotto da mandare in produzione sia riconducibile ad un modello.	Prodotto, Modello prodotto.	
Afferenza	Riconduce il prodotto da produrre all'ordine di appartenenza.	Prodotto, Ordine.	

Destinazione	Riconduce l'ordine al cliente.	Ordine, Cliente.	
Assegnazione	Individua le lavorazioni a cui sono assegnate determinate competenze.	Lavorazione, competenze	
Possidenza	Collega la competenze ai dipendenti.	Dipendenti, competenze.	
Attuazione	Esprime la parte dinamica delle lavorazioni. Ovvero nei works saranno presenti le lavorazioni da fare per produrre un determinato prodotto.	Lavorazione, works	
Partecipazione	Associa ai works i dipendenti che vi partecipano	Dipendenti, works.	
Composition	Simile a composizione, tuttavia qui è il prodotto che identifica le lavorazioni da fare.	Prodotto, works.	

2.6 Regole aziendali

Regole di vincolo

R.V 1: Quantità dell'associazione fornitura deve essere maggiore o uguale a 0;

R.V 2: Dipendenti necessari e durata dell'entità lavorazione devono essere interi maggiori o uguali a 0;

R.V.3: Quantità necessaria e quantità prodotta devono essere interi maggiori o uguali a 0;

R.V. 4: Quantità presente delle entità attrezzature, semilavorati, accessori devono essere interi maggiori o uguali a 0;

R.V 5: L'attributo durata dell'entità lavorazione deve essere maggiore o uguale a 0;

R.V 6: L'attributo durata effettiva dell'entità works deve essere un intero maggiore o uguale a 0;

R.V 7: L'attributo priorità dell'entità works deve essere un intero maggiore o uguale a 0;

R.V 8: L'attributo tipo dell'entità dipendente può essere "operaio" o "amministratore";

R.V. 9 L'attributo tipo dell'entità lavorazione può assumere i valori "free", "accessorio", "semilavorato", "materia prima";

R.V.10 L'attributo quantità presente dell'entità Materia Prima deve essere maggiore o uguale a 0.

Regole di derivazione

R.D 1: Gli attributi quantità presente di semilavorato e accessorio può essere calcolato facendo la differenza tra la quantità prodotta di work e la quantità necessaria usata dalle lavorazioni.

Capitolo 3

Progettazione logica

3.1 Tavola dei volumi

Concetto	Tipo	Volume
Fornitore	E	10
Materia prima	E	50
Lavorazione	E	100
Semilavorato	E	30
Accessorio	E	50
Modello prodotto	E	30
Prodotto	E	100
Ordine	E	80
Cliente	E	15
Competenza	E	10
Dipendente	E	20
Works	E	2000
Fornitura	R	40
Utilizzo	R	80
Input	R	60
Output	R	50
Ingresso	R	70
Uscita	R	60
Composizione	R	600
Produzione	R	800
Afferenza	R	1000
Destinazione	R	25
Precedenza	R	60
Assegnazione	R	70
Possidenza	R	10
Attuazione	R	500
Partecipazione	R	400
Composition	R	2000

3.2 Tavola delle operazioni

Operazione	Frequenza
Registra accessorio	Circa 50 volte l'anno
Registra semilavorato	Circa 90 volte l'anno
Registra modello prodotto	Circa 100 volte l'anno
Registra materia prima	Circa 60 volte l'anno

Registra fornitore	Circa 15 volte l'anno
Registra lavorazione	Circa 100 volte l'anno
Registra attrezzatura	Circa 50 volte l'anno
Registra prodotto	Circa 150 volte l'anno
Registra ordine	Circa 80 volte l'anno
Registra competenza	Circa 10 volte l'anno
Registra dipendente	Circa 30 volte l'anno
Registra cliente	Circa 20 volte l'anno
Registra work	Circa 1000 volte l'anno
Registra fornitura	Circa 200 volte l'anno
Registra uso materie prime	Circa 300 volte l'anno
Registra uso attrezzatura	Circa 40 volte l'anno
Registra ingresso	Circa 100 volte l'anno
Registra precedenza	Circa 60 volte l'anno
Registra composizione	Circa 300 volte l'anno
Registra assegnamento competenze	Circa 100 volte l'anno
Registra possidenza di competenze	Circa 25 volte l'anno
Registra partecipazione a work	Circa 1300 volte l'anno
Registra aggiunta accessori	Circa 400 volte l'anno
Modifica accessorio	Circa 10 volte l'anno
Modifica semilavorato	Circa 15 volte l'anno
Modifica modello prodotto	Circa 20 volte l'anno
Modifica materia prima	Circa 10 volte l'anno
Modifica fornitore	Circa 2 volte l'anno
Modifica lavorazione	Circa 10 volte l'anno
Modifica attrezzatura	Circa 2 volte l'anno
Modifica prodotto	Circa 10 volte l'anno
Modifica ordine	Circa 15 volte l'anno
Modifica competenza	Circa 2 volte l'anno
Modifica dipendente	Circa 2 volte l'anno
Modifica cliente	Circa 2 volte l'anno
Modifica work	Circa 2000 volte l'anno
Modifica fornitura	Circa 20 volte l'anno
Modifica uso di materie prime	Circa 2000 volte l'anno
Modifica uso di attrezzature	Circa 2 volte l'anno
Modifica ingresso	Circa 10 volte l'anno
Modifica precedenza	Circa 5 volte l'anno
Modifica composizione	Circa 10 volte l'anno
Modifica assegnazione di competenze	Circa 5 volte l'anno
Modifica possidenza di competenze	Circa 5 volte l'anno
Modifica partecipazione a work	Circa 1200 volte l'anno
Modifica aggiunta di accessori	Circa 100 volte l'anno
Elimina accessorio	Circa 5 volte l'anno
Elimina semilavorato	Circa 5 volte l'anno
Elimina modello prodotto	Circa 5 volte l'anno
Elimina materia prima	Circa 2 volte l'anno
Elimina fornitore	Circa 2 volte l'anno
Elimina lavorazione	Circa 5 volte l'anno
Elimina attrezzatura	Circa 2 volte l'anno

Elimina prodotto	Circa 900 volte l'anno
Elimina ordine	Circa 1700 volte l'anno
Elimina competenza	Circa 2 volte l'anno
Elimina dipendente	Circa 2 volte l'anno
Elimina cliente	Circa 2 volte l'anno
Elimina work	Circa 800 volte l'anno
Elimina fornitura	Circa 100 volte l'anno
Elimina uso materia prima	Circa 5 volte l'anno
Elimina uso attrezzatura	Circa 5 volte l'anno
Elimina ingresso	Circa 5 volte l'anno
Elimina precedenza	Circa 5 volte l'anno
Elimina composizione	Circa 5 volte l'anno
Elimina assegnazione di competenze	Circa 4 volte l'anno
Elimina possidenza di competenze	Circa 2 volte l'anno
Elimina partecipazione a work	Circa 1000 volte l'anno
Elimina aggiunta di accessori	Circa 800 volte l'anno
Visualizza accessori	Circa 800 volte l'anno
Visualizza aggiunta di accessori	Circa 800 volte l'anno
Visualizza attrezzature	Circa 30 volte l'anno
Visualizza clienti	Circa 50 volte l'anno
Visualizza competenze	Circa 50 volte l'anno
Visualizza composizioni	Circa 150 volte l'anno
Visualizza dipendenti	Circa 50 volte l'anno
Visualizza fornitori	Circa 20 volte l'anno
Visualizza forniture	Circa 50 volte l'anno
Visualizza ingressi	Circa 20 volte l'anno
Visualizza lavorazioni	Circa 100 volte l'anno
Visualizza materie prime	Circa 20 volte l'anno
Visualizza modelli prodotti	Circa 20 volte l'anno
Visualizza ordini	Circa 50 volte l'anno
Visualizza partecipazioni a work	Circa 50 volte l'anno
Visualizza possidenze di competenze	Circa 50 volte l'anno
Visualizza precedenze	Circa 20 volte l'anno
Visualizza prodotti	Circa 50 volte l'anno
Visualizza semilavorati	Circa 50 volte l'anno
Visualizza usi di attrezzature	Circa 50 volte l'anno
Visualizza usi di materie prime	Circa 50 volte l'anno
Visualizza works	Circa 100 volte l'anno
Visualizza forniture di un fornitore	Circa 20 volte l'anno
Visualizza forniture di una materia prima	Circa 20 volte l'anno
Visualizza gli attrezzi usati da una lavorazione	Circa 20 volte l'anno
Visualizza le materie prime usate in lavorazione	Circa 50 volte l'anno
Visualizza semilavorati per una lavorazione	Circa 50 volte l'anno
Visualizza le lavorazioni di un modello prodotto	Circa 50 volte l'anno
Visualizza gli accessori di un prodotto	Circa 80 volte l'anno
Visualizza i prodotti in base alla data di produzione	Circa 60 volte l'anno

