

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA  
CAMPUS DI CESENA

---

Scuola di Ingegneria e Architettura  
Corso di Laurea Magistrale in Ingegneria e Scienze Informatiche

**Reingegnerizzazione di un  
Sistema Informativo: analisi,  
diagnosi e prototipazione**

Tesi di laurea in  
SISTEMI INFORMATIVI

*Relatore*

**Prof. Matteo Golfarelli**

*Candidato*

**Francesco Gorini**

---



*Ad Aurora, ai miei genitori e parenti, ai miei amici*



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Obiettivi . . . . .	1
1.2	Il dominio in analisi . . . . .	2
1.3	Struttura della tesi . . . . .	2
<b>2</b>	<b>Analisi SI aziendale</b>	<b>3</b>
2.1	Il progetto . . . . .	3
2.2	Obiettivi . . . . .	4
2.3	Kick-off meeting . . . . .	5
2.3.1	Committente di progetto . . . . .	5
2.3.2	Team di progetto . . . . .	5
2.3.3	Pianificazione . . . . .	5
2.4	Analisi . . . . .	8
2.4.1	Interviste con responsabili . . . . .	8
2.4.2	Interviste con operativi . . . . .	9
2.5	Diagnosi . . . . .	10
2.5.1	Business Performance Analysis . . . . .	10
2.5.2	Digital Asset Analysis . . . . .	14
2.5.3	Analisi Strategica . . . . .	16
<b>3</b>	<b>Tecnologie utilizzate</b>	<b>21</b>
3.1	Architettura . . . . .	21
3.2	MySQL Workbench . . . . .	24
3.3	Power BI . . . . .	24

<b>4</b>	<b>Progettazione e prototipazione Data Mart</b>	<b>27</b>
4.1	L'area in analisi . . . . .	27
4.2	Obiettivi . . . . .	30
4.3	Pianificazione . . . . .	31
4.4	Progettazione e prototipazione Data Mart . . . . .	35
4.4.1	Analisi dei requisiti . . . . .	36
4.4.2	Analisi e riconciliazione delle sorgenti . . . . .	39
4.4.3	Progettazione concettuale . . . . .	41
4.4.4	Carico di lavoro e volume dati . . . . .	43
4.4.5	Progettazione logica . . . . .	45
4.4.6	Progettazione dell'ETL . . . . .	49
4.4.7	Progettazione fisica . . . . .	54
4.4.8	Implementazione della reportistica . . . . .	54
4.4.9	Testing . . . . .	55
<b>5</b>	<b>Conclusioni</b>	<b>57</b>
5.1	Risultati ottenuti . . . . .	57
5.2	Possibili sviluppi futuri . . . . .	58

# Elenco delle figure

2.1	diagramma di Gantt per la gestione di progetto. . . . .	6
2.2	esempio di interviste a piramide e di interviste a imbuto. . . . .	9
2.3	catena del valore di Porter dell'azienda. . . . .	11
2.4	moduli che compongono il Portafoglio Applicativo. . . . .	15
2.5	architettura funzionale SI. . . . .	17
2.6	proposta della nuova architettura funzionale del SI. . . . .	19
2.7	proposta della nuova divisione Sistemi Informativi. . . . .	20
3.1	tecnologie dei database sorgenti e di destinazione. . . . .	22
3.2	database <i>on-premise</i> e su <i>cloud</i> . . . . .	23
3.3	schermata di lavoro su MySQL Workbench. . . . .	24
4.1	il ciclo di sviluppo di un Data Mart. . . . .	28
4.2	organizzazione del processo di Customer Management all'interno dell'azienda. . . . .	29
4.3	rappresentazione grafica delle informazioni contenute nel ticket di Redmine. . . . .	30
4.4	infrastruttura del Data Mart. . . . .	34
4.5	database model del software Redmine. . . . .	40
4.6	schema concettuale utilizzando la rappresentazione DFM. . . . .	42
4.7	carico di lavoro rappresentato graficamente con il DFM. . . . .	44
4.8	schema logico non ottimizzato. . . . .	47
4.9	tuning della gerarchia condivisa <i>Data</i> . . . . .	48
4.10	risultato di progettazione logica. . . . .	50
4.11	rappresentazione semplificata della soluzione con VPN. . . . .	50
4.12	workflow ETL. . . . .	52
4.13	report iniziale creato con Power BI. . . . .	54
4.14	report che descrive l'andamento dei ticket. . . . .	55





# Elenco dei listati

4.1	esempio di stored procedure realizzata. . . . .	53
4.2	ETL realizzato con il costrutto <i>event</i> . . . . .	53



# Capitolo 1

## Introduzione

L'obiettivo di ciascuna azienda privata, piccola o grande che sia, è quello di ottenere utili attraverso la commercializzazione di beni o servizi. Per raggiungere ciò, la base da cui si parte è sempre una corretta organizzazione della struttura e dei processi aziendali. Questi ultimi, per poter raggiungere i risultati attesi, hanno bisogno costantemente di informazioni. All'interno di un'impresa la parte che si occupa della gestione di informazioni e processi viene chiamata **Sistema Informativo (SI)**. Oltre ai processi aziendali, la base da cui ogni azienda non può prescindere è il dato. Negli ultimi anni la mole da gestire è aumentata in maniera esponenziale e con essa la complessità di gestione. All'interno del Sistema Informativo, la porzione che si occupa della gestione attraverso tecnologie informatiche prende il nome di **Sistema Informatico**. Una corretta gestione di processi e dati è fondamentale per il raggiungimento degli obiettivi aziendali a corto, medio e lungo termine. Senza di ciò, l'intero sistema produttivo rischia di non produrre i risultati attesi o di farlo portando con sé inefficienze, errori e considerazioni non corrette.

### 1.1 Obiettivi

Questo progetto di tesi nasce dall'esigenza di un'azienda privata in ambito utility di analizzare il proprio Sistema Informativo con il duplice scopo di effettuare una diagnosi dell'attuale e progettare una possibile soluzione ottimale. Andando nello specifico, il progetto è stato suddiviso in due parti: la prima comprende tutta la fase di analisi del SI con la relativa diagnosi, mentre la seconda, ben più verticale, tratta la progettazione e prototipazione di un Data Mart per la gestione delle informazioni all'interno dell'azienda.

## 1.2 Il dominio in analisi

La fase di analisi viene effettuata su tutto l'ecosistema dell'azienda, che come detto, lavora in ambito utility. Le attività che appartengono a questo settore del commercio trattano l'erogazione di servizi, come ad esempio energia elettrica o gas metano, con i clienti finali. Il dominio nel quale si trovano a lavorare queste aziende si presenta molto dinamico e in continua evoluzione. Da diversi anni ormai è in atto la cosiddetta *rivoluzione verde* che mira a migrare la produzione di energia dai combustibili fossili alle fonti rinnovabili quali vento, acqua e così via. Un'ulteriore fattore di dinamicità del settore è stato portato dalla pandemia. Il prezzo dell'energia, dal 2019 in avanti, subisce nell'arco di brevissimo tempo fluttuazioni consistenti, raggiungendo picchi mai toccati prima. Infine, si deve considerare la dinamicità che hanno portato le tecnologie. Anche in questo ambito, infatti, basti pensare ai contatori di nuova generazione che forniscono automaticamente le letture, la mole di dati da gestire e processare cresce e cambia velocemente nel tempo.

## 1.3 Struttura della tesi

Questo progetto di tesi è stato suddiviso in ben quattro capitoli. Il primo tratta la prima parte del progetto, che comprende l'analisi del Sistema Informativo dell'azienda e la relativa diagnosi. I successivi due capitoli narrano la seconda parte della tesi; in particolare, il secondo capitolo esegue una panoramica delle tecnologie che verranno utilizzate, mentre il seguente parla di come è stato progettato e realizzato il prototipo del Data Mart. Infine, la tesi si conclude con il capitolo che tira le fila sul progetto, descrivendo i risultati che sono stati raggiunti e i possibili sviluppi futuri.

# Capitolo 2

## Analisi SI aziendale

La prima sezione di tesi fa parte di un progetto multidisciplinare di **trasformazione digitale**.

### 2.1 Il progetto

Il progetto nasce dalla richiesta dell'azienda di rivedere e razionalizzare il proprio ecosistema, formato da processi, asset digitali e figure professionali. In breve tempo la previsione è di aumentare notevolmente il portafoglio clienti; questo implica da un lato una maggiore complessità di gestione, dall'altra un aumento consistente delle informazioni e dei dati che circolano all'interno dei processi. A tutto ciò va aggiunto anche che, nella maggior parte dei casi, un'organizzazione sbagliata dei processi aziendali potrebbe portare ad interruzioni nell'operatività e a limitazioni consistenti in fase di analisi strategica.

Per poter competere sul mercato l'azienda si trova di fronte a diverse sfide. Innanzitutto, l'aumento ipotizzato del numero di clienti deve essere correlato con un'automatizzazione dei processi. In secondo luogo, occorre limitare le inefficienze di qualsiasi tipologia attraverso la digitalizzazione e la revisione dei processi in chiave digitale. Infine, sono necessarie figure professionali che siano in grado di governare le tecnologie e tutti gli asset digitali. All'interno di questi, viene definita questa prima parte di tesi. Nello specifico, solo un sottoinsieme dei requisiti viene preso in esame e dettagliato nello specifico. La parte che viene considerata rappresenta tutto ciò che ha a che fare con l'organizzazione del Sistema Informativo dal punto di vista tecnologico e digitale. Esula da questo

ambito, dunque, andare nello specifico sulle questioni di modello di business e delle sue performance.

## 2.2 Obiettivi

All'interno del progetto, viene definita questa prima parte di tesi. Nello specifico, solo un sottoinsieme dei requisiti viene preso in esame e dettagliato nello specifico. La parte che viene considerata rappresenta tutto ciò che ha a che fare con l'organizzazione del Sistema Informativo dal punto di vista tecnologico e digitale. Esula da questo ambito, dunque, andare nello specifico sulle questioni di modello di business e delle sue performance. Compresi i limiti entro i quali si muove questa prima parte di progetto di tesi, vengono definiti, nello specifico, i seguenti obiettivi.

- **Business Process Mapping** del Sistema Informativo attuale.
  - Identificare le macro-attività dei processi e le micro-attività in cui possono essere scomposte.
- Analisi dello stato attuale degli asset digitali.
  - Delineare l'infrastruttura informatica dal punto di vista tecnologico.
  - Valutare la copertura dei processi mappati.
  - Identificare problemi e opportunità di digitalizzazione.
- Diagnosi.
  - Elementi critici dei processi e opportunità di miglioramento.
  - Criticità e opportunità legate alla *digital transformation/evolution*.
  - Criticità sulle figure professionali.
  - Criticità e punti di forza degli asset digitali.
- Identificazione del percorso strategico per la trasformazione digitale.
- Identificazione delle figure professionali da acquisire.

## 2.3 Kick-off meeting

Una volta raggiunta l'approvazione a partire, viene organizzato il **Kick-off meeting**. Questa riunione sancisce di fatto l'inizio della parte operativa dell'intero progetto e viene effettuata con lo scopo di far conoscere le parti coinvolte. A questa sono invitati a partecipare:

- **committente**, composto da:
  - sponsor di progetto, in questo caso il Direttore;
  - responsabile operativo dell'azienda;
  - responsabili di tutti gli uffici presenti.
  
- tutti i membri del **team di progetto**.

### 2.3.1 Committente di progetto

Il committente del progetto è l'azienda: partendo dallo sponsor del progetto, ovvero dal Direttore, vengono descritte tutte le figure manageriali che la compongono. In particolare, ciascun responsabile effettua una prima veloce descrizione di che cosa si occupa l'ufficio che gestisce.

### 2.3.2 Team di progetto

La composizione del team di progetto, viene descritta nella tabella 2.1. In particolare si può notare che le figure che ne fanno parte hanno conoscenze in due diversi ambiti in modo da poter fornire un contributo completo sotto diversi aspetti.

In base alle competenze, si sono delineati da subito due sottogruppi: uno in ambito Sistemi Informativi, più tecnologico, e l'altro in ambito Business Analytics e Business Process Mapping, quindi più focalizzato su misurare inefficienze organizzative e di processo.

### 2.3.3 Pianificazione

L'azienda si trova in una situazione di crescita costante e le previsioni prevedono un aumento considerevole del portafoglio clienti anche nei prossimi anni. Per far fronte a questa

Conoscenza/competenze	Esperienza
Sistemi Informativi, Big Data, Machine Learning	Elevata
Business Analytics, Performance Management Systems	Elevata
Business Process Mapping, Financial Accounting	Media
Business Process Mapping, Financial Accounting	Media
Sistemi Informativi, Business Intelligence	Bassa

Tabella 2.1: composizione team di progetto.

mole in aumento, l'azienda necessita di una organizzazione più efficiente ed efficace degli attuali processi aziendali e dei relativi asset digitali. Dopo questa premessa, viene definita l'organizzazione per la fase operativa di analisi e raccolta delle informazioni. Lo strumento principale per effettuare ciò è l'**intervista**, in quanto, così facendo, sarà possibile raccogliere anche considerazioni, riflessioni e suggerimenti. L'approccio alle interviste sarà di tipo *top-down*: le prime ad essere interrogate saranno le figure manageriali dei vari uffici per poi concludere il giro con gli operativi. Le informazioni necessarie per le interviste sono riportate in tabella 2.2. Ogni figura all'interno del team dovrà avere tutte le informazioni in 2.2 per ogni intervista che verrà eseguita.

Per quanto concerne le tempistiche, il team di progetto fornisce il **diagramma di Gantt** in figura 2.1. Il diagramma di Gantt[1] è tra gli strumenti più famosi ed utilizzati in ambito *Project Management*, in quanto, oltre a mostrare una stima della durata delle attività, definisce anche il legame tra le stesse.

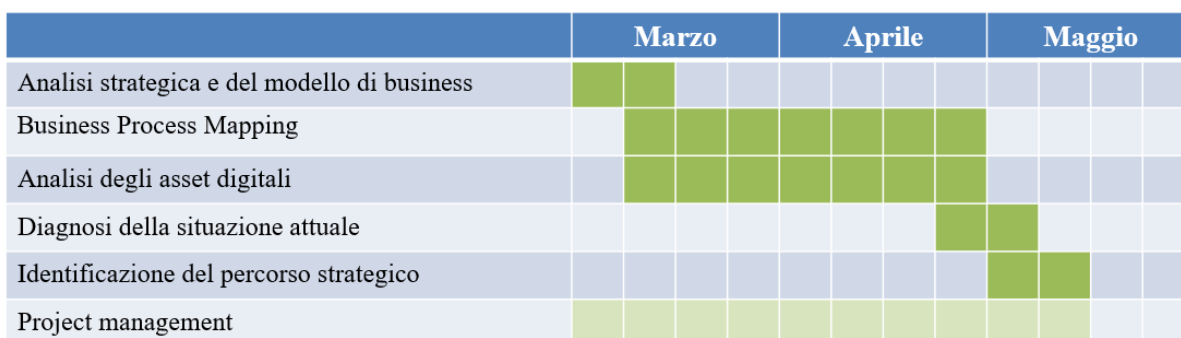


Figura 2.1: diagramma di Gantt per la gestione di progetto.

Come si può notare dalla figura 2.1, ad esempio, le fasi di *Business Process Mapping* e di *Analisi degli asset digitali* possono essere svolte in parallelo. Queste due attività, infatti,



Soggetto	Data Nominativo Area e ruolo Descrizione area
AS IS	Competenze digitali Asset digitali utilizzate Problemi, rischi e opportunità Note
TO BE	Vantaggi che si aspetta dalla digitalizzazione Tecnologie più interessanti Ambiti da digitalizzare Note
Sensazioni	Clima dell'intervista Punti di attenzione

Tabella 2.2: template per interviste su asset digitali.

verranno svolte in parallelo dai due sotto-team che compongono il gruppo di lavoro: la prima verrà seguita dal team con conoscenze in ambito economico, mentre la seconda dalle figure più esperte in ambito digitale/tecnologico. Inoltre, si può evincere che la fase di *Diagnosi della situazione attuale* potrà avvenire solo al termine delle interviste, così come il task *Identificazione del percorso strategico* avrà inizio dopo la fine di quest'ultima. Occorre sottolineare, infine, che:

- durante tutte le fasi di progetto verrà portata avanti anche una parte di *Project management* con monitoraggio e controllo delle attività;
- il progetto ha durata trimestrale e deve essere svolto nei mesi di marzo, aprile e maggio;
- le ultime due settimane di maggio sono state tenute da parte per gestire eventuali ritardi.