Visualizza gli ordini in base alla data di scadenza	Circa 80 volte l'anno
Visualizza gli ordini di un cliente	Circa 60 volte l'anno
Visualizza le competenze necessarie per una lavorazione	Circa 80 volte l'anno
Visualizza i dipendenti che hanno una competenza	Circa 50 volte l'anno
Visualizza i dipendenti che partecipano ad un work	Circa 200 volte l'anno
Dettaglio fornitore	Circa 20 volte l'anno
Dettaglio forniture	Circa 50 volte l'anno
Dettaglio materia prima	Circa 20 volte l'anno
Dettaglio attrezzature	Circa 20 volte l'anno
Dettaglio uso di attrezzi	Circa 20 volte l'anno
Dettaglio semilavorato	Circa 20 volte l'anno
Controllo di registro work	Circa 2500 volte l'anno
Controllo numero dipendenti	Circa 2500 volte l'anno
Calcola durata effettiva	Circa 1000 volte l'anno
Visualizza su schermo	Circa 3000 volte l'anno
Visualizza su schermo consigliate	Circa 3000 volte l'anno

3.3 Ristrutturazione schema concettuale

3.3.1. Analisi delle ridondanze

Le ridondanze presenti sono relative agli attributi “quantità presente” delle entità materia prima, accessori, semilavorato, attrezzature. Tali attributi coinvolgono operazioni simili sia come tipo di operazione (Lettura/Scrittura), sia come numero di accessi e anche in termini di frequenza. Dunque lo studio della ridondanza riportato è fatto sull’attributo “quantità presente” dell’entità materia prima considerando l’analogia con gli altri attributi.

Assenza di ridondanza:

L’operazione 109 prevede se sia possibile o meno registrare un work. Dunque è un controllo sulla presenza delle materie prime necessarie per poter eseguire il work.

Operazione 109			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Works	E	1	L
Attuazione	R	1	L
Lavorazione	E	1	L
Utilizzo	R	3	L
Materia prima	E	3	L
Fornitura	R	20	L

Recuperato il codice del work eseguiamo un accesso in lettura in attuazione e in lavorazione poiché la cardinalità massima di work in attuazione è 1. I tre accessi in utilizzo dipendono dal fatto che mediamente per una lavorazione (quindi un work) sono necessarie tre materie prime. Infine stimiamo che per tre materie prime ci siano venti forniture; contando il numero di materie prime da esse si risale al controllo dell'operazione.

L'operazione 81 visualizza tutte le materie prime. In assenza della quantità presente per ciascuna materia prima deve essere calcolata la quantità presente,

Operazione 81			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Materia prima	E	50	L
Fornitura	R	150	L
Materia prima	E	50	S

L'operazione procede eseguendo cinquanta accessi in lettura in quanto le materie prime da visualizzare sono pari al volume dell'entità. Stimando che per ogni materia prima sono previste circa tre forniture esegue 150 accessi in fornitura e infine accede cinquanta volte in materia prima per inserire il dato calcolato.

L'operazione 107 mostra il dettaglio di una materia prima. In assenza della quantità presente tale attributo deve essere calcolato per la specifica occorrenza di materia prima.

Operazione 107			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Materia prima	E	1	L
Fornitura	R	3	L
Materia prima	E	1	S

L'operazione effettua lo stesso percorso dell'operazione 81 ma il numero di accessi è limitato ad una particolare materia prima.

Operazione 14			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Fornitura	R	1	S

Presenza ridondanza:

Operazione 109			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Works	E	1	L
Attuazione	R	1	L
Lavorazione	E	1	L
Utilizzo	R	3	L
Materia prima	E	3	L
Materia prima	E	3	S

Operazione 81			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Materia prima	E	50	L

Operazione 107			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Materia prima	E	1	L

Operazione 14			
CONCETTO	COSTRUTTO	ACCESSI	TIPO
Fornitura	R	1	S
Materia prima	E	3	L
Materia prima	E	3	S

L'operazione 14 è la registra fornitura. Accedo alla relazione fornitura in scrittura per registrarla, stimando mediamente che ogni fornitura registra tre materie prime accedo tre volte in lettura per sapere la quantità presente delle rispettive materie prime e segue l'aggiornamento in scrittura.

Calcoli costi totali

Operazione	Costo	Frequenza(mensile)	Totale
109	29	230	6670
81	300	20	6000
107	6	20	120
14	1	20	20
Totale: 12810			

Operazione	Costo	Frequenza(mensile)	Totale
109	15	230	3450
81	50	20	1000
107	1	20	20
14	11	14	154
Totale: 4624			

Ne consegue che conviene mantenere la ridondanza

3.3.2. Scelta degli identificatori principali

Entità	Identificatore
Fornitore	Partita iva
Materia prima	Codice
Lavorazione	Codice
Semilavorato	Codice
Accessorio	Codice
Modello prodotto	Codice
Prodotto	Matricola
Ordine	Numero
Cliente	Partita iva
Competenze	Codice
Dipendente	Codice
Work	Codice
Attrezzature	Codice

In sostanza non essendoci né gerarchie né attributi multivalore e notando che dallo studio delle ridondanze lo schema E-R non viene modificato, la ristrutturazione dello schema restituisce lo schema E-R iniziale potendo dunque procedere direttamente con la traduzione verso il modello logico (relazionale).

3.4 Traduzione nel modello relazionale

Costrutto	Traduzione
Accessorio	Accessorio (Codice , nome, quantità_necessaria, quantità_presente, lavorazione_input)
Aggiunta	Aggiunta (Codice , prodotto_agg, accessorio_agg)
Assegnazione	Assegnazione (codice, competenza_ass, lavorazione_ass)

Attrezzature	Attrezzature (codice , nome, quantita_presente, categoria)
Cliente	Cliente (partita_iva , denominazione, e_mail, telefono)
Competenze	Competenze (Codice , nome)
Composizione	Composizione (codice , lavorazione_comp, modello_prodotto_comp)
Dipendente	Dipendente (codice , user_name, passw, tipo)
Fornitore	Fornitore (partita_iva , denominazione, e_mail, telefono)
Fornitura	Fornitura (Codice , data, litri, pezzi, metri_cubi, metri_quadri, fornitore, materia_prima)
Ingresso	Ingresso (codice , semilavorato_ingr, lavorazione_ingr, quantita_necessaria)
Lavorazione	Lavorazione (codice , nome, tipo, durata, dipendenti_necessari, semilavorato_uscita, categoria, accessorio_uscita)
Materia prima	Materia prima (codice , nome, quantita presente)
Modello prodotto	Modello_prodotto (codice , nome)
Ordine	Ordine (numero , data_scadenza, cliente)
Partecipazione	Partecipazione (codice, dipendente_part, work_part)
Possedente	Possedente (codice , competenza, dipendente)
Precedenza	Precedenza (codice , lavorazione_precedente, lavorazione_successiva)
Prodotto	Prodotto (matricola , stato, data_produzione, ordine, modello_prodotto)
Semilavorato	Semilavorato (codice , nome, quantita_presente)
Uso	Uso (codice , attrezzatura_uso, lavorazione_uso, quantita_necessaria)
Utilizzo	Utilizzo (codice , materia_prima_utili, lavorazione_utili, quantita_necessaria)
Works	Works (codice , data_ora_inizio, data_ora_fine, data_ora_inizio_sosp, data_ora_fine_sosp, durata_effettiva, quantita_prodotta, lavorazione_W, prodotto_W, priorit�)

Capitolo 4

Codifica e testing

Si riporta lo screenshot dello schema della base di dati e delle tabelle. La codifica delle operazioni invece è stata effettuata tramite metodi di classi implementate in php utile per la realizzazione del software che opererà sul database.

4.1 Definizione dello schema e screen delle tabelle

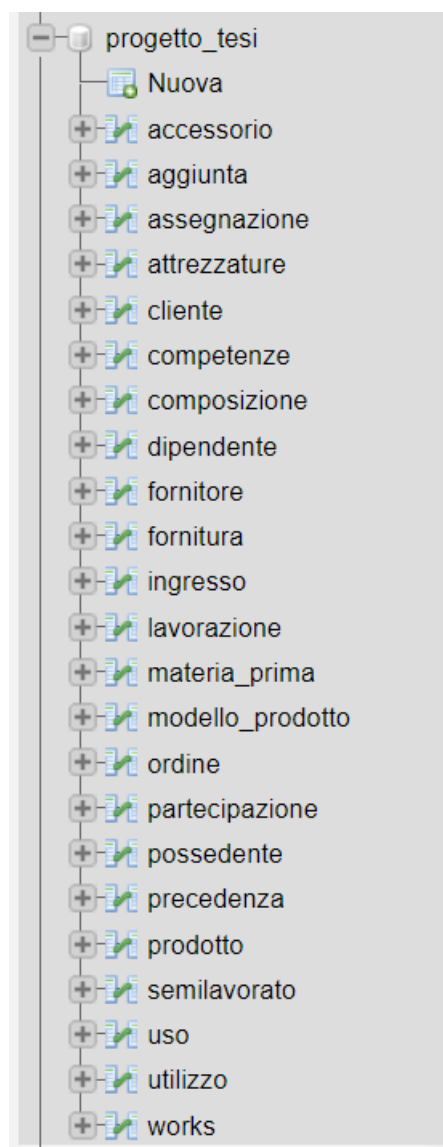


Fig. 4.1

```
|create TABLE accessorio (
    codice varchar(20) PRIMARY KEY,
    nome varchar(30) UNIQUE,
    quantita_necessaria int,
    quantita_presente int,
    check(quantita_presente>=0),
    check(quantita_necessaria>=0),
    lavorazione_input int references lavorazione (codice)
)
```

```
|create TABLE aggiunta(
    codice int PRIMARY KEY AUTO_INCREMENT,
    prodotto_agg varchar(20) REFERENCES prodotto(matricola),
    accessorio_agg varchar (30) REFERENCES accessorio(codice)
)
```

```
|CREATE TABLE assegnazione(
    codice int PRIMARY KEY AUTO_INCREMENT,
    lavorazione_ass varchar(50)REFERENCES lavorazione(codice),
    competenza_ass int REFERENCES competenze(codice)
)
```

```
|create table attrezzature(
    codice int Primary KEY AUTO_INCREMENT,
    nome varchar (200),
    quantita_presente int,
    check(quantita_presente>=0)
)
```

```
|CREATE TABLE cliente(
    partita_iva char(11) PRIMARY KEY,
    denominazione varchar(50) NOT NULL,
    e_mail varchar(50) NOT NULL,
    telefono varchar(15) NOT NULL
)
```

```
|CREATE TABLE competenze (
    codice int PRIMARY KEY AUTO_INCREMENT,
    nome varchar(30) NOT NULL UNIQUE
)
```

```
CREATE TABLE composizione(
    codice int PRIMARY KEY AUTO_INCREMENT,
    lavorazione_comp varchar(50) REFERENCES lavorazione (codice),
    modello_prodotto_comp varchar(25) REFERENCES modello_prodotto(matricola)
)
```

```
CREATE TABLE composizione(
    codice int PRIMARY KEY AUTO_INCREMENT,
    lavorazione_comp varchar(50) REFERENCES lavorazione (codice),
    modello_prodotto_comp varchar(25) REFERENCES modello_prodotto(matricola)
)
```

```
CREATE TABLE dipendente(
    codice int PRIMARY KEY AUTO_INCREMENT,
    user_name varchar(15) NOT NULL,
    passw varchar(40) NOT NULL,
    tipo varchar(40) NOT NULL,
    UNIQUE(user_name,passw)
)
```

```
CREATE TABLE fornitore(
    partita_iva char(11) PRIMARY KEY,
    denominazione varchar(20) UNIQUE NOT NULL,
    e_mail varchar (30) NOT NULL,
    telefono varchar(15) NOT NULL
)
```

```
CREATE TABLE forniture(
    codice int PRIMARY KEY AUTO_INCREMENT,
    data date NOT NULL,
    litri int,
    CHECK(litri>=0),
    metri_quadri int,
    CHECK(metri_quadri>=0),
    metri_cubi int,
    check(metri_cubi>=0),
    pezzi int,
    check(pezzi>=0),
    fornitore char(11) REFERENCES fornitore(partita_iva),
    materia_prima int REFERENCES materia_prima(codice)
)
```

```
|CREATE TABLE ingresso(  
    codice int PRIMARY KEY AUTO_INCREMENT,  
    semilavorato_ingr int REFERENCES semilavorato(codice),  
    lavorazione_ingr varchar(50) REFERENCES lavorazione(codice),  
    quantita_necessaria int NOT NULL,  
    check(quantita_necessaria>=0)  
)
```

```
|CREATE TABLE lavorazione (  
    codice varchar(50) PRIMARY KEY,  
    nome varchar(200) NOT NULL UNIQUE,  
    durata int,  
    check(durata>=0),  
    tipo varchar(50) NOT NULL,  
    dipendenti_necessari int,  
    check(dipendenti_necessari>=0),  
    semilavorato_uscita int REFERENCES semilavorato(codice),  
    categoria varchar(15),  
    accessorio_uscita varchar(20) references accessorio (codice)  
)
```

```
CREATE TABLE materia_prima(  
    codice int PRIMARY KEY AUTO_INCREMENT,  
    nome varchar(30) NOT NULL UNIQUE,  
    quantita_presente int,  
    check(quantita_presente >=0)  
)|
```

```
|create table modello_prodotto(  
  
    codice varchar(200) PRIMARY KEY,  
    nome varchar(25) UNIQUE  
)
```

```
|CREATE TABLE ordine(  
    numero int PRIMARY KEY AUTO_INCREMENT ,  
    data_scadenza date NOT NULL,  
    cliente char(11) REFERENCES cliente(partita_iva)  
)
```

```
CREATE TABLE partecipazione(
    codice int PRIMARY KEY AUTO_INCREMENT,
    dipendente_part int REFERENCES dipendente(matricola),
    work_part varchar(50) REFERENCES works (codice)
)
```

```
CREATE TABLE possidente (
    codice int PRIMARY KEY AUTO_INCREMENT,
    competenza int REFERENCES competenze(codice),
    dipendente int REFERENCES dipendente(matricola)
)
```

```
CREATE TABLE precedenza(
    codice int PRIMARY KEY AUTO_INCREMENT,
    lavorazione_precedente varchar(50)REFERENCES lavorazione(codice),
    lavorazione_successiva varchar(50)REFERENCES lavorazione(codice)
)
```

```
CREATE TABLE prodotto(
    matricola varchar(20) PRIMARY KEY,
    stato varchar(25),
    modello_prodotto varchar(10) REFERENCES modello_prodotto(codice),
    ordine int REFERENCES ordine(numero),
    data_messa_produzione date
)
```

```
create table semilavorato(
    codice int primary key AUTO_INCREMENT,
    nome varchar(30) UNIQUE,
    quantita_presente int,
    check(quantita_presente>=0)
)
```

```
CREATE TABLE uso (
    codice int PRIMARY KEY AUTO_INCREMENT,
    attrezzatura_uso int REFERENCES attrezzature(codice),
    lavorazione_uso varchar(200) REFERENCES lavorazione(codice),
    quantita_necessaria int,
    check(quantita_necessaria>=0)
)
```

```
CREATE TABLE utilizzo(
    codice int PRIMARY KEY AUTO_INCREMENT,
    materia_prima int REFERENCES materia_prima(codice),
    lavorazione_uti varchar(50) REFERENCES lavorazione(codiceL) ON DELETE cascade,
    quantita_necessaria int,
    check(quantita_necessaria>=0)
)
```

```
CREATE TABLE works(
    codice varchar(50) PRIMARY KEY,
    data_inizio date NOT NULL,
    ora_inizio time NOT NULL,
    data_fine date NOT NULL,
    ora_fine time NOT NULL,
    data_inizio_sosp date NOT NULL,
    ora_inizio_sosp time NOT NULL,
    data_fine_sosp date NOT NULL,
    ora_fine_sosp time NOT NULL,
    durata_effettiva time NOT NULL,
    quantita_prodotta int NOT NULL,
    check(quantita_prodotta>=0),
    check(durata_effettiva >=0),
    lavorazione_W varchar(50)REFERENCES lavorazione(codice),
    priorit  varchar(30),
    prodotto_W int references prodotto(matricola)
)
```