Per ultimo occorre definire i *deliverable* di progetto, che comprendono:

- **Business Performance Analysis:** riporta l'analisi dei processi aziendali, con particolare enfasi su criticità ed inefficienze riscontrate;
- **Digital Asset Analysis:** analizza il portafoglio applicativo aziendale e sottolinea problematiche rilevate in ambito digitale;
- **Analisi Strategica:** delinea la situazione attuale, con riferimenti alle azioni da seguire in ambito digitalizzazione (comprende progetti, asset digitali e figure professionali).

Queste informazioni saranno inviate all'azienda cliente attraverso dei file Excel o PowerPoint al termine del progetto.

## 2.4 Analisi

La prima fase di questa sezione prevede, dunque, l'analisi dell'attuale Sistema Informativo in termini di asset digitali e strumenti tecnologici. Esula da questi scopi, dunque, riportare tutte le interviste e le analisi effettuate in ambito economico. Di conseguenza viene mostrato solo il lavoro effettuato dal sotto-team che si occupa dei Sistemi Informativi.

### 2.4.1 Interviste con responsabili

In questa fase vengono presi in considerazione tutti gli uffici presenti all'interno dell'azienda, andando ad intervistare il relativo manager. In tabella 2.3 viene mostrato l'ordine delle interviste.

Ordine	Intervistato
1	Responsabile IT
2	Responsabile Operativo
3	Responsabile Back-office e Front-office
4	Responsabile Marketing
5	CEO

Tabella 2.3: ordine interviste responsabili.

In questa fase le domande vengono effettuate seguendo una metodologia a **piramide**; questo metodo segue un approccio induttivo, nel quale l'intervistatore parte da domande dettagliate per superare eventuali difficoltà nell'intervistato, il quale potrebbe provare scetticismo nei confronti delle domande sottoposte. Le interviste si concludono poi con questioni più ad ampio spettro e con dei suggerimenti su come il manager vorrebbe che fossero digitalizzati i processi di cui è responsabile.

### 2.4.2 Interviste con operativi

In questa seconda fase, le figure da intervistare non hanno ruoli manageriali bensì operativi. Questi permettono di mostrare i processi aziendali da un lato differente rispetto a ciò che è stato narrato dai responsabili, in modo da avere un quadro quanto più completo e dettagliato.

Lo schema da seguire per queste interviste è di tipo deduttivo e viene chiamato a **imbuto**. Per mettere a proprio agio gli intervistati, questa metodologia prevede di partire con domande a risposta aperta e molto generali, per poi terminare con questioni più specifiche. In figura 2.2 si può notare la differenza tra quest'ultima metodologia e quella utilizzata precedentemente con i responsabili dei vari uffici.



Figura 2.2: esempio di interviste a piramide e di interviste a imbuto.

## 2.5 Diagnosi

Terminata la parte di interviste, si procede con la fase di diagnosi del Sistema Informativo. In questo momento, il team di lavoro si prende del tempo per rielaborare tutti gli appunti presi e, una volta consolidati, si riunisce in varie riunioni. Queste hanno lo scopo di definire con precisione tutte i *deliverable* che erano stati descritti all'inizio del progetto.

### 2.5.1 Business Performance Analysis

In questa fase iniziale il team si concentra nella definizione del deliverable che descrive i processi aziendali. Occorre sottolineare che esula da questo progetto di tesi, analizzare le performance di ogni fase, ma ci si sofferma ad un'analisi di massima dei processi aziendali. Tra gli scopi di questo progetto di tesi, infatti, è stato definito di effettuare solo una identificazione delle attività per poter poi effettuare una diagnosi degli asset digitali.

Per poter descrivere i processi aziendali correttamente, vista la natura di azienda che commercializza servizi per i cittadini, è stato deciso di identificare la **Catena del Valore di Porter**[2] come in figura 2.3. Questa rappresentazione viene descritta dallo stesso Porter nel 1985 così: *"Every firm is a collection of activities that are performed to design, produce, market, deliver, and support its product. All these articles can be represented using a value chain"*. Inoltre, definisce tre tipologie di *activity*:

- **primarie:** ovvero tutti i macro-processi principali all'interno di un'azienda;
- **di supporto ai processi:** attività che forniscono input, risorse, ecc. alle attività principali;
- **di supporto all'azienda:** cioè tutti quei task che hanno a che fare con l'ecosistema aziendale, come ad esempio fornitori e distributori.

Fornita la definizione teorica di *catena del valore*, in questo specifico caso è stata definita la soluzione in figura 2.3.

In figura 2.3 sono state definite le attività principali e quelle di supporto. Queste, però, definiscono solo parzialmente e in maniera fin troppo superficiale le macro-attività che sono presenti all'interno dell'azienda. A questo punto, dunque, occorre dettagliare ciascuna di esse in micro-attività descrivendo anche eventuali inefficienze e criticità a livello di processo.

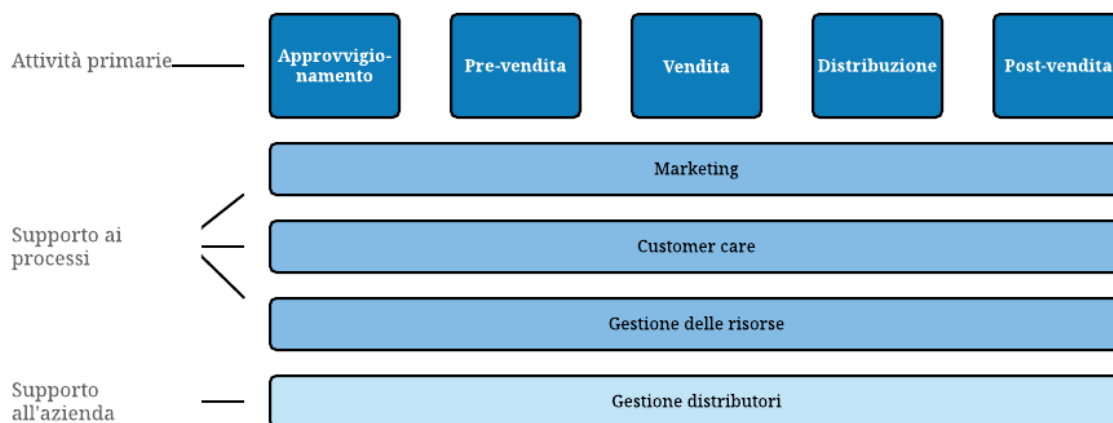


Figura 2.3: catena del valore di Porter dell'azienda.

**Approvvigionamento** Comprendono tutto ciò che è collegato all'acquisizione delle materie prime, che in questo caso consistono in energia elettrica e gas metano. Comprende le seguenti micro-attività:

- *stima della quantità di materia prima da acquistare;*
- *analisi di mercato;*
- *analisi del valore in borsa;*
- *gestione dei fornitori.*

Le criticità riscontrate in questa fase risultano essere legate soprattutto alla stima della materia prima da acquistare. Questo valore, infatti, risulta essere molto importante per ridurre i costi. Periodicamente l'ufficio preposto a tale scopo deve definire con quanta più esattezza possibile la materia prima che andrà ad utilizzare per il prossimo periodo, attraverso una comunicazione ai relativi distributori di energia. A livello informatico, questo calcolo viene effettuato con delle interrogazioni molto intensive sul database di fatturazione, per poi essere elaborate con dei fogli di Excel. Questa gestione presenta delle notevoli inefficienze in quanto le lunghe interrogazioni rallentano non solo il processo, bensì anche gli altri operativi che stanno utilizzando le stesse informazioni. Inoltre, l'utilizzo di Excel con grandi quantità di dati risulta essere spesso non percorribile. Le altre attività

son difficilmente migliorabili in termini di efficienza attraverso la digitalizzazione, e non rientra perciò tra questi scopi evidenziarne le criticità.

**Pre-vendita** Questa macro-attività definisce la fase nel quale il cliente non è ancora in fornitura con l'azienda. Può essere suddivisa in:

- *ricerca del cliente;*
- *contrattualizzazione;*
- *gestione dei canali di vendita;*
- *gestione dei venditori.*

Questa parte della *customer journey* risulta essere sicuramente chiave per quanto riguarda l'azienda. Rendere maggiormente efficiente, dunque, tale sezione porterebbe giovamento a tutti gli uffici all'interno dell'azienda. In questa fase sono presenti molti stakeholder in outsourcing, e non essere in grado di monitorare il loro andamento porta sicuramente a dei limiti di efficienza.

In questo caso le criticità son legate soprattutto alla mancanza di uno strumento (e di un processo) per raccogliere le informazioni ed analizzarle. Attualmente, soprattutto chi ricopre ruoli manageriali, per creare della reportistica utilizza delle esportazioni definite su viste non parametrizzabili. Qualora si vogliano lavorare i dati, ciò viene effettuato con dei fogli di Excel. Le analisi, dunque, risultano essere limitate, o addirittura impossibili da effettuare in quanto, talvolta, non sono stati raccolti nemmeno i dati.

**Vendita** Al suo interno confluiscono tutti i controlli che vengono effettuati prima di far entrare in fornitura il cliente. Le sue attività sono:

- *controllo di morosità;*
- *analisi qualità della vendita;*
- *controllo numero di switch del cliente;*
- *controllo dati anagrafici;*
- *controllo dati catastali.*

Prima di ufficializzare l'entrata in fornitura di un cliente, vengono eseguiti una serie di controlli. Questi, vista la numerosità crescente dei clienti, vengono in parte affidati ad outsourcer esterni che lavorano la pratica e forniscono il risultato all'ufficio preposto. Anche qui, vengono spesso riscontrate problematiche e limitazioni dovute ad una mancata digitalizzazione. In primo luogo, viene riportato che la difficoltà nel reperire e lavorare le informazioni genera rallentamenti anche in queste procedure. Successivamente, un'altra fonte di inefficienza importante è data dalla pluralità di sistemi informatici che vengono utilizzati. Per effettuare questa serie di controlli, infatti, l'operatore deve avvalersi di diverse applicazioni che non comunicano tra di loro. Questo, oltre che generare scontento nel lavoratore, produce anche un *overhead* di gestione delle pratiche.

**Distribuzione** Superati i controlli, il cliente entra in fornitura con l'azienda e viene effettuata la vendita vera e propria. Questa parte può essere suddivisa in:

- *vendita;*
- *gestione delle comunicazioni verso il distributore;*
- *gestione dello Switch;*
- *gestione delle comunicazioni dal distributore.*

Questa fase risulta essere ben più snella rispetto alle altre, anche se prevede anch'essa le relative criticità. In particolare, la principale problematica è la mancanza di un'unica anagrafica. A causa della crescita destrutturata dell'azienda in breve tempo, questa si è trovata a dover sdoppiare la propria base dati in due parti dove una funge da *Customer Relationship Management* e l'altra da sistema di *Billing*. A causa di questa duplice sorgente, quando un cliente entra in fornitura con l'azienda, nasce una micro-attività definita **Switch** che copia i dati da dove è stato inserito il contratto al sistema di fatturazione. Per quanto riguarda la gestione dell'informazione da fornire al distributore non sono state riscontrate particolari criticità sotto questi aspetti.

**Post-vendita** In questa fase il cliente è in fornitura effettivamente con l'azienda. I sotto-processi sono:

- *gestione pratiche di connessione;*

- *fatturazione;*
- *gestione delle comunicazioni del cliente;*
- *gestione delle autoletture;*
- *gestione delle letture inviate dai distributori;*
- *gestione degli incassi;*
- *gestione delle morosità;*
- *gestione degli indennizzi.*

La Catena del Valore di Porter si conclude con tutti i servizi che l'azienda fornisce al cliente una volta che è entrato in fornitura. Per quanto riguarda ciò, le principali criticità sono dovute al sistema di *ticketing* utilizzato per tracciare tutte le comunicazioni. Per sviluppare questo software è stata utilizzata una versione del sistema open-source chiamato **Redmine**[3]. Questo strumento, a tratti non performante, risulta essere un ulteriore collo di bottiglia per gli operativi in quanto si trovano ad utilizzare un ulteriore applicativo e che presenta dei vincoli molto stringenti in termini di personalizzazione. Infine, occorre sottolineare che importanti limitazioni, in questa fase, sono riscontrate a livello di lavorazione delle informazioni. Per poter effettuare analisi sulle letture dei contatori dei clienti, ad esempio, occorre gestire una notevole quantità di dati che con gli strumenti attualmente in uso risulta essere del tutto impossibile.

### 2.5.2 Digital Asset Analysis

Il secondo deliverable deve descrivere, con quanti più dettagli possibili, l'intero **Portafoglio Applicativo** aziendale. Come definito in [4], con questo termine si intende l'insieme di tutte le applicazioni usate all'interno di un ecosistema, cioè tutti gli specifici sistemi informatici che forniscono supporto diretto al business. Questa definizione non lascia spazio ad equivoci e definisce, quindi, che i soggetti dell'analisi non saranno solo un sottoinsieme dei sistemi bensì tutto ciò che in qualsiasi modalità contribuisce a raggiungere gli obiettivi aziendali.

Per poter definire con esattezza il portafoglio applicativo occorre aver analizzato i processi aziendali con un adeguato livello di dettaglio. Partendo dall'analisi effettuata



in 2.5.1, si è definito in figura 2.4 la suite di moduli informatici utilizzati all'interno dell'azienda.

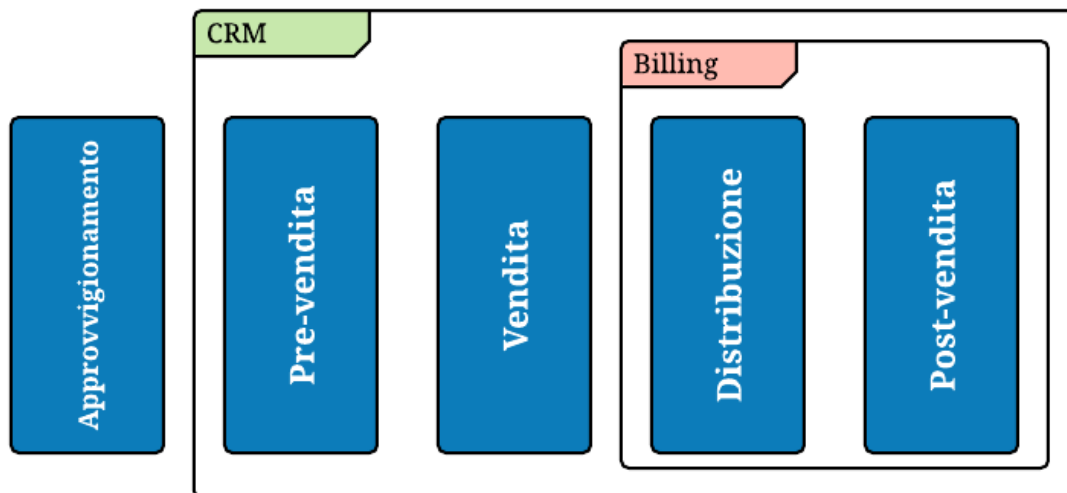


Figura 2.4: moduli che compongono il Portafoglio Applicativo.

Per comprendere meglio il diagramma in figura 2.4, occorre definire gli strumenti software che utilizza l'azienda. Occorre subito sottolineare che al momento nell'azienda non è presente un **Enterprise Resource Planning (ERP)**[5]. La mancanza di questo sistema in grado di gestire l'intera gamma di processi porta con sé svariate mancanze all'interno dell'azienda. Prima di tutto non è presente *univocità dell'informazione*, in secondo luogo risultano difficili inserimenti di moduli incrementalmente ed, infine, ciò porta ad una mancanza di *prescrittività*. Queste lacune si notano, operativamente, nelle decine di controlli ridondanti e *off-system*, nella complessità di gestione dei processi e così via.