4.2 Codifica delle operazioni e screen di esecuzione

Dal punto di vista della progettazione del database dovremmo mostrare il codice SQL usato per l'implementazione delle varie operazioni. In realt  le operazioni elencate in fase di raccolta e analisi dei requisiti sono spesso pi  complesse di una singola query SQL. Inoltre a seguito di questo progetto, sar  realizzato un software che interagir  con questo database. Per tale motivo abbiamo direttamente creato delle funzioni in php che permettono l'interazione con il database e che concettualmente implementano le operazioni che a livello logico corrispondono a quelle precedentemente raccolte. In tal senso abbiamo creato una serie di classi che corrispondono alle entit  dello schema E-R e per ognuna inserito i metodi che eseguono le queries necessarie al database. Per ogni classe sono comuni i metodi registra(), elimina(), modifica(), lista(), dettaglio() che corrispondono rispettivamente alla registrazione, eliminazione e modifica di una occorrenza specifica della corrispondente relazione e alla visualizzazione di tutti gli attributi dell'occorrenza con dettaglio() e al recupero di tutte le occorrenze con lista(). Molte di queste classi contengono metodi pi  complessi a cui non corrisponderanno uniche query. Per non appesantire molto questo

documento, di seguito ne riportiamo alcuni di esempio e in particolar modo cercheremo di mostrare metodi che hanno permesso fare alcune riflessioni più interessanti.

Per le operazioni comuni a tutte le classi si è scelto di mostrare a titolo di esempio quelle della classe Ordine, lasciando banalmente immaginare come si estendono per le altre classi.

Registra ordine

```
public function registra ($data_scadenza, $cliente){  
    $inserimento_ordine = "INSERT INTO ordine (data_scadenza, cliente) VALUES  
    (\".$data_scadenza.\",\".$cliente.\")\";  
    $stm = $this->conn->prepare($inserimento_ordine);  
    try{  
        $stm->execute();  
        return array(true,"000","ORDINE REGISTRATO CORRETTAMENTE");  
    } catch (PDOException $ex){  
        return array(false, "025", "ERRORE NEL DB: ".$ex->getMessage());  
    }  
}
```

Modifica ordine

```
public function modifica ($numero,$data_scadenza, $cliente){  
    $modifica = "UPDATE ordine SET data_scadenza = \".$data_scadenza.\", cliente =  
    \".$cliente.\" WHERE numero = \".$numero.\"\"\"\";  
    $stm = $this->conn->prepare($modifica);  
    try{  
        $stm->execute();  
        return array(true,"000","ORDINE MODIFICATO CORRETTAMENTE");  
    } catch (PDOException $ex) {  
        return array(false, "026", "ERRORE NEL DB: ".$ex->getMessage());  
    }  
}
```

Elimina ordine

```
public function elimina ($numero){  
    $elimina = "DELETE FROM ordine WHERE numero =\".$numero.\"\"\"\";  
    $stm = $this->conn->prepare($elimina);  
    try{
```

```

        $stm->execute();
        return array(true,"000","ORDINE ELIMINATO CORRETTAMENTE");
    } catch (PDOException $ex) {
        return array(false, "027", "ERRORE NEL DB: ".$ex->getMessage());
    }
}

```

Lista ordini ordinata per data scadenza

```

public function lista() {
    $lista = "SELECT * FROM ordine ORDER BY data_scadenza";
    $stm = $this->conn->prepare($lista);
    $stm->execute();
    $array_tabella = array();
    while ($array_riga = $stm->fetch(PDO::FETCH_ASSOC)) {
        $array_app["numero"] = $array_riga["numero"];
        $array_app["data_scadenza"] = $array_riga["data_scadenza"];
        $array_app["cliente"] = $array_riga["cliente"];
        array_push($array_tabella, $array_app);
    }
    return $array_tabella;
}

```

Dettaglio ordine

```

public function dettaglio ($numero) {
    $dettaglio = "SELECT data_scadenza, cliente FROM ordine WHERE numero= '
    ".$numero."'";
    $stm = $this->conn->prepare($dettaglio);
    $stm->execute();
    $array_app = array();
    if ($array_riga = $stm->fetch(PDO::FETCH_ASSOC)) {
        $array_app = $array_riga;
        $array_app["data_scadenza"] = $array_riga["data_scadenza"];
        $array_app["cliente"] = $array_riga["cliente"];
    }
    return $array_app;
}

```


Mostro di seguito, invece l'esempio di due operazioni che permettono di effettuare controlli.

Controllo registro work

Questo metodo viene richiamato ogni qual volta un prodotto deve essere messo in produzione. Il controllo avviene per ogni lavorazione che deve essere effettuata per il prodotto. Ha lo scopo di verificare che tutte le risorse (semilavorati, accessori, attributi e materie prime) necessarie all'inizio di una lavorazione sono disponibili nelle quantità richieste dalla lavorazione stessa.

```
public function controllo($codice){  
    $contaSemi = "SELECT COUNT(I.codice) as contaS FROM ingresso I JOIN semilavorato  
    S ON (I.semilavorato_ingr = S.codice) WHERE I.quantita_necessaria > S.quantita_presente  
    AND lavorazione_ingr = '". $codice. "'";  
    $contaAcc = "SELECT COUNT(codice) as contaA FROM accessorio WHERE  
    quantita_necessaria > quantita_presente AND lavorazione_input = '". $codice. "'";  
    $contaAttr = "SELECT COUNT(U.codice) as contaAT FROM uso U JOIN attrezzature T  
    ON (U.attrezzatura_uso = T.codice) WHERE U.quantita_necessaria > T.quantita_presente  
    AND U.lavorazione_uso = '". $codice. "'";  
    $conta_materie_prime = "SELECT COUNT(Z.codice) as contaM FROM utilizzo Z JOIN  
    materia_prima M ON (Z.materia_prima = M.codice) WHERE Z.quantita_necessaria >  
    M.pezzi AND Z.quantita_necessaria > M.metri_quadri AND Z.quantita_necessaria >  
    M.metri_cubi AND Z.quantita_necessaria > M.litri AND Z.lavorazione_uti = '". $codice. "'";  
    $stm1 = $this->conn->prepare($contaSemi);  
    $stm1->execute();  
    $stm2 = $this->conn->prepare ($contaAcc);  
    $stm2->execute();  
    $stm3 = $this->conn->prepare($contaAttr);  
    $stm3->execute();  
    $stm4 = $this->conn->prepare($conta_materie_prime);  
    $stm4->execute();  
    $risultato1 = $stm1->fetch(PDO::FETCH_ASSOC);  
    $risultato2 = $stm2->fetch(PDO::FETCH_ASSOC);  
    $risultato3 = $stm3->fetch(PDO::FETCH_ASSOC);  
    $risultato4 = $stm4->fetch(PDO::FETCH_ASSOC);  
    $ris["contaS"] = $risultato1["contaS"];  
    $ris["contaA"] = $risultato2["contaA"];
```

```

$ris["contaAT"] = $risultato3["contaAT"];
$ris["contaM"] = $risultato4["contaM"];
if($ris["contaS"]==0 AND $ris["contaA"]==0 AND $ris["contaAT"]==0 AND
$ris["contaM"]== 0){
    return true;
}else{
    return false;
}
}

```

Controlla numero dipendenti

Questo metodo viene richiamato ogni qual volta un prodotto deve essere messo in produzione. Il controllo avviene per ogni lavorazione che deve essere effettuata per il prodotto. L'idea è quella di verificare che i dipendenti che si sono assegnati il work sono in numero pari a quello dei dipendenti necessari per quel work.

```

public function controllaNumDip ($lavorazione_W){
    $controlla_dip = "SELECT L.dipendenti_necessari as numero_dipendenti FROM works W
JOIN lavorazione L ON(W.lavorazione_W = L.codice) WHERE W.lavorazione_W =
"."$lavorazione_W."";
    $controlla_part = "SELECT COUNT(P.codice) as partecipantiFROM partecipazione P
JOIN works W ON (W.codice = P.work_part) WHERE W.lavorazione_W =
"."$lavorazione_W."";
    $stm1 = $this->conn->prepare($controlla_dip);
    $stm1->execute();
    $stm2 = $this->conn->prepare($controlla_part);
    $stm2->execute();
    $risultato1 = $stm1->fetch(PDO::FETCH_ASSOC);
    $risultato2 = $stm2->fetch(PDO::FETCH_ASSOC);
    $ris["numero_dipendenti"] = $risultato1["numero_dipendenti"];
    $ris["partecipanti"] = $risultato2["partecipanti"];
    if($ris["numero_dipendenti"] == $ris["partecipanti"]){
        return true;
    }else{
        return false;
    }
}

```

}

Inizio Work

Questa operazione corrisponde all'atto del dipendente nell'assegnarsi un work che sta per eseguire. In questo momento dobbiamo considerare che il work sarà cominciato e saranno impegnate tutte le risorse che esso eventualmente richiede: semilavorati in ingresso, materie prime, accessori e attrezzature. Prima di poter cominciare il work, è necessario controllare che le risorse siano disponibili. Inoltre, ovviamente, sarà necessario settare la data e l'ora esatta dell'inizio della work. Allora sarà possibile eseguire concettualmente tre task: il primo è quello che controlla che le risorse necessarie siano a disposizione per il work, il secondo modifica l'attributo data_inizio della relazione Works, il terzo calcola la quantità usata da quel work per le risorse e per ognuna aggiorna gli attributi di disponibilità attuale. Ne consegue:

- 1) Metodo controllo(\$codice_work) della classe Work;
- 2) Metodo inizioWork(\$codice_work) della classe Work;
- 3) Metodo aggiornaRisorse(\$codice_work) della classe Work;

Di seguito l'implementazione dei metodi stessi:

```
public function controllo($codice){  
    $contaSemi = "SELECT COUNT(I.codice) as contaS FROM ingresso I JOIN semilavorato S  
    ON (I.semilavorato_ingr = S.codice) WHERE I.quantita_necessaria > S.quantita_presente  
    AND lavorazione_ingr = '". $codice. "'";  
    $contaAcc = "SELECT COUNT(codice) as contaA FROM accessorio WHERE  
    quantita_necessaria > quantita_presente AND lavorazione_input = '". $codice. "'";  
    $contaAttr = "SELECT COUNT(U.codice) as contaAT FROM uso U JOIN attrezzature T  
    ON (U.attrezzatura_uso = T.codice) WHERE U.quantita_necessaria > T.quantita_presente  
    AND U.lavorazione_uso = '". $codice. "'";  
    $conta_materie_prime = "SELECT COUNT(Z.codice) as contaM FROM utilizzo Z JOIN  
    materia_prima M ON (Z.materia_prima = M.codice) WHERE Z.quantita_necessaria >  
    M.pezzi AND Z.quantita_necessaria > M.metri_quadri AND Z.quantita_necessaria >  
    M.metri_cubi AND Z.quantita_necessaria > M.litri AND Z.lavorazione_uti = '". $codice. "'";  
    $result_contaSemi = execute($contaSemi);  
    $result_contaAcc = execute($contaAcc);  
    $result_contaAttr = execute($contaAttr);  
    $result_contaMaPrim = execute($conta_materie_prime);
```

```

if($result_contaSemi == 0 AND $result_ContaAttr == 0 AND $result_contaAcc == 0 AND
$result_contaMatPrim == 0){
    return true;
}else{
    return false;
}
}

public function inizioWork ($codice,$data,$ora){
    execute( "UPDATE works SET data_inizio = ".$data.", ora_inizio = ".$ora." WHERE
    codice = ".$codice.""
}

public function aggiornaRisorse($codice){

    $query_semi = "SELECT semilavorato_ingr, quantita_necessaria FROM Ingresso WHERE
    lavorazione_ingr = ".$codice."";
    $query_mat_prim = "SELECT materia_prima, quantita_necessaria FROM Utilizzo
    WHERE lavorazione_uti = ".$codice."";
    $query_acc = "SELECT codice, quantita_necessaria FROM Accessorio WHERE
    lavorazione_input = ".$codice."";
    $query_attr = "SELEC attrezzatura_uso, quantita_necessaria FROM Uso WHERE
    lavorazione_uso = ".$codice."";
    $result_semi = execute($query_semi);
    $result_mat_pri = execute($query_mat_pri);
    $result_acc = execute($query_acc);
    $result_attr = execute($query_attr);
    foreach ($cod, $qta_nec in $result_semi){
        execute("UPDATE Semilavorato SET quantita_presente = quantita_presente -
        $qta_nec WHERE codice = $cod");
    }
    foreach($cod, $qta_nec in $result_mat_pri){
        execute("UPDATE materia_prima SET quantita_presente = quantita_presente -
        $qta_nec WHERE codice = $cod");
    }
    foreach($cod, $qta_nec in $result_acc){