Un secondo aspetto da sottolineare è l'utilizzo del **Customer Relationship Management (CRM)** che ricopre, come in figura 2.4, un numero eccessivo di processi. Infatti, come da definizione in [6], il CRM si dovrebbe occupare della gestione strategica del cliente. In questo caso, invece, viene utilizzato in molti processi all'interno dell'azienda, portando con sé inefficienze e limitazioni. Infatti, i dati di profilazione dei clienti, solo per fare un esempio, sono presenti in maniera molto limitata e non forniscono i dettagli necessari per effettuare un'analisi adeguata. Un'ulteriore fattore di criticità proviene da

chi ha la responsabilità di gestione di tale strumento. La creazione e gestione di questo strumento software è stata commissionata, qualche anno fa, ad una software house esterna molto piccola (5-10 dipendenti). Seppur mantenendo costi di gestione limitati, questo risulta un notevole fattore di rischio strategico dell'azienda perché eventuali problematiche sotto questo aspetto potrebbero causare rallentamenti alla maggior parte dei processi. All'interno dell'azienda, inoltre, non sono presenti esperti di Sistemi Informativi che ne governano la gestione.

Oltre a questo software, l'outsourcer esterno ha creato anche il sistema di ticketing partendo dal prodotto open-source *Redmine* citato precedentemente. Limitazioni sotto questo aspetto sono soprattutto legate alle prestazioni della piattaforma che risulta, ormai, obsoleta.

Infine, è presente il sistema di billing utilizzato nelle attività di *fatturazione* e di *gestione degli incassi*. Questo si presenta come un software ben organizzato, creato da una società esterna notevolmente più strutturata rispetto alla precedente. Qui si presentano limiti in termini di univocità del dato in quanto, comunicando parzialmente con il sistema CRM, le informazioni si trovano ad essere duplicate su entrambi i sistemi. Questo genera, inutile sottolineare, una quantità elevata di inefficienze e di errori all'interno dei processi.

Conclusa la panoramica sugli attuali sistemi in uso all'interno dell'azienda, è possibile definire in figura 2.5 l'attuale architettura funzionale del SI.

Come si può notare i clienti possono comunicare con l'azienda in più punti, facendo cadere la definizione di CRM assegnata inizialmente ad uno dei software. Si può notare, inoltre, attraverso la visualizzazione di due database differenti come si traduce a livello architetturale la *doppia anagrafica*.

### 2.5.3 Analisi Strategica

Compreso il dominio e le relative problematiche in ambito digitale, questo ultimo deliverable propone un insieme di azioni da seguire per migliorare i processi dell'azienda.

L'attività principale da portare avanti immediatamente risulta essere la modifica ed il conseguente efficientamento dell'architettura funzionale del SI. In figura 2.7 si può notare la soluzione che è stata proposta all'azienda. Questa prevede di eliminare notevolmente le criticità evidenziate precedentemente. Lo si può notare dalla istituzione di un modulo CRM ben definito, ad esempio. Questo dovrà convogliare al proprio interno tutte le

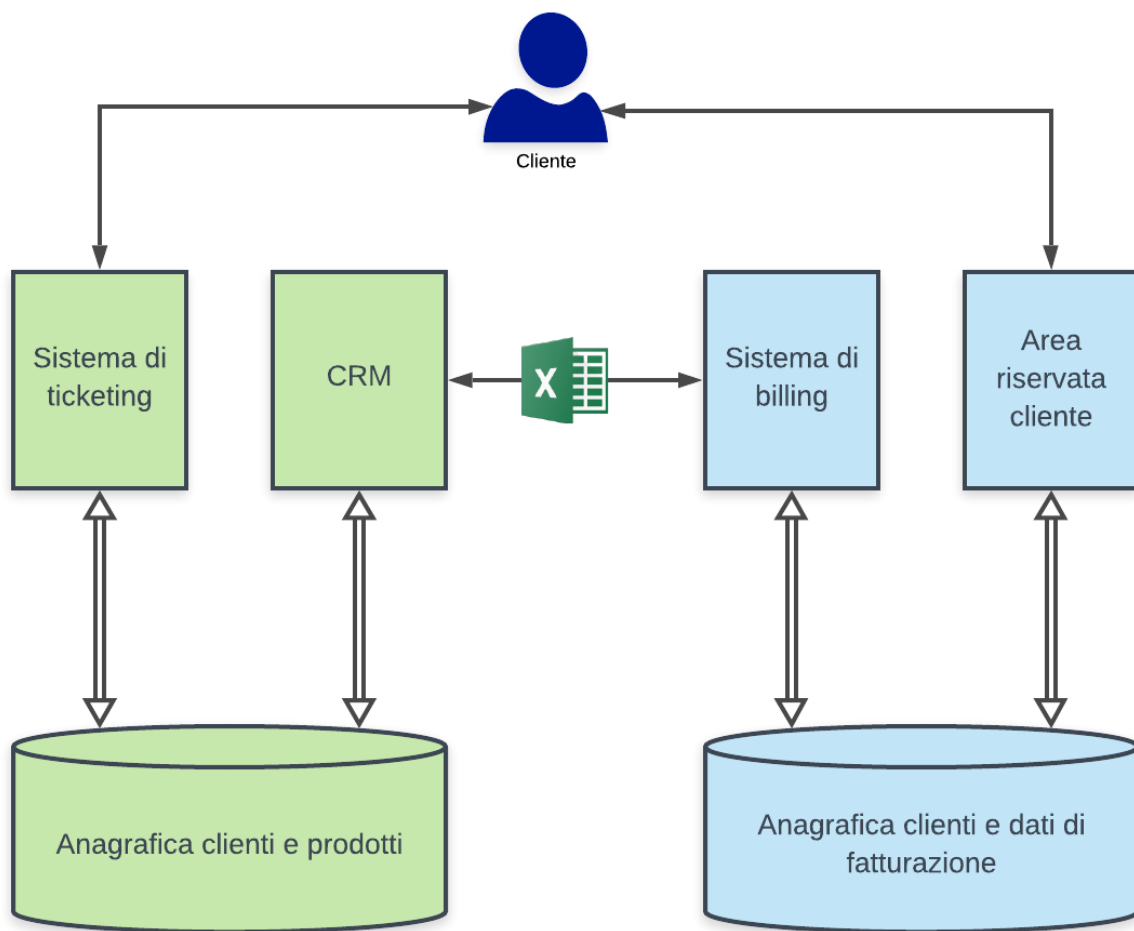


Figura 2.5: architettura funzionale SI.

funzionalità presenti nei processi che comunicano con il cliente, fornendo un unico punto di accesso.

Un'ulteriore modifica che deve essere fatta per migliorare le prestazioni dei processi aziendali è rappresentata dalla creazione di un sistema di **Data Warehouse (DW)**. Questo database ha il compito, come descritto in [7], di collezionare informazioni provenienti da più sorgenti eterogenee tra di loro e di supportare interrogazioni ed analisi strategiche. In questo modo si avrà un notevole miglioramento delle performance dei processi, in quanto si andrà ad alleggerire il carico di lavoro sulle sorgenti e, allo stesso tempo, si fornirà un'unica versione normalizzata del dato che sia facilmente esaminabile. Per poter avere questo strato di informazioni che sia sempre aggiornato, corretto e interrogabile, questa soluzione prevede la creazione di un ulteriore database chiamato **Operational Data Store (ODS)**. Questa soluzione prevede molteplici vantaggi:

- necessità di avere un'area all'interno della quale vanno normalizzati e puliti i dati prima di essere inseriti all'interno del DW;
- possibilità di creare reportistica operativa per alleggerire il carico dal DW;
- libertà di aggiungere funzionalità personalizzabili.

Per poter realizzare questa soluzione, occorre istituire anche una vera e propria **divisione Sistemi Informativi**. Attualmente mancante all'interno dell'azienda, questa avrà il compito di gestire tutti i progetti che riguardano il SI aziendale, ma anche di governare dati, processi e asset digitali. In base alla grandezza e agli obiettivi dell'azienda, una soluzione ottima per questa parte è raffigurata in figura 2.7.

Va sottolineato che questa soluzione si presenta come *ottima* e che possono essere prese in considerazione, almeno inizialmente, anche decisioni parziali di tale organigramma.

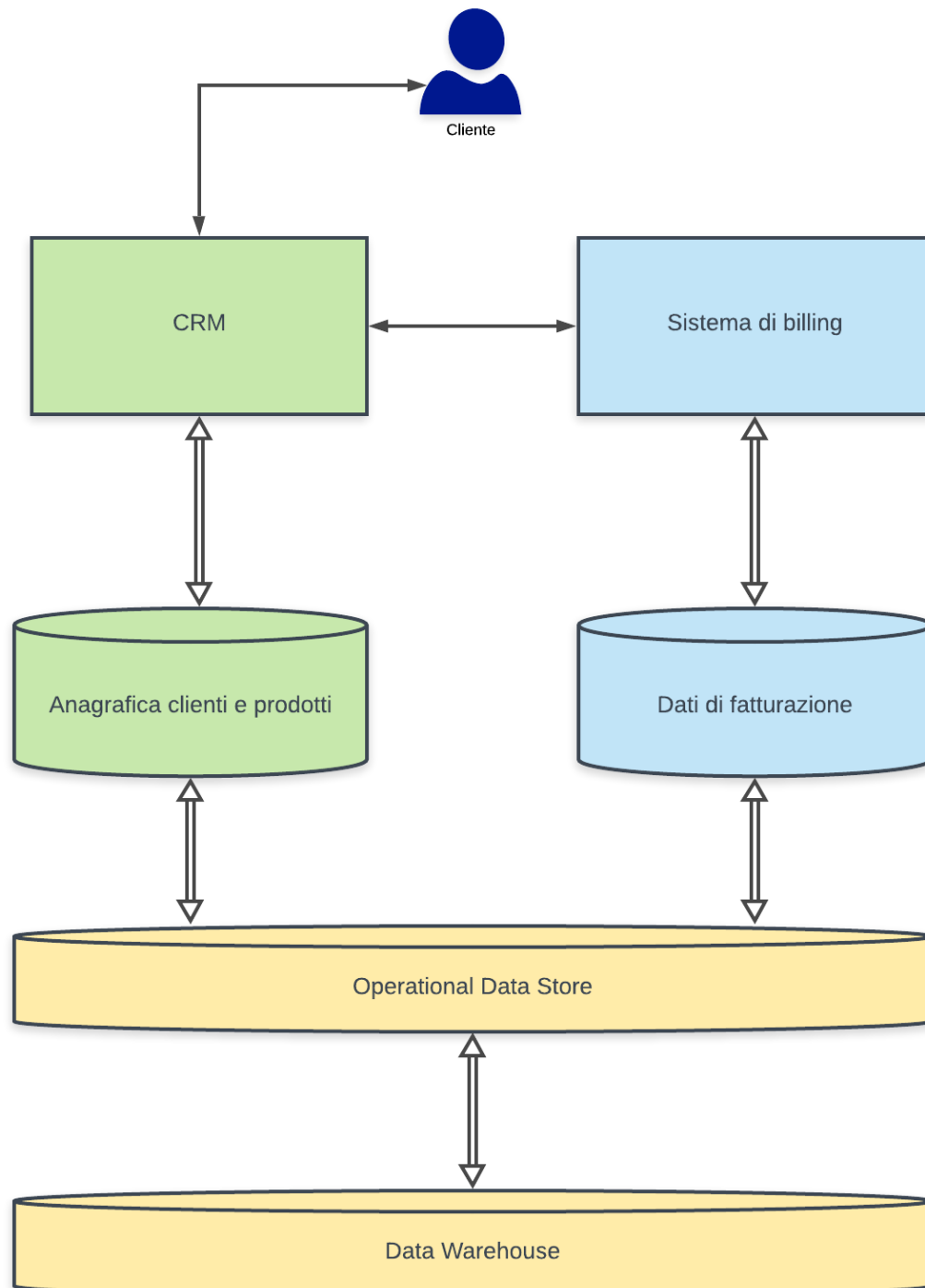


Figura 2.6: proposta della nuova architettura funzionale del SI.

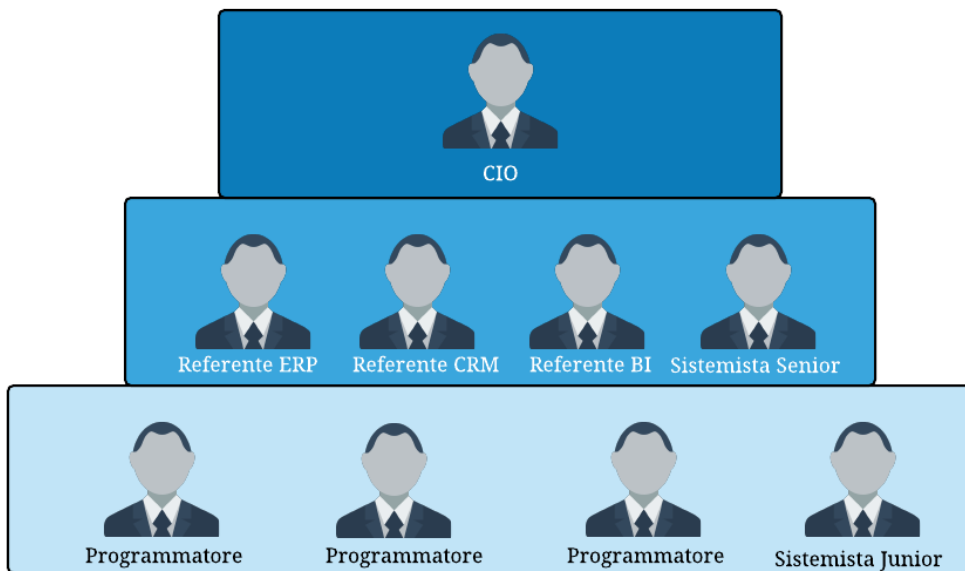


Figura 2.7: proposta della nuova divisione Sistemi Informativi.

# Capitolo 3

## Tecnologie utilizzate

La seconda parte di tesi si apre con una panoramica sulle tecnologie utilizzate in questa sezione. La scelta di questi strumenti è stata dettata dalle indicazioni dell'azienda che ha richiesto la progettazione del Data Mart.

### 3.1 Architettura

Per prima cosa occorre definire come è organizzata attualmente l'azienda evidenziando le tecnologie utilizzate. Come evidenziato nella prima parte del progetto di tesi, le informazioni che circolano nell'azienda sono suddivise all'interno di due database distinti, gestiti da due outsourcer differenti. Questi due, in particolare come definito in figura 3.1, utilizzano diversi **Database Management System (DBMS)**: con questo termine si indica il sistema software progettato per consentire la creazione, la manipolazione e l'interrogazione efficiente dei database.

Il focus di questa seconda parte di tesi, come già descritto, rappresenta la progettazione e prototipazione di un Data Mart. A causa delle limitate risorse, soprattutto in quanto a tempo disponibile, si è deciso di progettare il Data Mart partendo dalle informazioni contenute nel database con i dati sui clienti e sui prodotti. In questo progetto, dunque, le banche dati che contengono le informazioni operative utilizzano il DBMS **MySQL**[8]. Questo servizio, ad oggi, è il più utilizzato motore di database relazionali nel mondo. I motivi di questo risultato son da ricercare soprattutto sulla sua natura *open-source*, oltre ad una corretta gestione della sicurezza e delle performance rispetto ad altre soluzioni SQL. D'altrocanto, evidenzia tutti i limiti dei database relazionali soprattutto in termini

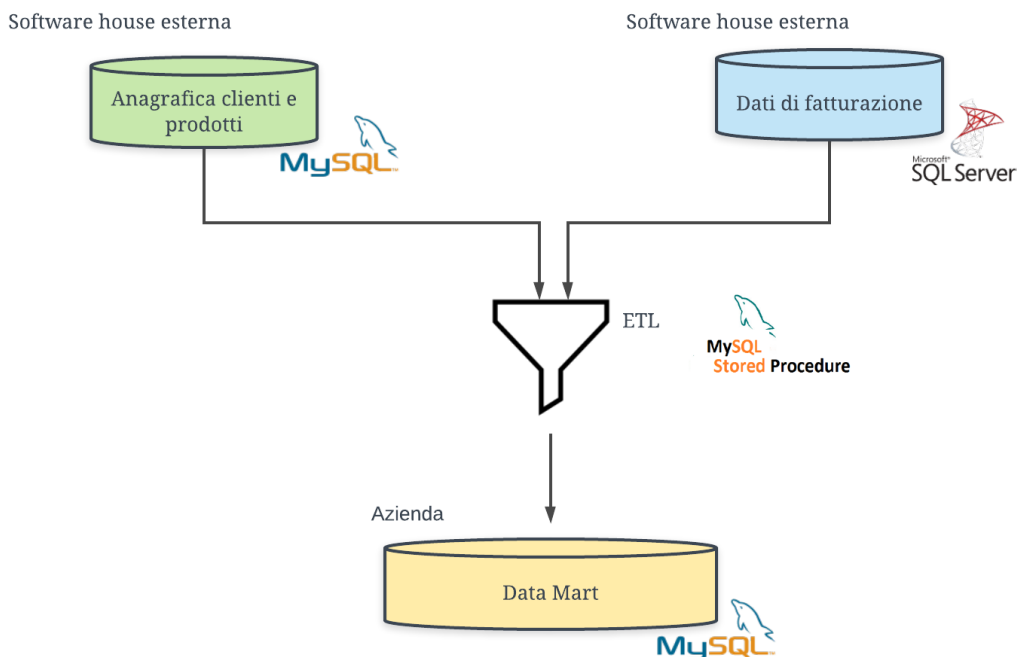


Figura 3.1: tecnologie dei database sorgenti e di destinazione.

di scalabilità e *throughput* nei confronti delle versioni NoSQL[9]. Questi stessi limiti sono presenti anche nel DBMS **SQL Server** in quanto anch'esso di tipo relazionale. In particolare, però, occorre notare che esistono diverse peculiarità che lo contraddistinguono nei confronti di MySQL. Senza scendere troppo in dettagli tecnici, la differenza principale è che il primo, sviluppato da Microsoft, non è utilizzabile gratuitamente come il secondo.

Per quanto concerne l'architettura fisica dei database, come definito in figura 3.2, essi si trovano in due condizioni distinte: il database contenente le informazioni legati ai clienti si trova *on-premise*, fisicamente dislocato all'interno della software house, mentre quello che contiene i dati di fatturazione si trova su *cloud* con una soluzione di tipo **Hardware As A Service (HAAS)**. Questo modello, come descritto in [10], permette di utilizzare l'hardware presente su *cloud* come se fosse un servizio. In questo modo è possibile integrare più sistemi fisici distribuiti in ogni parte del mondo in maniera trasparente per l'utilizzatore, con la caratteristica ulteriore di creare anche macchine virtuali. Questa soluzione presenta notevoli vantaggi, soprattutto legati alle performance in quanto, con notevole semplicità, è possibile scalare dinamicamente accendendo (o spegnendo) nuove macchine solo per i momenti necessari. Su di queste macchine, inoltre, è possibile in-



stallare e configurare tutti gli strumenti che possono essere necessari, come se fosse una macchina fisica, ma senza doversi preoccupare di tutti i problemi legati all'installazione e alla manutenzione degli apparati. La tecnologia che è stata utilizzata per questa soluzione è il **Google Cloud Storage**[11], ovvero la parte del colosso americano che permette di memorizzare in maniera efficiente grandi quantità di dati, garantendo sicurezza, scalabilità e performance elevate.

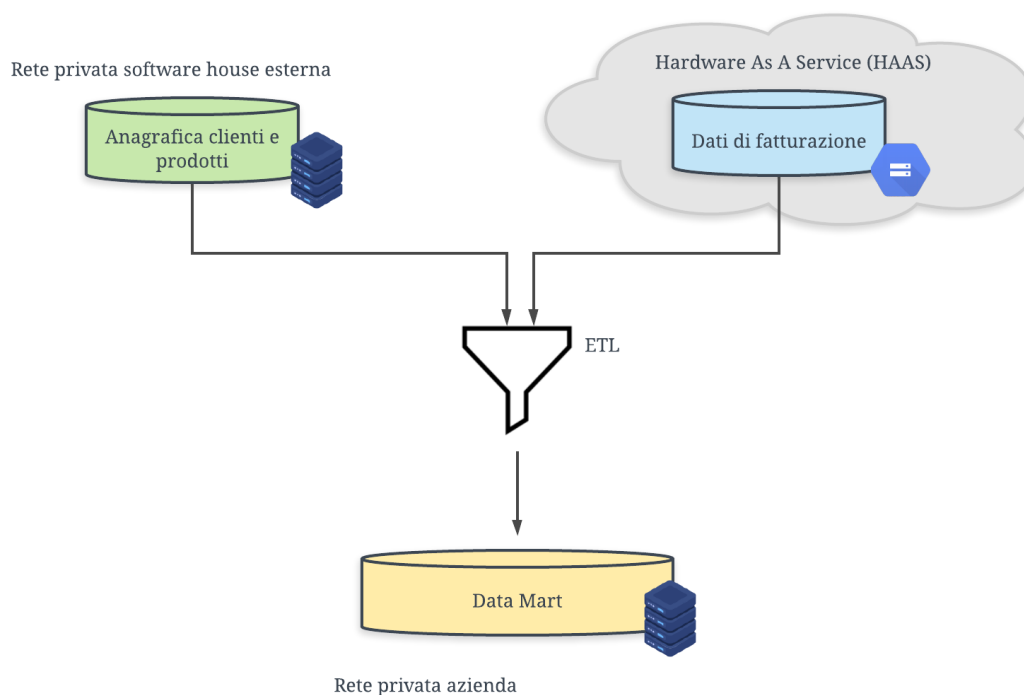


Figura 3.2: database *on-premise* e su *cloud*.

A livello tecnologico, lo strumento che è stato utilizzato per implementare l'ETL è, principalmente, il linguaggio **MySQL stored program language**[12] su database MySQL. In particolare, questo linguaggio mette a disposizione diversi costrutti che permettono di raggiungere gli obiettivi previsti. Il risultato finale, infatti, non sarà altro che una successione di **Stored procedure** che, in maniera temporizzata, andranno a leggere il database sorgente per popolare il Data Mart. Per quanto riguarda il database sul quale creare il prototipo del Data Mart, si è deciso, su suggerimento dell'azienda, di utilizzare un database MySQL interno al proprio perimetro.

## 3.2 MySQL Workbench

All'interno del mercato dei tool utili agli scopi definiti precedentemente, il responsabile IT dell'azienda ha suggerito l'utilizzo di **MySQL Workbench**. Questo tool permette di interrogare la sorgente della software house esterna, ma, in aggiunta, anche di creare e gestire database. In figura 3.3 è possibile notare come si presenta la *UI* dello strumento MySQL Workbench.

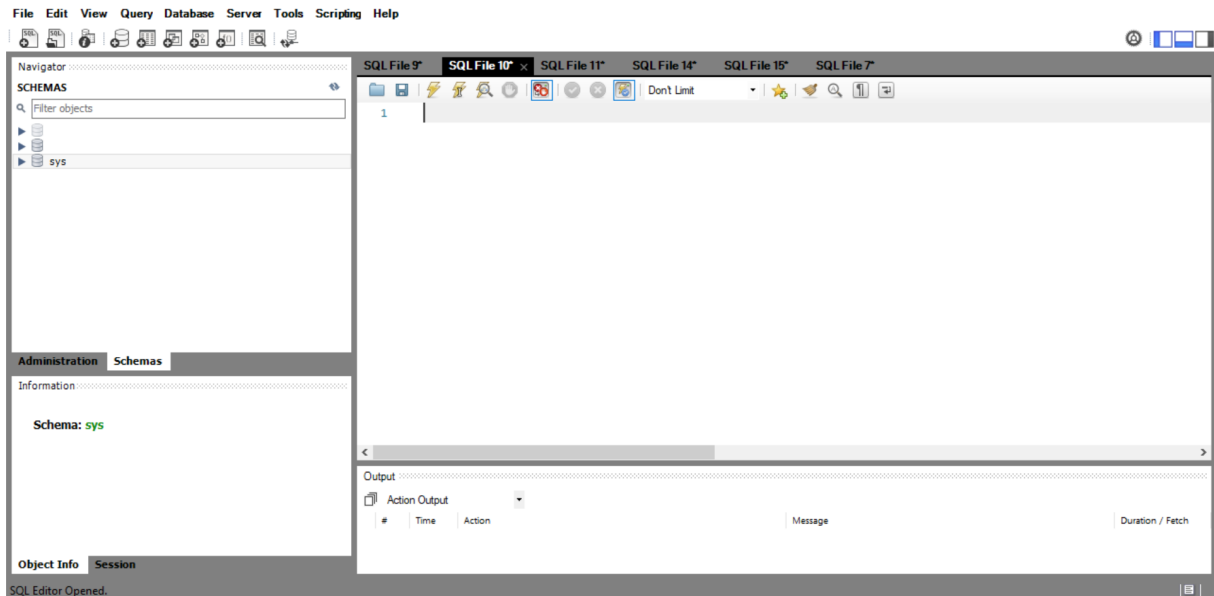


Figura 3.3: schermata di lavoro su MySQL Workbench.

## 3.3 Power BI

Il tool di *data visualization* rappresenta un ulteriore strumento che è utile in questa parte di tesi. Una volta elaborati i dati, infatti, risulta fondamentale avere la possibilità di visualizzarli graficamente. Questa necessità è dovuta al fatto che:

- grandi mole di dati potrebbero essere difficilmente visualizzabili senza l'utilizzo di tali strumenti;
- attraverso la rappresentazione grafica è possibile evincere pattern dai dati con facilità;
- l'uso di colori e/o forme differenti favorisce la comprensione delle informazioni;

- è necessario poter leggere facilmente le informazioni dovunque e con qualsiasi apparato;
- la gestione dei ruoli e dei permessi è fondamentale all'interno dell'azienda;
- occorre mantenere la *data freshness*;
- è necessario prevedere la possibilità di interagire con le informazioni.

Durante il progetto di tesi, all'interno dell'azienda sono presenti delle licenze di **Power BI**[13], perciò è stato scelto questo come tool di data visualization. Sviluppato da Microsoft, rappresenta uno degli strumenti di questo tipo più utilizzati in ambiti aziendali. Al suo interno prevede una suite di servizi che permettono di importare dati da più basi dati eterogenee tra di loro, lavorarli e visualizzarli in maniera semplice ed intuitiva. Tra le funzionalità più utilizzate, sono presenti quelle legate al mondo dell'**Artificial Intelligence (AI)** fornite dalla piattaforma cloud *Azure*. Essendo entrambi strumenti Microsoft, questi sono anche nativamente collegati, portando l'importante vantaggio di utilizzare *Azure* come sorgente dati e *Power BI* per l'analisi e la reportistica. Tra le funzioni occorre ricordare che *Power BI* permette di creare la reportistica attraverso la propria applicazione desktop esistente per tutti i principali sistemi operativi. Nei confronti della concorrenza, questo rappresenta un vantaggio importante in quanto altri strumenti, come ad esempio *Data Studio* di Google, sono completamente *browser-based*. Inoltre, è presente anche una folta *community* che contribuisce quotidianamente a segnalare malfunzionamenti e ad aumentare la quantità di visualizzazioni utilizzabili all'interno dello strumento.



# Capitolo 4

## Progettazione e prototipazione Data Mart

Allo stato attuale, all'interno dell'azienda non è presente un vero e proprio ufficio di Business Intelligence e le informazioni utilizzate per effettuare analisi vengono estratte ed elaborate in fogli Excel. I limiti di questo approccio son evidenti a chiunque, perciò l'idea di realizzare un sistema che sia in grado di elaborare e visualizzare le informazioni è stata subito reputata fondamentale.

Dunque, la seconda parte del progetto di tesi, ben più verticale e tecnologica, verte sulla progettazione di un Data Mart all'interno dell'azienda. Andando più nello specifico nella definizione di Data Warehouse rispetto a quanto fatto in 2.5.3, il DW si può definire come l'insieme di più **Data Mart**. Questi corrispondono ad una aggregazione di dati nel DW primario contenenti l'insieme delle informazioni rilevanti per una particolare area del business, una particolare divisione dell'azienda, una particolare categoria di soggetti. Per gli scopi di questa parte di tesi, dunque, ci si limiterà alla progettazione e prototipazione di solo un Data Mart, seppur progettando il DW in maniera tale da prevederne la creazione di altri in futuro. Fornita questa panoramica iniziale, il **ciclo di sviluppo** che si è seguito per la progettazione e prototipazione del Data Mart è rappresentato in figura 4.1.

### 4.1 L'area in analisi

Tutte le aziende sono organizzate in diversi uffici che svolgono spesso compiti molto diversi ed eterogenei tra loro. Ciascuno di essi presenta al proprio interno delle peculiarità e delle



Figura 4.1: il ciclo di sviluppo di un Data Mart.

necessità che possono anche essere molto differenti. Per poter procedere con questa parte di tesi, dunque, è stato deciso di analizzare all'interno dell'azienda quali siano le esigenze primarie in termini di reportistica operativa e direzionale.

Tra i *business process* più critici sotto questi aspetti, è stato rilevato che i dati che vengono creati ed elaborati durante il **Customer Management Process** hanno particolari necessità di essere gestiti. Questo insieme di attività si occupa della gestione del cliente in tutto il suo *journey* dalla pre-vendita alla post-vendita. Questa parte risulta essere fondamentale per il raggiungimento degli obiettivi in qualsiasi azienda, in quanto rappresenta il mezzo di comunicazione con il cliente. All'interno di tale processo circolano una quantità molto elevata di dati che, se correttamente analizzati, possono diventare un vero valore aggiunto per l'azienda. In maniera sempre più dirompente, negli ultimi tempi, la capacità di capire il comportamento e le abitudini del cliente è diventata una necessità dalla quale non ci si può esimere. Nell'ambito del commercio di servizi (come ad esempio energia elettrica e gas), risulta fondamentale analizzare tutti i momenti nei quali il cliente comunica con le aziende e, viceversa, tutte le risposte che gli vengono fornite. Una corretta gestione di queste attività può portare ad un miglioramento di tutti i settori dell'azienda e al raggiungimento di una *qualità del servizio* eccellente. Per gestire questo

processo, viene utilizzato il software di *ticketing* Redmine. In figura 4.2 viene riportato il flusso che attualmente l'azienda sta utilizzando per la gestione delle richieste dei clienti.

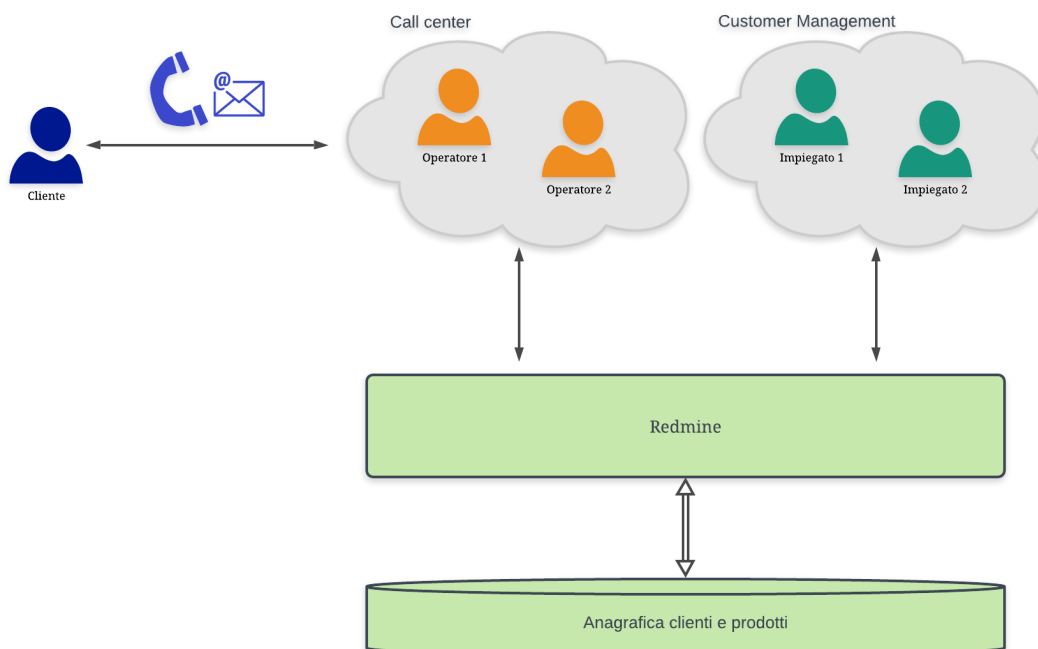


Figura 4.2: organizzazione del processo di Customer Management all'interno dell'azienda.