```

```

        execute("UPDATE accessorio SET quantita_presente = quantita_presente -
        $qta_nec WHERE codice = $cod");
    }
    foreach($cod, $qta_nec in $result_attr){
        execute("UPDATE attrezzature SET quantita_presente = quantita_presente -
        $qta_nec WHERE codice = $cod");
    }
}

```

Visualizza A schermo

L'operazione di visualizzare a schermo i works da eseguire è di necessaria importanza, nel senso che l'ordine con cui esse sono visualizzate comporta anche l'ordine di base con cui dovrebbero essere eseguite. Innanzitutto l'ordine è dettato dalla priorità, che calcoliamo in fase di registrazione dei works, a seguito dell'inserimento di un nuovo ordine. Non ci dilunghiamo in questo momento troppo su cosa implica registrare i works, ma possiamo immaginare che saranno inseriti nell'opportuna relazione da database tutti i work ogni qual volta registriamo un nuovo ordine. L'ordine è costituito da un insieme di prodotti e per ogni prodotto conosciamo le lavorazioni che sono necessarie con tutti i vincoli di precedenza tra le stesse. In pratica saranno registrati tutti i works corrispondenti a tutte le lavorazioni necessarie alla realizzazione di tutti i prodotti di quell'ordine. In questa fase calcoliamo una priorità da assegnare al works. Tale priorità è un intero che si calcola tenendo conto della data di scadenza entro la quale quel prodotto deve essere consegnato (scadenza dell'ordine) e che viene incrementata a seconda del numero di lavorazioni che la seguono in precedenza, cioè che sono conseguenze dell'esecuzione di quella lavorazione. Questo permette, quindi, non solo di mostrare prima le lavorazioni dei prodotti da consegnare prima, ma tra quelle dello stesso prodotto di mostrare prima quelle che sono necessarie all'esecuzione delle successive.

Grazie a questo meccanismo la visualizzazione delle lavorazioni a schermo corrisponderà ad una selezione dei works ordinandoli per il campo priorità.

E' importante osservare (approfondimento nel capitolo conclusivo) che nel fornire questo ordine, non è tenuto in considerazione un aspetto spesso importante dal punto di vista dell'ottimizzazione della produzione, soprattutto in termini di costi. Spesso nelle fabbriche manifatturiere la presenza di macchinari (in questo schema appartenenti all'entità Attrezzature) significa anche la necessità di sfruttarli il numero di ore maggiore possibile. Da un lato alcune di queste macchine hanno tempi e costi di avvio onerosi, per cui conviene tenerle sempre accese, dall'altra parte, sono in grado di

agevolare molte fasi produttive e quindi permettere una rendita tempo/quantità prodotta nettamente superiore a quella di un dipendente. Anche in questo secondo caso sarebbe importante usarle il più possibile.

Non è approfondito l'argomento in questo documento ma è proposto un suggerimento implementativo. Quando si decide di visualizzare a schermo l'elenco dei works da eseguire, è possibile immaginare di dedicare una sezione a quelli che chiamati WORKS CONSIGLIATI, ovvero alcuni works che si consiglia di fare indipendentemente dalla priorità che essi hanno in quel momento e quindi indipendentemente dall'ordine di appartenenza. In questa sezione sono inseriti quei works che permetterebbero, se eseguiti, di impegnare continuamente i macchinari dell'azienda. Nell'esempio precedente si è mostrato come eseguire l'operazione di inizio work. In particolare modo si può ricordare che nel caso in cui il controllo riporta esito positivo, oltre ad aggiornare il campo di data inizio del work, sono aggiornate anche le quantità presenti di tutte le risorse. In pratica gli attributi `quantita_presente` delle relazioni `Semilavorati`, `Materie_Prime`, `Accessori` e `Attrezzature`. Essi esprimono in ogni istante la quantità di quegli oggetti presenti. Nel caso di attrezzature tale attributo va interpretato come la quantità di attrezzi (macchinari) attualmente non in uso. Allora basterà verificare quali attrezzature hanno quell'attributo ad un valore maggiore di zero per farci rendere conto che quell'attrezzo non si sta usando in quel momento. Una volta individuato è possibile cercare tra i works quelli che necessitano dell'attrezzo per essere eseguiti, quindi stamparli tra i consigliati. L'ideale sarebbe inserire questa procedura direttamente nel calcolo della priorità, in modo da obbligare gli operai a eseguire tali work, ma questo potrebbe comportare una perdita del principio di esecuzione di works per priorità di consegna. La problematica è più complessa e per lo scopo di questa tesi non si è ritenuto opportuno affrontarla.

Per l'implementazione sono stati inseriti due metodi all'interno della classe `Works` nominati rispettivamente `visualizzaAschermo()` e `visualizzaConsigliati()`.

```
public function visualizzaAschermo(){
    $query = "SELECT W.codice, L.nome, L.categoria, L.durata, W.priorita, W.prodotto_W
    FROM Works W JOIN Lavorazione L ON W.lavorazione_W = L.codice ORDER BY priorita
    DESC;
    $result = execute($query);
    echo $result;
}

public function visualizzaConsigliati(){
    $query_attr = "SELECT codice FROM Attrezzature WHERE quantita_presente > 0";
    $result_attr = execute($query_attr);
```

```

$selenco_codici_attr = ( ..., ...) da $result_attr;
$query_consigliati = "SELECT W.codice, L.nome, L.categoria, L.durata, W.priorita,
W.prodotto_W, A.nome FROM Works W JOIN Lavorazione L JOIN Uso U JOIN
Attrezzature A ON W.lavorazione_W = L.codice AND U.lavorazione_uso = L.codice AND
U.attrezzatura_uso = A.codice WHERE A.codice IN $selenco_codici_attr ORDER BY
W.priorita";
echo $result_consigliati;
}

```

Come si può osservare, nel secondo metodo si cerca prima l'elenco dei codici di attrezzature che attualmente non sono usate e successivamente si cercano quei works scaturiti da quelle lavorazioni che sono in relazione (USO) con quelle attrezzature.

Calcolo durata effettiva

Questo metodo è un'operazione di tipo batch. Esso viene richiamato ogni qual volta il dipendente inizia, finisce o sospende una lavorazione (mediante l'uso di un tablet).

```

public function inizioWork ($codice,$data,$ora){
    $modifica_inizio = "UPDATE works SET data_inizio = ".$data.", ora_inizio = ".$ora."
    WHERE codice = ".$codice."";
    $stm = $this->conn->prepare($modifica_inizio);
    try{
        $stm->execute();
        return array(true,"000","DATA INIZIO MODIFICATA CORRETTAMENTE");
    } catch (PDOException $ex) {
        return array(false, "0100", "ERRORE NEL DB: ".$ex->getMessage());
    }
}

```

In questo modo si modifica il particolare work settando la data e l'ora di inizio.

```

public function sospendiWork ($codice,$data,$ora, $ora_inizio, $data_fine_sosp,$ora_fine_sosp,
$durata_effettiva){
    if($data_fine_sosp == "0000-00-00" AND $ora_fine_sosp=="00:00:00"){
        $durata_effettivagg = $this-> differenza_tempo($ora,$ora_inizio);
    }
}