Questo sistema viene utilizzato, oltre che per il *customer management*, anche per altre attività all'interno dell'azienda. Esempi dei dati che vengono gestiti attraverso Redmine sono:

- le comunicazioni che provengono dai clienti attraverso tutti i canali a disposizione (telefono, e-mail, ecc.);
- le comunicazioni verso il cliente;
- le comunicazioni interne tra gli uffici;
- lo stato delle *pratiche di connessione*: con questo termine, in ambito utility, vengono definite tutte le attività che vanno a gestire direttamente i contatori dei clienti come ad esempio volture, nuove attivazioni e così via;
- lo stato delle pratiche dell'ufficio credito.

Il dato viene modellato attraverso il **Ticket**: questo concetto rappresenta ogni informazione modellata da Redmine che circola all'interno dell'azienda. Ad esempio, si utilizzano per tenere traccia delle segnalazioni dei clienti, ma anche per creare informative interne, e così via. Vista la grande mole di dati (che, tra l'altro, risulta ingestibile analizzare con Excel) e la sua importanza cruciale per l'azienda, si è deciso di partire dalla progettazione di un Data Mart interno che riesca a modellare questo concetto.

Spinto, dunque, da queste forti esigenze è partita questa fase del progetto di tesi con lo scopo ultimo di creare cruscotti che potessero aiutare gli operativi attraverso report giornalieri sullo stato di avanzamento dei ticket, ma anche per supportare chi ricopre ruoli decisionali in tutti gli uffici presenti all'interno dell'azienda. Verificare l'efficacia di una campagna di marketing, piuttosto che analizzare le performance di una determinata area o di un outsourcer sono solo alcuni esempi di ciò che può essere effettuato partendo dalle informazioni contenute all'interno dei ticket, come si può evincere dalla rappresentazione di esempio in figura 4.3.

PROVA		« Precedente   1/327   Successivo »	
Aggiunto da Redmine Admin 3 mesi fa. Aggiornato meno di un minuto fa.			
<b>Stato:</b>	Nuovo	<b>Inizio:</b>	10-08-2021
<b>Priorità:</b>	Normale		
<b>Assegnato a:</b>	-		
<b>Nome Cognome Chiamante:</b>		<b>CODICE POD :</b>	
<b>Codice Fiscale Chiamante:</b>		<b>CODICE PDR:</b>	
<b>Nr. Cliente:</b>		<b>Campagna:</b>	
<b>Tipo servizio:</b>	EE	<b>Tipo di info:</b>	Info generica precontratto (potenziale cliente)
<b>Link GreenPortal:</b>		<b>Canale:</b>	Telefono
<b>Nome Completo Cliente:</b>	PROVA	<b>id_fornitura:</b>	1
<b>Indirizzo di spedizione/mail:</b>		<b>id_cliente:</b>	1
<b>Recapito telefonico:</b>	XX000X1233456		
<b>Sottoattività</b>			<a href="#">Aggiungi</a>
<b>Segnalazioni correlate</b>			<a href="#">Aggiungi</a>

Figura 4.3: rappresentazione grafica delle informazioni contenute nel ticket di Redmine.

## 4.2 Obiettivi

L'obiettivo principale per questa seconda sezione di progetto di tesi è quello di creare un processo di Data Warehousing per la gestione dei ticket. Questo può essere ulteriormente suddiviso nei seguenti sotto-obiettivi.



- Effettuare un'analisi e riconciliazione delle sorgenti operazionali.
- Effettuare un'analisi del dominio, con particolare enfasi per i processi del *customer management*.
- Progettare il cubo di BI.
- Progettare il processo di **Extraction-Transformation-Loading (ETL)** che estragga le informazioni automaticamente dalle sorgenti operazionali.
- Utilizzare lo strumento Power BI per la creazione di report.

### 4.3 Pianificazione

Prima di cominciare con la fase di progettazione vera e propria, occorre effettuare una fase di pianificazione che porti alla realizzazione del risultato che ci si attende.

A livello organizzativo, la responsabile interna all'azienda di questo progetto di tesi è stata identificata nella Customer Service Manager. Per questo motivo, congiuntamente a tempo e risorse limitate, si è deciso di utilizzare un approccio bottom-up nella realizzazione del DW. Questo permette, infatti, di raggiungere risultati concreti in tempi brevi senza richiede investimenti elevati, mantenendo costantemente l'attenzione solo sulla parte dei processi aziendali in esame. Per questo motivo, durante tutte le fasi del progetto sono stati riscontrati feedback (almeno una volta al giorno ove possibile) con il personale interno in modo da aumentare il grado di soddisfazione finale. Analizzare solo una parte di dominio, inoltre, permette di concentrarsi con precisione su determinati processi aziendali, limitandone la complessità, il gruppo di persone con il quale interagire e, allo stesso tempo, fornire un risultato concreto che possa fungere da pubblicità anche con le altre aree interne all'azienda. Questa soluzione prevede di creare un Data Mart distinto per ogni gruppo di processi, andandoli ad assemblare iterativamente per creare un vero e proprio Data Warehouse. Questo potrebbe portare ad una visione parziale del dominio di interesse, problematica notevolmente mitigata dalla prima parte di tesi nel quale sono stati intervistati i responsabili della gran parte delle aree presenti all'interno dell'azienda.

Le possibili dimensioni che sono venute a galla immediatamente, durante le prime interviste effettuate con i dipendenti dell'azienda, sono legate al concetto di cliente (quindi

dati anagrafici, geografici e così via), alla tipologia di fornitura e prodotto che ha sottoscritto e all'area che interessa il ticket. Di queste possibili dimensioni che si potrebbero creare, solo durante la fase di analisi e riconciliazione delle sorgenti verranno evidenziate quali andranno poi ad essere effettivamente create. A livello metodologico, l'approccio utilizzato è stato di tipo **Demand-driven** poiché:

1. le sorgenti sono rappresentate da sistemi legacy di complessità tale da sconsigliarne la ricognizione e la normalizzazione;
2. i requisiti utente sono portati in primo piano;
3. non è fattibile un'analisi approfondita a priori delle sorgenti, in quanto i dati non sono memorizzati nei database interni all'azienda.

L'utilizzo di questo approccio prevede di partire dalle interviste effettuate agli utenti. Da queste viene effettuata la progettazione concettuale e successivamente quella logica; una volta analizzato anche il carico di lavoro, viene creato lo schema fisico del Data Mart. Parallelamente a questo flusso, viene effettuata la progettazione dell'alimentazione con l'analisi dei database sorgenti in modo tale da popolare correttamente tutto il Data Mart. Questo procedimento in generale è maggiormente difficile da perseguire rispetto ad un approccio guidato dai dati, in quanto potrebbe risultare complicata la progettazione dell'alimentazione del Data Warehouse. Inoltre, potrebbe non essere possibile creare fatti, misure e gerarchie in grado di rispettare in maniera precisa le specifiche dettate dagli utenti. Tuttavia, per gli stessi motivi descritti precedentemente (progetto di tesi con scope non eccessivamente rigido, ecc.), l'utilizzo di tale approccio risulta comunque attuabile e maggiormente adatti agli scopi preposti.

Come già descritto precedentemente, i costi non rappresentano un ostacolo in quanto l'intero elaborato viene compreso all'interno del progetto di tesi. Ciò rappresenta, dunque, un valore aggiunto; l'azienda in questo modo può usufruire di questa potenzialità per testare l'efficacia e i relativi vantaggi (e svantaggi, qualora ce ne siano) di un sistema di Data Warehousing. Infatti, quello che ci si aspetta da questo progetto di tesi è quello di evidenziare, seppur realizzando un Data Mart prototipale che copre solo parte dei dati aziendali, tutte il guadagno in termini di tempo, leggibilità e autenticità di un'infrastruttura tale, soprattutto rispetto a report estemporanei creati in Excel. La mancanza di un processo di Data Warehousing porta nella maggior parte dei casi ad analisi errate e

forvianti (ammesso che sia possibile estrarre e lavorare i dati). Tutto ciò viene amplificato, nel mondo di oggi, dalla crescita esponenziale delle informazioni e dalla necessità di monitorare ogni processo aziendale.

Esistono numerosi rischi quando si affronta un progetto di Data Warehousing, soprattutto legati alla mancanza di un approccio metodologico in modo tale da minimizzare le possibilità di insuccesso. Spesso, inoltre, viene attribuito a ciò il ruolo di panacea per tutti i problemi presenti all'interno dell'azienda, creando aspettative eccessive di difficile realizzazione. I principali fattori di rischio che portano a naufragare progetti di questo tipo sono legati a:

- gestione del progetto;
- tecnologie;
- dati e progettazione;
- organizzazione.

Parlando nello specifico di questo progetto, il rischio di insuccesso è notevolmente ridotto dal fatto che si tratta di un progetto di tesi e per questo le aspettative sono comunque ridimensionate. Questo implica anche una notevole duttilità nella gestione in termini di tempistiche e di scope, in quanto l'intero progetto è stato seguito solo da me con sporadiche interviste al personale interno dell'azienda. Inoltre, la possibilità di sfruttare le potenzialità di uno strumento di data visualization permette di essere nelle condizioni di soddisfare le aspettative di chi poi andrà ad utilizzare i report. Il problema legato alla qualità dei dati presenti nelle sorgenti operazionali resta un'incognita che potrà essere risolta solo dopo una profonda fase di analisi delle stesse. La possibilità, però, di avere un scope di progetto definibile in corso d'opera e non eccessivamente stringente permette comunque di considerare questo progetto fattibile e realizzabile nei tempi e nei modi che un progetto di tesi richiede.

Prima di progettare e realizzare il Data Mart, è importante definire con precisione l'infrastruttura che si vuole andare a creare per la realizzazione dell'intero Data Warehouse aziendale. La soluzione in figura 4.4 che si è definita deve prevedere i requisiti di separazione, scalabilità, estendibilità, sicurezza e amministrabilità, ovvero le caratteristiche che ogni DW ben costruito deve prevedere.

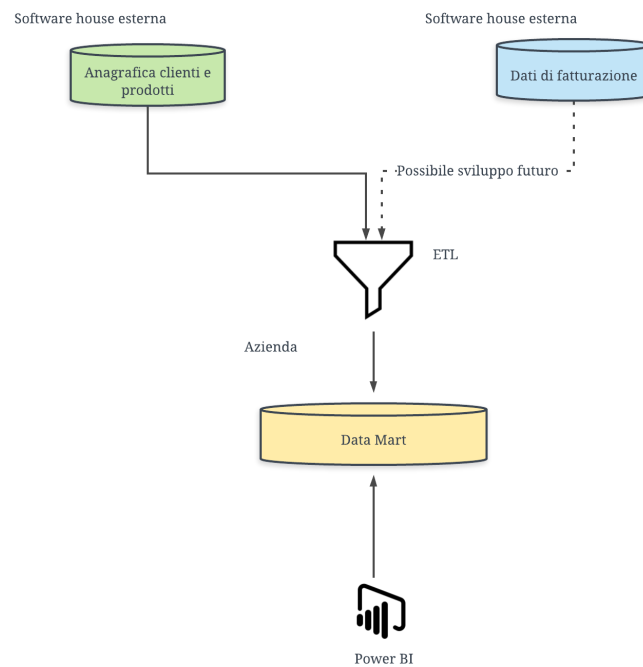


Figura 4.4: infrastruttura del Data Mart.

In particolare è stata creata un'architettura a 2 livelli: questa implementazione prevede la creazione di vari Data Mart all'interno del DW senza l'utilizzo di un ODS. Questa è dettata dal fatto che i database sorgenti sono costituiti da sistemi legacy e, seguendo anche le direttive della metodologia basata sui requisiti, la costruzione dell'ODS, non fondamentale in questa parte, potrebbe risultare molto complicata. Tale infrastruttura prevede, comunque, che siano realizzati degli strumenti ETL che permettano di estrarre i dati dalle sorgenti operazionali e di lavorarli in modo tale da essere direttamente utilizzati all'interno del DW. Partendo da questa base teorica, l'architettura che si è andata a creare è così formata:

- basi dati operazionali presso software house esterna;
- staging area e strumenti ETL realizzata nei db relazionali interni all'azienda;
- Data Mart realizzato nei db relazionali interni all'azienda;
- strumento di analisi dati per creazione di reportistica in near real-time.

Questa decisione è stata dettata dal fatto che è importante avere un accesso diretto all'interno dei db aziendali alle informazioni su cui viene effettuata la business intelligen-

ce. Questo per un duplice motivo: da un lato permette un accesso e un'operatività più efficiente da parte di chi ci deve lavorare, poiché in questo modo si evitano tutta una serie di problematiche "fisiche" che potrebbero nascere. Dall'altro lato, è importante avere i dati interessanti all'interno dell'azienda per diversi motivi: questi rappresentano infatti informazioni particolarmente importanti per il business e su cui è fondamentale evitare problemi di inconsistenza, sicurezza e valore per l'attività economica. Organizzando in tale modo l'architettura è possibile validare tutti i requisiti definiti precedentemente. In particolare, la separazione viene mantenuta in quanto l'elaborazione analitica viene fatta in sorgenti completamente distaccate da quella transazionale. In termini di scalabilità è possibile aumentare le dimensioni dell'architettura in maniera efficiente in previsione della crescita dei volumi di dati da gestire, in quanto si lavora con database locali e quindi di più facile gestione. Per quanto concerne, invece, l'estendibilità, elemento molto delicato all'interno dell'azienda proprio perché si sta attuando una migrazione verso altri sistemi, si è cercato di progettare il sistema in modo che sia semplice e non troppo costoso trasferire le informazioni. Avendo creato il tutto su database interni, un'eventuale modifica delle sorgenti operazionali comporterebbe "solo" la creazione di un nuovo ETL. La sicurezza delle informazioni, inoltre, è un requisito soddisfatto automaticamente in quanto nell'azienda è già attivo un processo che garantisca che le informazioni al suo interno siano salvate in maniera congrua con i requisiti di sicurezza richiesti. Infine, l'architettura è stata creata quanto più semplice possibile, ed è stata tutta opportunamente documentata, in modo da soddisfare il requisito di amministrabilità. In conclusione, in questa fase non è necessario effettuare un'ulteriore classificazione in termini di consistenza dell'architettura tra i Data Mart del DW in quanto questo progetto di tesi prevede la creazione di un singolo data silos, o di una parte di esso.

## 4.4 Progettazione e prototipazione Data Mart

Passando alla parte di progettazione e implementazione dei singoli data mart, questa prevede di creare iterativamente ognuno di essi andando a definire in maniera incrementale l'intero sistema. Per gli scopi di questo progetto, dunque, è sufficiente eseguire una singola iterazione delle parti che verranno descritte in seguito, realizzando un data mart completo e corretto.

### 4.4.1 Analisi dei requisiti

La prima sezione di questo processo prevede l'analisi dei requisiti. In questa parte, dunque, diversi esperti del dominio si sono messi gentilmente a disposizione per essere intervistati in ogni modo. I motivi per i quali sono state portate avanti interviste a diverse figure sono molteplici:

- necessità di una visione da più dipendenti dell'azienda con diversi ruoli e/o visioni;
- importanza cruciale della fase di analisi dei requisiti;
- esigenza di non ostacolare continuamente il lavoro quotidiano di determinate figure.