```

```

$modifica_sosp1 = "UPDATE works SET data_inizio_sosp =
".$data.",ora_inizio_sosp = ".$Sora.", durata_effettiva = ".$durata_effettivagg."
WHERE codice = ".$codice."";
$stmt = $this->conn->prepare($modifica_sosp1);
try{
    $stmt->execute();
    return array(true,"000","DATA INIZIO SOSPENSIONE MODIFICATA
CORRETTAMENTE");
}catch (PDOException $ex) {
    return array(false, "0101", "ERRORE NEL DB: ".$ex->getMessage());
}
}else{
    $diff = $this->differenza_tempo($Sora , $Sora_fine_sosp);
    $durata_effettivagg = $durata_effettiva + $diff;
    $modifica_sosp2 = "UPDATE works SET data_inizio_sosp = ".$data.",
ora_inizio_sosp = ".$Sora.", durata_effettiva = ".$durata_effettivagg." WHERE
codice = ".$codice."";
    $stmt = $this->conn->prepare($modifica_sosp2);
    try{
        $stmt->execute();
        return array(true,"000","DATA INIZIO SOSPENSIONE MODIFICATA
CORRETTAMENTE");
    }catch (PDOException $ex) {
        return array(false, "0102", "ERRORE NEL DB: ".$ex->getMessage());
    }
}
}
}

```

Questo metodo invece viene usato nel momento in cui il dipendente deve sospendere il lavoro. Ad esempio perché è finita la giornata lavorativa e dunque il work verrà ripreso il giorno successivo. La durata effettiva viene aggiornata facendo la differenza tra l'intervallo di tempo che intercorre tra la sospensione e l'inizio lavorazione oppure tra l'inizio sospensione e la fine sospensione (nel caso in cui il work viene più volte sospeso).

```

public function riprendiWork ($codice,$data,$Sora){

```



```

$modifica_fine = "UPDATE works SET data_fine_sosp = ".$data.", ora_fine_sosp = '
".$Sora.'" WHERE codice = ".$codice.'";
$stmt = $this->conn->prepare($modifica_fine);
try{
    $stm->execute();
    return array(true,"000","DATA E ORA FINE SOSPENSIONE MODIFICATA
CORRETTAMENTE");
} catch (PDOException $ex){
    return array(false, "0102", "ERRORE NEL DB: ".$ex->getMessage());
}
}

```

Il metodo viene richiamato nel momento in cui bisogna riprendere il work. In questo caso non c'è calcolo della durata effettiva.

public function fineWork

```

($codice,$data,$Sora,$data_fine_sosp,$Sora_fine_sosp,$Sora_inizio,$durata_effettiva){
    if($data_fine_sosp == "0000-00-00" AND $Sora_fine_sosp == "00:00:00"){
        $durata_effettivagg = $this->differenza_tempo($Sora, $Sora_inizio);
        $fine = "UPDATE works SET data_fine = ".$data.", ora fine =
".$Sora.",durata_effettiva = ".$durata_effettivagg.'" WHERE codice = ".$codice.'";
        try{
            $stm->execute();
            return array(true,"000","DATA E ORA FINE WORK MODIFICATA
CORRETTAMENTE");
        } catch (PDOException $ex) {
            return array(false, "0103", "ERRORE NEL DB: ".$ex->getMessage());
        }
    }else{
        $diff = $this->differenza_tempo($Sora, $Sora_fine_sosp);
        $durata_effettivagg= $durata_effettiva + $diff;
        $fine1 = "UPDATE works SET data_fine = ".$data.", ora fine =
".$Sora.",durata_effettiva = ".$durata_effettivagg.'" WHERE codice = ".$codice.'";
        $stm = $this->conn->prepare($fine1);
        try{
            $stm->execute();

```

```

        return array(true, "000", "DATA E ORA FINE WORK MODIFICATA
CORRETTAMENTE");
    } catch (PDOException $ex) {
        return array(false, "0103", "ERRORE NEL DB: ".$ex->getMessage());
    }
}
}

```

Nel momento in cui il work viene terminato si calcola la durata effettiva, nel caso in cui il work non ha avuto sospensioni la durata effettiva è ottenuta sottraendo l'intervallo di tempo che intercorre tra fine work e inizio work; altrimenti essa è ottenuta aggiungendo alla durata calcolata la differenza di tempo tra fine work e fine sospensione.

4.3 Operazioni di analisi

Calcola tempo inattività

```

public function tempoInattivita ($dipendente, $data_inizio){
    $inattivita = "SELECT SUM(W.durata_effettiva) as durata_effettiva FROM works W JOIN
partecipazione P ON (P.work_part = W.codice) WHERE dipendente_part = '". $dipendente. "'
AND W.data_inizio = '". $data_inizio. "'";
    $stm = $this->conn->prepare($inattivita);
    $stm->execute();
    $array_riga = $stm->fetch(PDO::FETCH_ASSOC);
    $array_app["durata_effettiva"] = $array_riga["durata_effettiva"];
    $tempo_inattivo = 8 - $array_app["durata_effettiva"];
    return $tempo_inattivo;
}

```

Il metodo va a verificare la durata effettiva dei work a cui il dipendente ha partecipato in una particolare data. Ottenendo così la durata di operatività che sottratta alla giornata operativa riporta il tempo di inattività.

Calcola tempo produzione

```

public function tempoProduzione ($prodotto_W){

```

```
$tempoProduzione = "SELECT COUNT(W.durata_effettiva) as tempo FROM works W
WHERE prodotto_W = ".$prodotto_W."";
$stmt = $this->conn->prepare($tempoProduzione);
$stmt->execute();
$array_riga = $stmt->fetch(PDO::FETCH_ASSOC);
$array_app["tempo"] = $array_riga["tempo"];
return $array_app;
}
```

Il metodo riporta la somma delle durate effettive dei work associati al determinato prodotto.

Capitolo 5

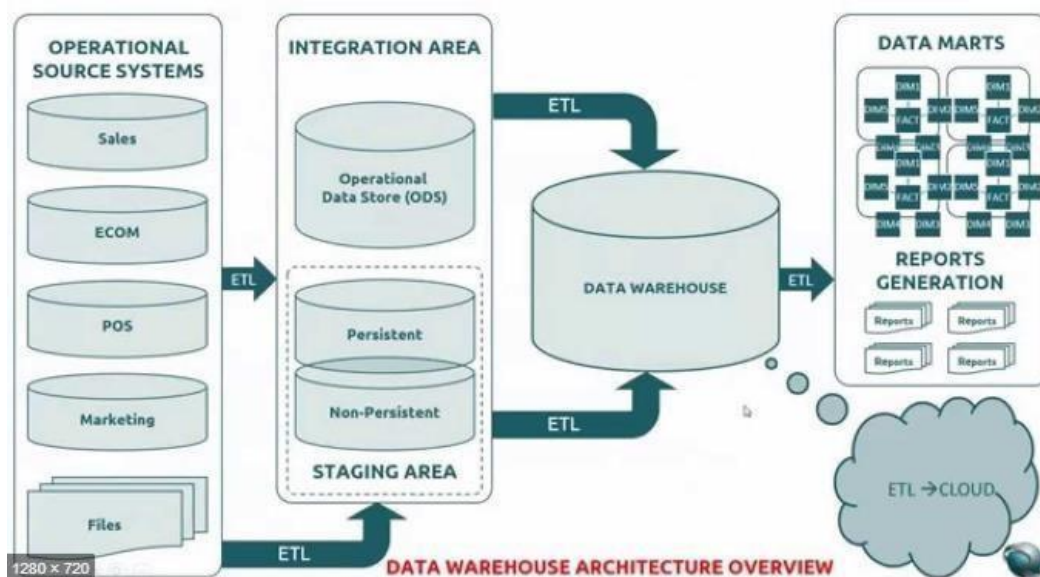
Conclusione

5.1 Considerazioni su operazioni di analisi

Sotto richiesta del titolare dell'azienda sono state eseguite delle operazioni di analisi dei dati. Le operazioni in questione sono le seguenti:

1. Calcolare i minuti di inattività degli operai (una volta a settimana).
2. Calcolare il tempo impiegato per la produzione di un prodotto (circa 1 volta a settimana).