Le interviste, non sempre strutturate, sono state organizzate in maniera tale da poter conoscere con un buon livello di dettaglio il dominio applicativo. Infatti, questa parte del progetto risulta cruciale: da un lato il progetto potrebbe naufragare a causa di eventuali problematiche nel reperire le figure e di conseguenza le informazioni necessarie, dall'altro è possibile che il risultato non sia rispecchi ciò che gli utilizzatori si aspettano. Fortunatamente, nei momenti nei quali nessuna figura è disponibile, mi sono stati messi a disposizione diversi documenti in modo che io potessi leggere e analizzare il dominio applicativo autonomamente. Al termine delle prime interviste, l'analisi di dominio ha sottolineato la necessità di creare, in base alla diversa granularità delle informazioni, due cubi di BI differenti:

- cubo generale con il livello di dettaglio legato alle informazioni del ticket;
- cubo fine con il livello di dettaglio corrispondente al singolo stato del ticket.

Successivamente si è andato a snocciolare i vari concetti emersi nelle rispettive informazioni e, in particolare, quelle legate al cubo generale su cui è necessario fare analisi sono riportati di seguito.

- Ticket
  1. Data apertura
  2. Data chiusura
  3. Durata
  4. ID cliente

## 5. Numero di modifiche

## • Progetto

1. Nome progetto
2. Area di competenza

## • Categoria

1. Nome categoria
2. Durata attesa
3. Tolleranza

## • Dettaglio categoria

1. Nome dettaglio categoria

## • Dettaglio stato

1. Nome dettaglio stato
2. Nome stato

Partendo dal concetto di **Progetto**, ovvero la classificazione ad alto livello attraverso la quale vengono categorizzati i ticket, è stata effettuata un'analisi con un maggiore livello di dettaglio e con domande più mirate. In particolare, per ciascun progetto sono state considerate quale **Categoria**, ovvero un raggruppamento più fine dell'informazione, è da considerare, quali possono essere eventuali ticket da pulire e normalizzare, e così via. Dopo questa fase sono emerse delle considerazioni. Innanzitutto occorre sottolineare che i progetti sono da considerare come molto eterogenei tra di loro, e quindi un' unica analisi non può valere per tutti. Alcuni progetti non sono stati creati per vari motivi: alcuni, ad esempio, sono ancora in via di sviluppo e non organizzati a livello di processo non è utilizzato in maniera sufficientemente corretta da poter essere utilizzati per effettuare delle analisi.

Oltre al concetto di progetto, ad ogni ticket viene associato anche uno **Stato**: questo si presenta come un insieme molto numeroso di posizioni in cui il dato si può trovare. Gli stati si suddividono in:

- stati di apertura o di transito: comprendono le posizioni entro le quali il ticket può essere ancora lavorato. Di questo insieme fanno parte anche tutti quei stati iniziali che rappresentano il primo passaggio.
- stati di chiusura: di tale raggruppamento fanno parte quelli che non possono più essere lavorati e che terminano il flusso del ticket.

Tra di queste ne esistono una grande varietà per ciascuna tipologia, e corrispondo al concetto di **Dettaglio Stato**. Importante notare che quest'ultimi sono diversi per ciascuna categoria quindi spesso non è possibile confrontare lo stato dei ticket tra i diversi concetti presenti nei progetti. Per riassumere, la fase di analisi dei requisiti ha fatto emergere una serie di necessità o esigenze, riassunte di seguito.

- Necessità di gestione di ticket entro determinate scadenze in base alla categoria.
- Comprendere dei pattern del customer journey.
- Capire l'andamento dei ticket in base ai progetti, alle categorie, ecc.
- Effettuare un'analisi in base ai dati dei clienti .
- Effettuare un'analisi sulle prestazioni di determinati processi aziendali.
- Analizzare il comportamento di determinati outsourcer.
- Creare delle statistiche sulla gestione dei ticket.
- Possibilità di estrapolare dei report in formato tabellare per evidenziare eventuali insiemi di informazioni da consegnare agli operativi.
- Necessità di filtrare dinamicamente le informazioni contenute nei report.
- Fornire una reportistica di tipo operativo.
- Fornire una reportistica di tipo strategico-decisionale.
- Necessità di aggiornare in near real-time (una volta al giorno) le statistiche sui report.
- Necessità di non appesantire con estrazioni OLAP le sorgenti dati, già sovraccaricate dalle query OLTP.



- Normalizzare la sorgente dati.
- Fare analisi sullo storico dei ticket, considerando cioè tutti i passaggi di stato e le modifiche effettuate su di essi.

#### 4.4.2 Analisi e riconciliazione delle sorgenti

Per motivi di tempistiche e di scope, si è deciso di concentrarsi esclusivamente su di una sorgente dati operativa, collocata fisicamente all'interno del database di un outsourcer. La sorgente operativa dalla quale è partita il processo è costituita da un database relazionale in linguaggio MySQL. Questo è stato creato partendo da Redmine, il quale permette di installare all'interno della propria azienda un web server già configurato, che comprende anche lo schema del database. Il lavoro in questa fase si è concentrato soprattutto nell'analisi e comprensione degli schemi della sorgente dati, partendo dalla documentazione fornita online in figura 4.5, senza gli obiettivi di creare uno schema riconciliato ma solo con la necessità di mostrare quali informazioni sono contenute al suo interno.

Lo scopo di questa parte iniziale è di capire quali, quante e in che modo sono disponibili le informazioni all'interno del database sorgente. Fin da subito si è notato che si tratta di un database sorgente legacy e non organizzato in maniera efficiente e corretta. Le tabelle non risultano essere normalizzate in alcun modo e talvolta è complicato anche comprendere il contenuto delle stesse. In un primo momento quindi quello che è stato fatto comprendeva una visualizzazione delle informazioni contenute nelle tabelle. Questo procedimento, seppur lungo e tedioso, è stato fondamentale per capire l'organizzazione dei concetti all'interno del database. In questo modo si è notato che non tutte le informazioni che potevano essere utili agli scopi di questo progetto erano disponibili: ad esempio non è presente l'anagrafica dei prodotti, oppure, ben più grave, non sono presenti i dati di profilazione del cliente. Giunto alla conclusione che non tutti i dati sarebbero stati presenti all'interno di questo database, si è cercato di capire come sono organizzate le informazioni esistenti. Questo ha sollevato non poche problematiche per i motivi già descritti in precedenza di organizzazione delle tabelle. Per fare solo un esempio, l'informazione dell'ID univoco del ticket è stata duplicata in molteplici tabelle, da cui consegue un notevole sforzo per estrarre dati corretti e consistenti. Inoltre, le colonne all'interno delle tabelle non hanno nomi sufficientemente esplicativi, aumentando maggiormente la complessità di analisi. Tutto ciò, per i tempi e i modi dettati dal fatto che si tratta di un progetto

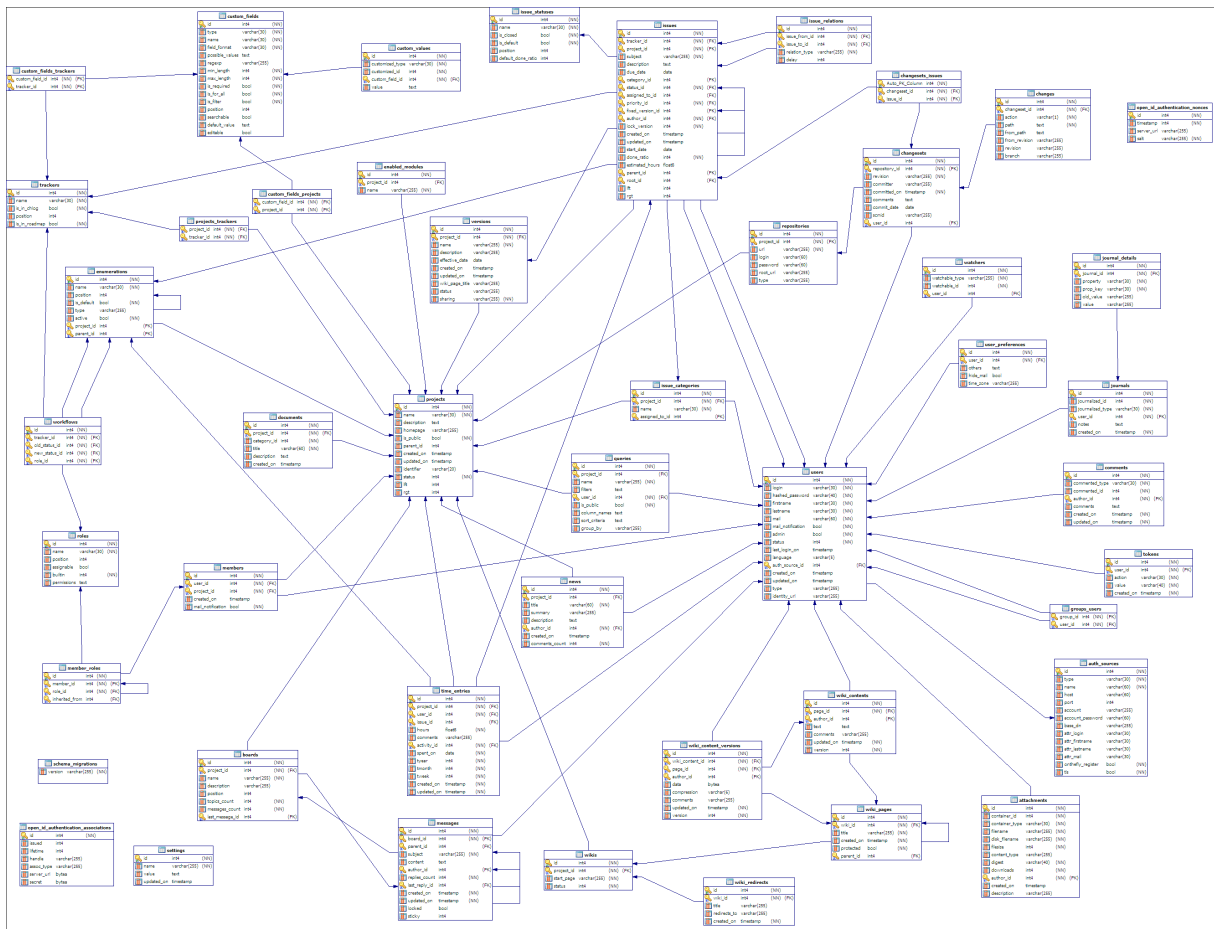


Figura 4.5: database model del software Redmine.

di tesi, ha portato alla conferma dell'impossibilità di creare uno schema riconciliato da cui ricavare un ODS. Per quanto rappresenti un approccio metodologico molto efficace, la costruzione dell'ODS stesso avrebbe occupato la maggior parte del tempo e dell'effort del progetto di tesi, senza poter progettare e sviluppare il Data Mart.

### 4.4.3 Progettazione concettuale

#### Il Dimensional Fact Model

Il primo passo di progettazione del Data Mart consiste nella creazione dello schema concettuale. Questo viene creato considerando sia i requisiti utenti, sia le informazioni emerse dall'analisi delle sorgenti. Il formalismo utilizzato è quello del **Dimensional Fact Model (DFM)**, rappresentazione grafica definita in [7] come una *sestupla*:

$$f = (M, A, N, R, O, S)$$

dove:

- $f$  rappresenta il **fatto**, ovvero il concetto di interesse per il processo decisionale;
- $M$  è un set di **misure**, ovvero proprietà numeriche di un fatto che ne descrivono un aspetto quantitativo di interesse;
- $A$  è un set di **attributi dimensionali**, cioè proprietà con un dominio finito che descrivono le coordinate in analisi;
- $N$  è un set di **attributi descrittivi**, che si differenziano dai precedenti per il fatto che contiene valori continui di cui non è necessario effettuare aggregazioni;
- $R$  rappresenta un insieme di coppie ordinate di elementi di  $A$  con collegamenti di tipo *molti-a-uno*;
- $O$  rappresenta l'**arco opzionale**, ovvero collegamenti tra due dimensioni il cui dominio può comprendere il valore *null*;
- $S$  rappresenta l'**aggregazione**, ovvero un insieme composto da misure, dimensioni e operazioni di aggregazione.

Attraverso questa rappresentazione, dunque, è possibile visualizzare immediatamente la progettazione concettuale, fornendo anche un'ottima base sulla quale formulare le interrogazioni dell'utente e favorire il dialogo con lo stesso allo scopo di raffinare i requisiti. Inoltre, questa rappresentazione funge anche da aiuto in fase di progettazione logica ed è ottima come documentazione finale da fornire a posteriori. Per definire la rappresentazione dello schema concettuale, in quanto si è seguiti un approccio demand-driven, in base alle interviste realizzate viene scelto il fatto, con le relative misure e dimensioni. Ci si occuperà solo in un secondo momento di come avverrà il collegamento tra questa rappresentazione e le sorgenti operazioni.

### Il progetto concettuale sul ticket

In figura 4.6 viene rappresentato lo schema concettuale realizzato per questo specifico problema.

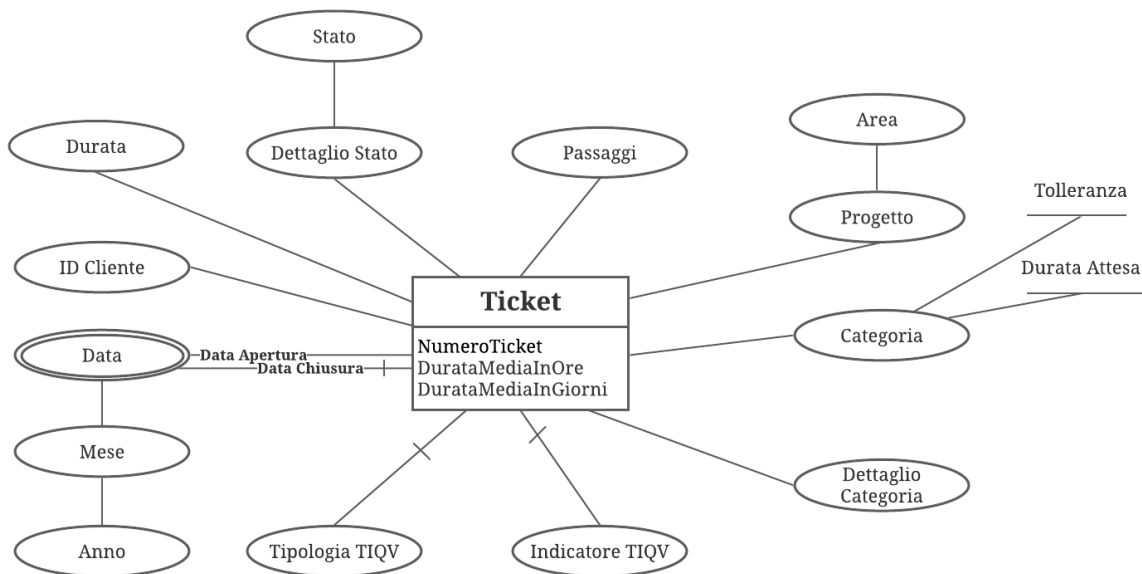


Figura 4.6: schema concettuale utilizzando la rappresentazione DFM.

Occorre sottolineare che il DFM mette a disposizione delle rappresentazioni grafiche per mostrare immediatamente tutti i concetti necessari. Con riferimento alla figura 4.6, questo formalismo definisce il fatto come un rettangolo posto al centro da qui partono tutte le gerarchie delle dimensioni. Quest'ultime vengono rappresentate con un ovale, salvo casi in cui vengono utilizzati costrutti avanzati. Tra di essi, in figura 4.6, viene utilizzato

il formalismo del doppio ovale con la dimensione *Data* per rappresentare il concetto di **gerarchia condivisa**: questa evenienza consiste di fatto in un'abbreviazione utile per sottolineare che una porzione di gerarchia è replicata più volte nello schema. Un altro costrutto avanzato che viene utilizzato è quello dell'arco opzionale, descritto precedentemente. Ad esempio nel caso della *Data Chiusura*, attraverso quel formalismo grafico, viene evidenziato il concetto che non tutte le istanze del fatto presentano un valore per questa dimensione. Infine, troviamo *Tolleranza* e *Durata Attesa* che ci modellano il concetto di attributo descrittivo. Per concludere questa parte di progettazione concettuale, si è validata la rappresentazione creata in base all'analisi effettuata. Seppur non rappresenta una complessità elevata, attraverso questa soluzione è possibile rispondere a tutte le richieste effettuate dagli utenti e perciò il risultato si può considerare come corretto.

Va sottolineato che il DFM in questione è stato creato tenendo in considerazione l'analisi effettuata del database sorgente che rappresenta notevoli limiti in termini di informazioni che mette a disposizione. Per fare un esempio, legate al cliente esistono ben poche informazioni utili e che siano organizzate in maniera strutturata.

#### 4.4.4 Carico di lavoro e volume dati

Il carico di lavoro preliminarmente espresso dagli utenti viene verificato sullo schema concettuale. Per effettuare questo passaggio ci si può avvalere di mezzi grafici, come ad esempio in figura 4.7.

In particolare, la rappresentazione in figura 4.7 corrisponde all'interrogazione "*Numero di ticket nel mese di marzo aggregato per Progetto e Categoria*". In questo modo, in maniera del tutto intuitiva ed immediata, è stato possibile verificare che tutte le richieste possano essere realizzabili e, dunque, ha permesso di stimare il carico di lavoro. In questo ambito non si ha lo scopo di ottimizzare le prestazioni del sistema in quanto le richieste degli utenti potrebbero cambiare nel tempo e, dopo la messa in produzione del sistema, occorre effettuare un'ulteriore fase di *tuning*. Ciò vale soprattutto nella realtà entro la quale è stato realizzato questo sistema, in quanto gli utenti si trovano a dover formalizzare le richieste per la prima volta; ciò è stato evidenziato soprattutto attraverso le interviste.

La progettazione del Data Mart prevede anche una fase entro la quale vengono profilati gli utenti per determinare le modalità di accesso ai dati definendo chi può accedere

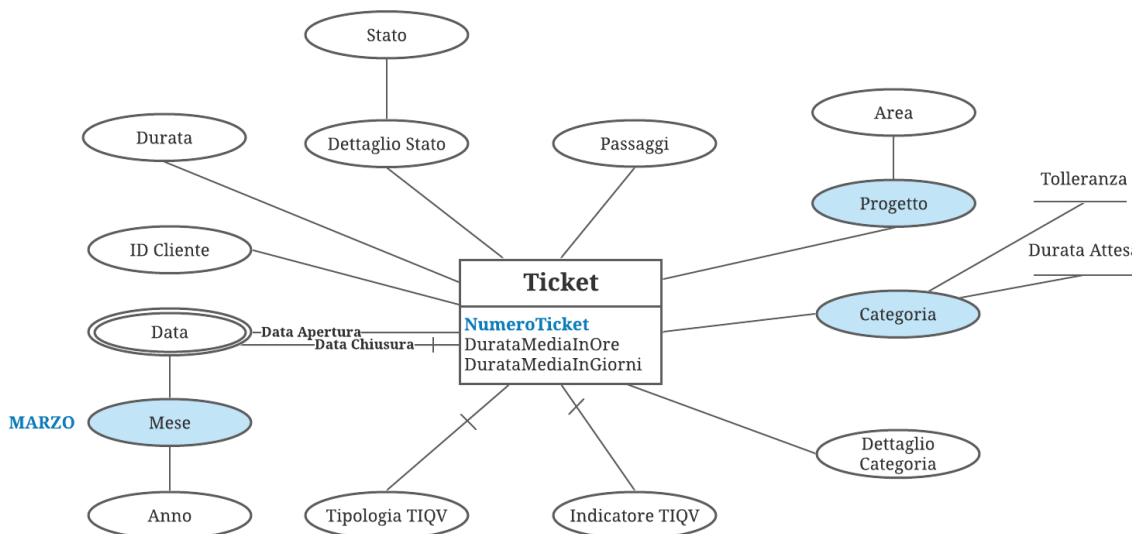


Figura 4.7: carico di lavoro rappresentato graficamente con il DFM.

a quali dati e con quale modalità. Per fare ciò è necessario, dunque, classificare gli utenti in base ai ruoli che ricoprono all'interno dell'azienda, come in tabella 4.1.

Figura	Ruolo	Permessi
CEO	Direzione	Accesso a tutti i dati
Customer Service Manager	Manager	Accesso a tutti i dati
Responsabile pre-vendita	Pre-vendita	Progetto: <i>Back-office</i>
Responsabile reclami	Reclami	Progetto: <i>Reclami</i>
Responsabile post-vendita	Post-vendita	Progetto: <i>Logistica</i>
Operativo post-vendita gas	Post-vendita gas	Categoria: <i>Voltura Gas</i>
Operativo post-vendita luce	Post-vendita luce	Categoria: <i>Voltura Luce</i>

Tabella 4.1: definizione ruoli.

Come evidenziato in tabella 4.1, le figure che appartengono ad un determinato ruolo avranno a disposizione le informazioni che appartengono solo al loro ufficio di competenza, cioè in base alla funzione aziendale svolta. Infine, va riportato il volume dati. In quanto si tratta di un progetto di tesi e lo scopo è di creare un prototipo, questa parte non è stata valutata. A livello teorico, infatti, occorrerebbe determinare la dimensione di tutto

il Data Mart partendo dal numero di valori distinti degli attributi nelle gerarchie e dal numero di eventi del fatto.

#### 4.4.5 Progettazione logica

Mentre la modellazione concettuale è indipendente dal modello logico prescelto, per questa fase occorre definire quale modello deve essere utilizzato. Esistono diverse implementazioni, come definito in [14]:

- **Relational On-Line Analytical Processing (ROLAP)**: utilizza il modello relazione per la rappresentazione dei dati multidimensionali dei Data Mart. I vantaggi di tale implementazione sono soprattutto legati all'esperienza dei progettisti in ambito relazione e alla semplicità di gestione degli stessi, a discapito soprattutto delle performance;
- **Multidimensional On-Line Analytical Processing (MOLAP)**: memorizza i dati utilizzando strutture intrinsecamente multidimensionali come ad esempio vettori multidimensionali. Questi hanno il vantaggio di rappresentare la soluzione naturale e dunque hanno ottime prestazioni, soprattutto nei confronti di ROLAP, in quanto si evita la creazione di query complesse. Tra gli svantaggi va riportata la mancanza di strutture dati standard di questo tipo e la possibile problematica della **sparsità**, ovvero la presenza di molte celle vuote all'interno della rappresentazione;
- **Hybrid-OLAP (HOLAP)**: combinazione dei due precedenti, sfrutta le potenzialità di entrambi i modelli precedentemente elencati, tuttavia non è ancora utilizzato nel panorama attuale.

La scelta in questo progetto di tesi è stata quella di utilizzare il sistema ROLAP in quanto all'interno dell'azienda sono presenti dei database relazionali sui quali possibile implementare il cubo, senza dover richiedere l'utilizzo di nuove piattaforme di tipo MOLAP o HOLAP. La modellazione multidimensionale su sistemi relazionali è basata sul cosiddetto **schema a stella**. Questa soluzione prevede la creazione di una tabella definita **Dimension Table (DT)** per ogni dimensione, composta da una chiave primaria e da un insieme di attributi che descrivono le dimensioni a diversi livelli di aggregazione. Oltre alla DT, viene prevista anche la **Fact Table (FT)**, la quale importa tutte le chiavi

delle DT (che fungono da chiave primaria) e contiene al suo interno un attributo per ogni misura. Questa soluzione non soffre della problematica della sparsità in quanto vengono memorizzate solo tuple dello spazio multi-dimensionale per cui esistono eventi. Un ulteriore elemento che deve essere considerato prima di effettuare la progettazione logica è rappresentato dagli **scenari temporali** che devono essere rispettati. Il fatto all'interno del Data Mart è considerato come elemento *dinamico* che, dunque, varia e si modifica nel tempo. In questo caso il ticket cambia ogni qualvolta un operatore deve modificare un campo, lo stato, e così via. Oltre al fatto, possono cambiare anche le dimensioni nel tempo. Queste, in particolare, sarebbe utopistico considerarle come *statiche*, soprattutto nelle realtà aziendali. Per far fronte a quest'evenienza, occorre definire lo scenario temporale che si vuole rispettare. In particolare, attraverso lo schema a stella possono essere implementate due soluzioni:

- **attualizzazione** nella quale i dati vengono sempre interpretati in base all'attuale configurazione della gerarchia;
- **verità storica** che prevede l'interpretazione del dato in base alla configurazione valida al momento dell'inserimento.

Già in fase di analisi dei requisiti, era stata considerata questa possibilità e la scelta è ricaduta sulla prima soluzione, in quanto le modifiche che dovrebbero essere eseguite alle dimensioni corrisponderanno soprattutto ad aggiustamenti dei valori e non occorre mantenere la verità storica. In fase di progettazione logica, dunque, per implementare questa soluzione non occorre effettuare modifiche sostanziali ma ci si limiterà ad attribuire a tutti i dati, anche quelli passati, il valore attuale, andando a sovrascrivere il vecchio con il nuovo ogni volta che si verifica un cambiamento.

Il primo passo per poter realizzare lo schema logico, prevede la traduzione dello schema concettuale. In questa fase, si parte dal DFM realizzato e si va a creare una prima bozza di schema logico come in figura 4.8. Per semplicità di rappresentazione dall'immagine 4.8 vengono omesse tutte le DT che comprendono solo la chiave e la dimensione, che son state comunque considerate in questa fase.

Questo, dunque, si presenta come notevolmente non ottimizzato, quindi occorre effettuare una fase di tuning successiva. In particolare vanno gestiti tutti i costrutti avanzati e si deve cercare di limitare ridondanze, inefficienze, e così via. Si può subito notare



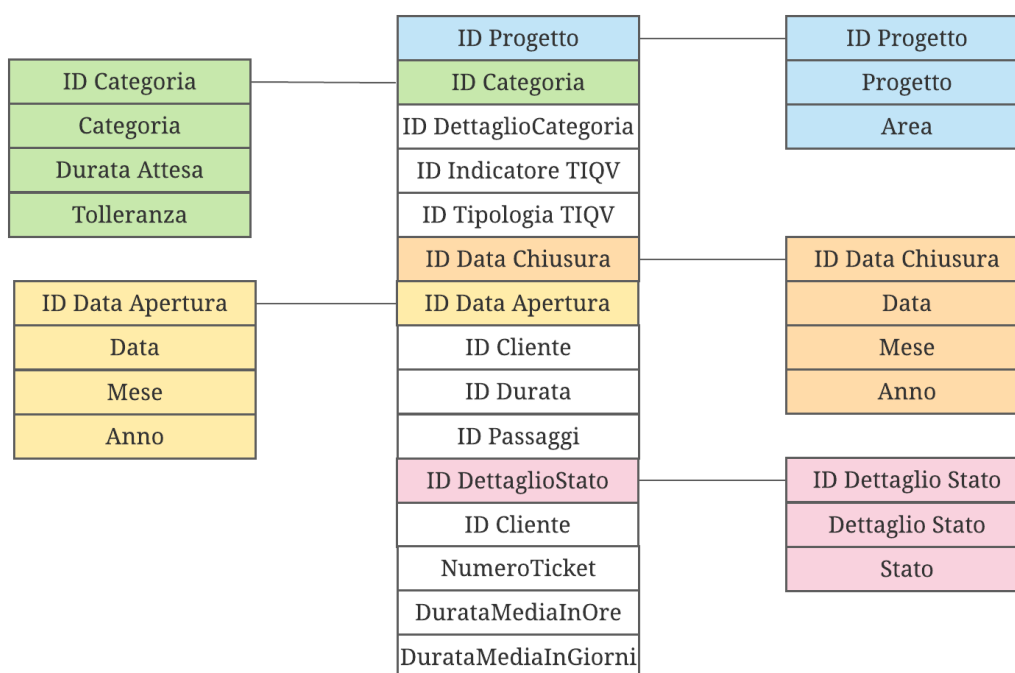


Figura 4.8: schema logico non ottimizzato.

dall'immagine in figura 4.8 dalla gestione fortemente ridondante delle informazioni legate alla gerarchia condivisa *Data*. In particolare le due strutture contengono precisamente le stesse informazioni quindi possono andare ad essere accorpate importando due volte la chiave primaria nella FT, come descritto in figura 4.9.

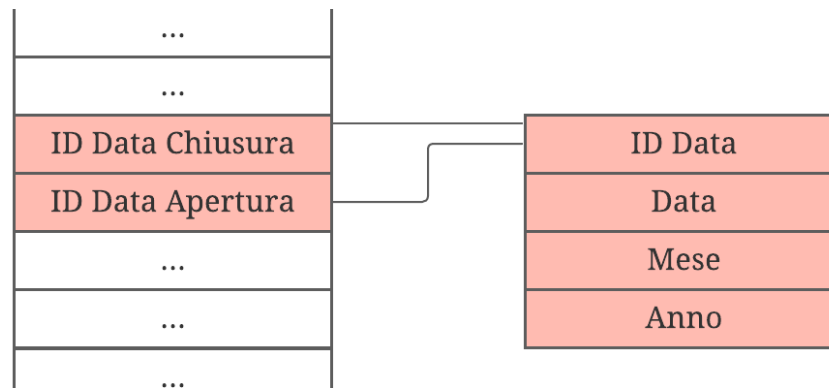


Figura 4.9: tuning della gerarchia condivisa *Data*.

Per quanto riguarda gli attributi descrittivi, in quanto non collegati direttamente al fatto non occorre una gestione particolare. D'altro canto, occorre tenere particolare riguardo nei confronti degli archi opzionali. Se questi corrispondono solo a delle porzioni di gerarchie, come ad esempio nel caso di *Indicatore TIQV*, ci si limita all'inserimento di un valore *null*. Diversamente va gestita la casistica della dimensione *Data Chiusura*, in quanto ad essere opzionale è l'intera gerarchia. In questo caso occorre inserire il valore *null* per tutta la porzione di gerarchia, che corrisponde ad un'intera tupla fittizia nella DT. L'ultima casistica prevede la gestione delle dimensioni **degeneri**: quest'eventualità si verifica quanto la gerarchia prevede, oltre alla chiave, solo un elemento. In questo caso si ragiona in due modi: se la lunghezza dell'attributo non è eccessiva, esso può essere direttamente inserito in FT, altrimenti è possibile modellare una **Junk Dimension** che andrà a comprendere tutte le dimensioni che presentano tale caratteristica.

Per valutare correttamente la soluzione migliore in termini di occupazione di memoria occorre definire, seppur in maniera *eurisitica*, quanto occupano le soluzioni che prevedono l'utilizzo della **Junk Table (JT)** o delle DT per le dimensioni degeneri. Simulando la situazione in cui la cardinalità del fatto è 500k e le chiavi abbiano la stessa occupazione di memoria di un INT, in tabella 4.2 vengono riportati i valori presi in considerazione, a cui fanno seguito i calcoli.

Dimensione degenera	Tipo	Occupazione	Cardinalità
Passaggi	INT	5 byte	100
ID Cliente	INT	5 byte	300k
Durata	INT	5 byte	1000
Dettaglio Categoria	STRING	20 byte	30
Indicatore TIQV	STRING	20 byte	20
Tipologia TIQV	STRING	20 byte	20

Tabella 4.2: definizione occupazione di memoria dimensioni degeneri.