Il risultato di tali operazioni contiene delle informazioni di carattere storico/temporale che godono di esistenza autonoma e che dunque possono essere utili per l'azienda nelle decisioni a livello gestionale. Ciò potrebbe suggerire l'implementazione di un'architettura per l'analisi dei dati. Una datawarehouse è sostanzialmente una base di dati strutturata le cui sorgenti provengono o da database di tipo operativo oppure da fonti esterne (indagini, sistemi privi di DBMS). Esse godono delle seguenti proprietà: Integrità e consistenza, rappresentativa di informazioni temporali, orientata ai soggetti e non volatile. L'architettura di una datawarehouse è schematizzata in figura,



Dalla figura notiamo un primo blocco Operational source system in cui vengono descritte le possibili sorgenti di natura diversa (database operativi, indagini, files generici). Le informazioni prima di poter essere sfruttate dalla datawarehouse devono essere integrate tra loro. Questo passaggio viene fatto attraverso gli strumenti ETL che attuano operazioni di estrazioni di dati dalle sorgenti, di trasformazione, di pulizia e di caricamento. In questo modo la datawarehouse viene popolata. L'accesso alla struttura sono soltanto in lettura e costituiscono uno storico rivolto a soggetti d'impresa.

Nella realtà di interesse dello studio la struttura della datawarehouse imploderebbe in un semplice database che raccoglie dati di analisi estratti dal database operativo attraverso le operazioni di analisi effettuate. Ad esempio il database che per comodità chiamerò di analisi potrebbe contenere una relazione chiamata "inattività" con attributi "dipendente" e "tempo inattività" e l'operazione di analisi "calcola tempo inattività" servirebbe per popolare il database. Questa soluzione è suggerita dal fatto che l'operazione di analisi in questione ha una frequenza elevata. Tuttavia considerata la quantità minima di operazioni di analisi e il loro tempo minimo di esecuzione nel database operativo la realizzazione di un nuovo database è da escludere.

5.2 Considerazioni sulle operazioni

Le operazioni implementate sono state quelle richieste dal titolare dell'azienda, inoltre considerato che il software di produzione non è stato progettato alcune operazioni saranno da rivedere e altre da aggiungere. In questo paragrafo sono riportate delle considerazioni che rendono le operazioni implementate.

L'operazione "inizia works" descritta nel capitolo 4 può presentare dei problemi di concorrenza nel caso in cui essa venga richiamata contemporaneamente per due works differenti. Se ad esempio ci fossero due works che utilizzano le stesse materie prime e queste risultano insufficienti per realizzare entrambi i works si incorrerebbe ad incoerenza dei dati. Per questo motivo l'operazione composta "inizio works" deve costituire una transazione.

L'operazione "controllo" che fa parte di "inizio works" verifica se sia possibile registrare o meno una lavorazione in base alla quantità disponibile di materie prime, accessori, semilavorati e attrezzature, ma non prevede lo scarto possibile durante la lavorazione. In sostanza può accadere che la quantità necessaria (di materie prime, semilavorati, accessori e attrezzature) non sia quella indicata. A tal proposito una possibile soluzione può essere la predisposizione di un valore di quantità necessaria maggiore di quella prevista (la differenza tra questo valore e la quantità prevista dovrà essere arbitrario). In questo modo l'operazione andrà rivista effettuando il confronto tra quantità presente e questo nuovo valore.

Riguardo l'operazione "visualizza consigliate", essa viene richiamata nel momento in cui gli operai dovranno accedere al software per assegnarsi le lavorazioni. Il dipendente vedrà la sezione "consigliate" in cui compariranno le lavorazioni ordinate in base alla priorità (come discusso nel capitolo 4). Considerando che nelle lavorazioni consigliate sono presenti solo le lavorazioni legate alle attrezzature di tipo macchinario, la priorità calcolata sul work può essere messa in discussione utilizzando invece della priorità un algoritmo per l'ottimizzazione dei tempi di uso dei macchinari. A questo punto invece di utilizzare la priorità per ordinare i work si potrebbe utilizzare la durata ordinando in base al work che ha durata minore e realizzando in questo modo l'algoritmo SRTF. Se invece si volesse utilizzare un modo per ottimizzare lo sfruttamento dei macchinari in generale, ovvero non limitato a quei macchinari che sono disponibili andrebbe eliminata l'operazione di controllo sulla disponibilità dei macchinari rendendo in questo modo le lavorazioni che li coinvolgono slegate dal resto delle lavorazioni. In questo modo l'uso dei macchinari può essere programmato indipendentemente,

5.3 Sviluppi futuri

Sicuramente la realizzazione del database è legata alla progettazione e implementazione del software ad esso correlato che in futuro sarà realizzato con tutte le caratteristiche necessarie per essere comodamente fruibile ai dipendenti e all'amministrazione in fase di analisi.

Sarebbe opportuno approfondire l'aspetto di Ottimizzazione della produzione. L'incentivo all'uso continuo dei macchinari è un primo step di questo processo di ottimizzazione, ma osservando da un punto di vista più ampio si può mettere in risalto il fatto che la base dati operativa, e il suo funzionamento nel tempo costituirà un insieme di informazioni fondamentali per poter analizzare e valutare ordini di esecuzione differenti. Nelle lavorazioni si è usato il campo durata all'interno del quale sarà inserita una durata stimata dell'operazione. Man mano che il software sarà utilizzato si potrà verificare il tempo che effettivamente i dipendenti ci impiegano, quindi ottenere dati sempre più fini e veritieri. La precedenza che sussiste tra le operazioni, la scadenza degli ordini, nonché dei prodotti sono un insieme di vincoli da rispettare obbligatoriamente.

In poche parole si potrà impostare un algoritmo che risolva un problema tipico della ricerca operativa ponendosi l'obiettivo di ottimizzare la produzione nel rispetto dei vincoli. Volendo azzardare una analisi preliminare di tale problema, è possibile individuare alcune funzioni obiettivo:

- Funzione che calcola il tempo di produzione di un gruppo di prodotti (da minimizzare);
- Funzione che calcola il tempo di inattività dei macchinari (da minimizzare);
- Funzione che calcola le ore lavorative ed eventuali straordinari dei dipendenti (da minimizzare);

Sono diversi aspetti, da tenere tutti in considerazione allo scopo comune di ridurre i costi. E' importante però osservare che a queste possiamo aggiungere tutte le equazioni che esprimono i vincoli del problema:

- Vincoli di scadenza degli ordini;
- Vincoli di precedenza tra le lavorazioni sullo stesso prodotto;
- Vincoli sulla disponibilità dei macchinari;
- ...

Sarebbe assurdo trattare in questo documento una possibile impostazione del problema, ma è stato importante mettere in risalto questa possibilità e mostrare come la base di dati operativa che abbiamo creato possa costituire un buon punto di partenza per la soluzione. L'impostazione e la soluzione di questa ottimizzazione comporterebbe una rivisitazione dell'algoritmo attraverso cui si è calcolata la priorità dei works, che quel punto sarà fornita proprio dalla soluzione al problema.

Bibliografia e Sitografia

Bibliografia

Basi di dati – 1 febbraio 2014 di Paolo Atzeni, Stefano Ceri, Piero Fraternali, Stefano Paraboschi, Riccardo Torlone, McGraw Hill

Sitografia

<https://www.php.net/>

<https://www.html.it/guide/guida-php-di-base/>

<https://www.apachefriends.org/it/index.html>

<https://www.mysql.com/it/>

<https://www.php.net/manual/en/book.pdo.php>