## 1. Soluzione senza Junk Table

$$FT = 500000 * (5 + 5 + 5 + 5 + 5 + 5)byte = 15MB$$

$$DT_C = 30 * (5 + 20) = 750byte$$

$$DT_T = 20 * (5 + 20) = 500byte$$

$$DT_I = 20 * (5 + 20) = 500byte$$

## 2. Soluzione con Junk Table

$$FT = 500000 * (5 + 5 + 5 + 5)byte = 10MB$$

$$JT = 30 * 20 * 20(5 + 20 + 20 + 20) = 0.78MB$$

Come si può notare, dunque, la soluzione migliore è quella che presenta la JT come raggruppamento delle dimensioni degeneri. A questo punto può essere definita la soluzione di progettazione logica come in figura 4.10. Occorre sottolineare che non sono state considerate ulteriori ottimizzazioni, come la creazione di viste, in quanto per gli scopi di prototipazione di questo progetto non risulta essere fondamentale.

#### 4.4.6 Progettazione dell'ETL

Come descritto precedentemente, i database con al loro interno le informazioni necessarie per la seconda parte di tesi sono gestiti da delle software house esterne. Questo crea, dunque, un problema di accesso ai dati che, soprattutto in realtà aziendali, potrebbe risultare

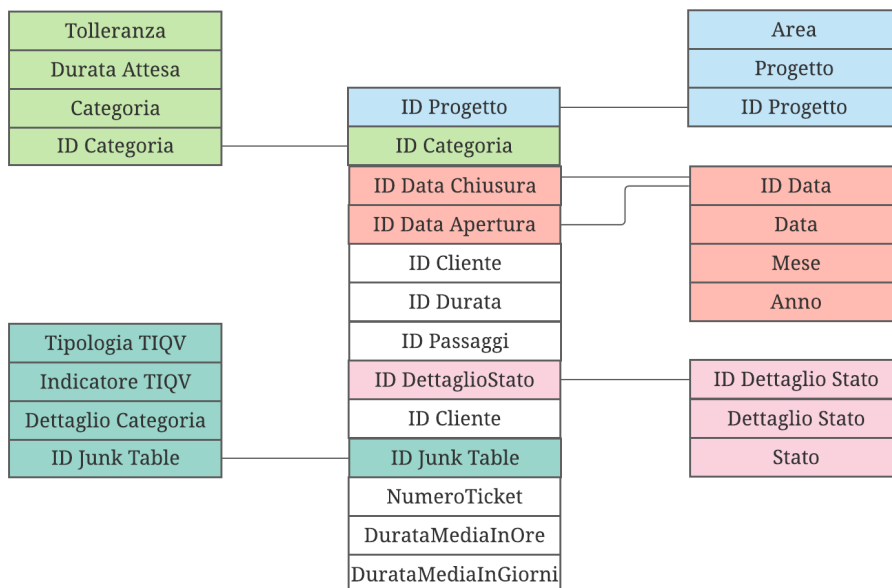


Figura 4.10: risultato di progettazione logica.

molto complicato. La soluzione che l'ufficio IT aziendale ha deciso di utilizzare passa attraverso la creazione di una **Virtual Private Network (VPN)**. Come descritto in [15], questa tecnica prevede di creare una connessione tra due parti sicura e controllata all'interno di un *common communications medium* tipicamente inaffidabile come ad esempio *Internet*. In figura 4.11 si può notare quale come è stata utilizzata questa soluzione.

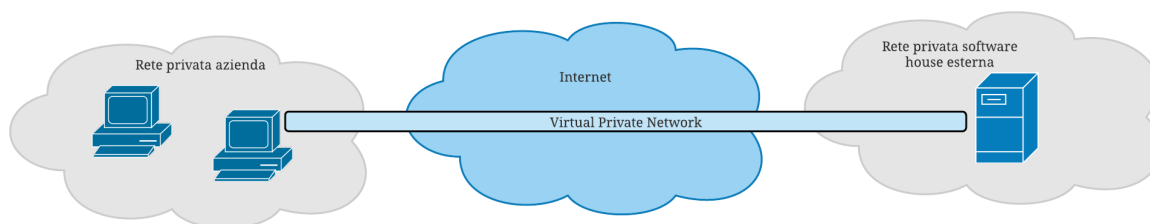


Figura 4.11: rappresentazione semplificata della soluzione con VPN.

I benefici che derivano da questa soluzione, in generale, sono molteplici; seppur potrebbe non avere le stesse prestazioni di una rete privata, la VPN rappresenta un'ottima soluzione quando una parte dei sistemi non è raggiungibile da reti locali. Essa garantisce i requisiti di sicurezza attraverso la creazione di un tunnel crittografato all'interno della rete non sicura. Per creare questi canali, vengono utilizzati in informatica e nelle

telecomunicazioni specifici protocolli chiamati **Transport Layer Security (TLS)** e il suo predecessore **Secure Sockets Layer (SSL)**. TLS[16] viene definito come un protocollo *client-server* di tipo *connection oriented*. Al giorno d'oggi rappresenta il protocollo standard *de facto* per le comunicazioni attraverso Internet in quanto garantisce autenticazione, integrità e confidenzialità tra le due parti. L'utilizzo di tali protocolli, dunque, risulta essere fondamentale per la corretta creazione di canali di comunicazione sicuri. Esula dagli scopi di questo progetto di tesi analizzare la robustezza e l'affidabilità degli algoritmi di crittografia.

In merito alla progettazione dell'ETL, viene definita in maniera più specifica la sua funzionalità e come può essere effettuato. Innanzitutto, l'ETL prevede l'esecuzione in serie di diverse operazioni: la prima (*extraction*) si occupa di estrarre i dati dai database sorgenti; successivamente vi è la fase di *transformation* che può essere ulteriormente scomposta in pulizia (elimina gli errori) e trasformazione (rende conforme i dati delle sorgenti con quelli nel Data Mart). Infine, troviamo la fase di caricamento (*loading*) che si occupa di inserire le informazioni all'interno del Data Mart. Per effettuare queste operazioni, viene creato un apposito database, definito **Staging Area**, all'interno del quale i dati sostano momentaneamente per essere lavorati. Andando nello specifico delle relative fasi, la fase di estrazione può essere definita come:

- **statica**: ovvero ad ogni esecuzione dell'ETL lo stato viene ricreato *ex-novo*;
- **incrementale**: ovvero vengono aggiunte solo le modifiche effettuate sul database sorgente dall'ultima esecuzione dell'ETL. Questa può essere ulteriormente scomposta in:
  - *immediata*, ovvero ogni modifica nel database sorgente viene subita riversata nel Data Mart;
  - *ritardata*, cioè in un determinato momento viene eseguito l'ETL che andrà a considerare tutte le informazioni che si sono modificate.

A causa dei notevoli limiti di performance del database sorgente dal quale vengono estratte le informazioni, si è deciso di utilizzare un'estrazione di tipo incrementale. Per gli stessi motivi, considerando anche le relativamente basse esigenze di effettuare analisi in real-time, si è deciso di creare l'ETL con azione ritardata.

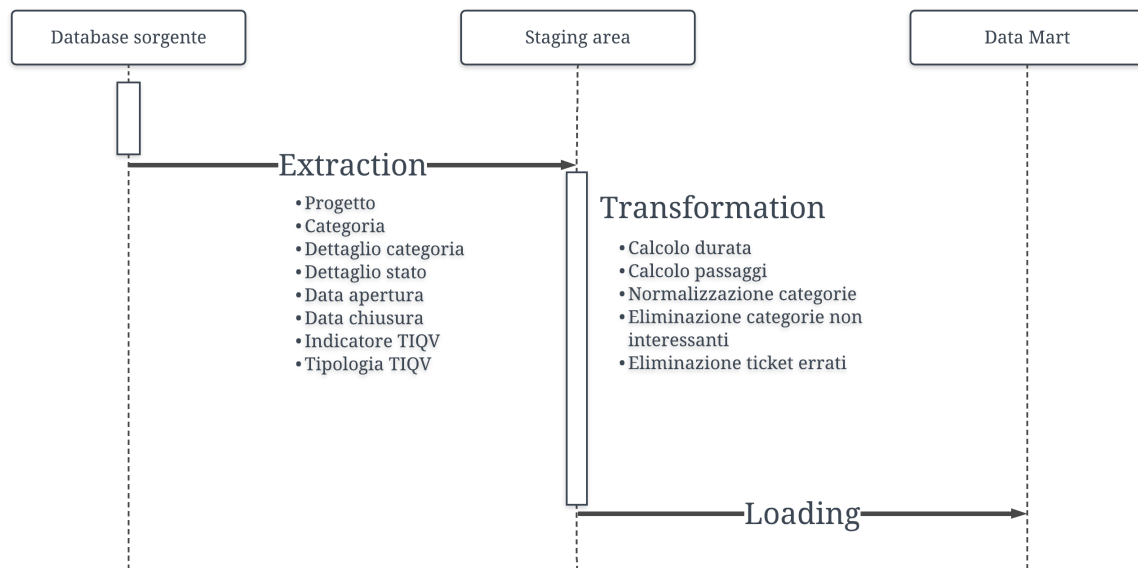


Figura 4.12: workflow ETL.

In questo modo, progettando la partenza dello stesso durante la notte sarà possibile non appesantire in maniera eccessiva le informazioni e allo stesso tempo avere il dato con le caratteristiche di freschezza richieste. Per effettuare quest'implementazione, oltre che alle specifiche tecniche del sistema che esulano dagli scopi di questo progetto di tesi, occorre sottolineare che il database aveva a disposizione la *timestamp* sull'ultima modifica effettuata su quel ticket. In questo modo è stato possibile implementare tale situazione e far sì che non sia necessario ricaricare *ex-novo* tutte le informazioni. Le tecniche di pulizia si sono concentrate, invece, nel fornire il dato visualizzato nella maniera che era stata richiesta, mentre le trasformazioni sono state eseguite soprattutto per far sì che sia rispettato lo schema logico prodotto, in quanto il database sorgente si è dimostrato da subito in maniera molto denormalizzata. In figura 4.12 si può notare i passaggi che sono stati seguiti per la creazione del workflow di ETL. La parte di *transformation* è stata creata all'interno dello stesso database contenente il Data Mart in quanto, per questi scopi, non era possibile utilizzare altre modalità. Al termine della fase di *loading*, le informazioni contenute all'interno delle tabelle di staging verranno eliminate in modo tale da non occupare spazio inutilmente con informazioni non necessarie. Come descritto precedentemente, per costruire l'ETL si è utilizzato il linguaggio nativamente inserito dentro MySQL. Le trasformazioni, dunque, corrispondono a delle *stored procedure* che vanno ad effettuare delle query di tipo **Data Manipulation Language (DML)** sui

dati, come descritto in 4.1. In particolare, la funzione in 4.1 rappresenta la trasformazione che calcola la durata del ticket, intesa come la differenza tra data di chiusura e data di apertura in ore.

```

1 CREATE DEFINER=`root`@`localhost` PROCEDURE `InserisciDurataTicket`()
2 BEGIN
3 UPDATE ft_ticket
4 SET ft_ticket.Durata = if(isnull(ft_ticket.DataChiusura),null,
5     timestampdiff(hour, ft_ticket.DataApertura,ft_ticket.DataChiusura))
6 WHERE ft_ticket.Durata is null;
7 END;

```

Source Code 4.1: esempio di stored procedure realizzata.

I limiti di questo approccio son legati soprattutto a performance ed espressività del linguaggio. Infatti, per poter eseguire un'operazione semplice come quella definita in 4.1 occorrono svariati minuti. A ciò va aggiunta anche una limitata espressività dei costrutti che non permettono di avere le stesse potenzialità di *processing* dei dati rispetto a linguaggi come Python oppure *R*.

Per concatenare più *stored procedure*, è stato utilizzato il costrutto **event**. Questo rende possibile eseguire in sequenza più funzioni ed, inoltre, permette di schedulare con precisione la sua esecuzione. Come descritto nel codice 4.2, il costrutto permette di eseguire in sequenza e quotidianamente tutte le *stored procedure* che sono state create. Per semplicità sono state inserite solo tre funzioni, che, ovviamente, corrispondono a più *stored procedure*.

```

1 CREATE EVENT etl
2 ON SCHEDULE
3 EVERY 1 DAY
4 DO
5     call Extraction();
6     call Transformation();
7     call Loading();
8 END |

```

Source Code 4.2: ETL realizzato con il costrutto *event*.

### 4.4.7 Progettazione fisica

La parte di progettazione fisica comprende un'analisi a riguardo degli indici del Data Mart che, per gli scopi di questo progetto di tesi, non sono stati presi in considerazione.

### 4.4.8 Implementazione della reportistica

Partendo dal carico di lavoro definito, è possibile ora implementare la reportistica creata attraverso l'utilizzo dello strumento Power BI. Attraverso questo strumento molto potente è possibile realizzare delle rappresentazioni grafiche che permettano di visualizzare immediatamente le informazioni necessarie. Il report principale che è stato creato considera tutte le dimensioni come dei filtri selezionabili e la misura in opera è il *Numero Ticket*. In figura 4.13 viene mostrato, dunque, un esempio di reportistica creata in questo modo.

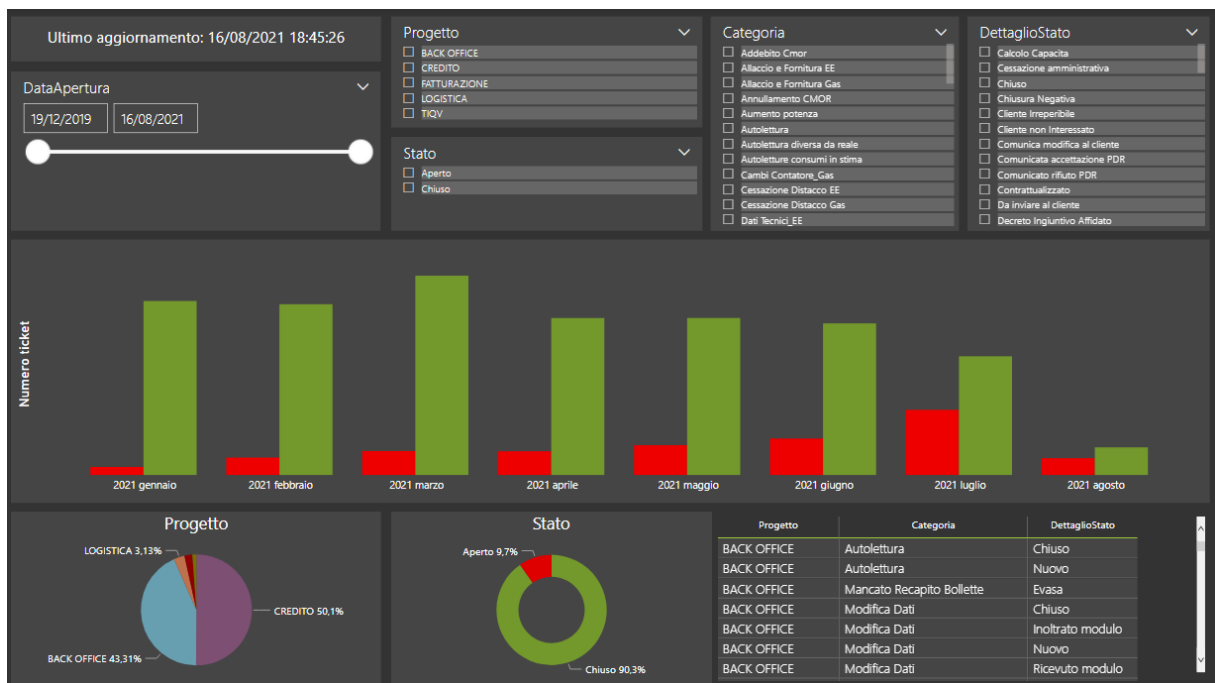


Figura 4.13: report iniziale creato con Power BI.

Si può notare come Power BI metta a disposizione diverse rappresentazioni grafiche delle informazioni. Ad esempio, è possibile inserire istogrammi, come al centro, grafici a torta oppure visioni tabellare. Inoltre, come descritto in alto in 4.13, si possono inserire filtri sulle dimensioni che permettono di selezionare uno o più elementi. Infine, occorre segnalare che è stata tenuta traccia, come si può vedere nella parte alta a sinistra, della freschezza dei dati. Su suggerimenti del manager, infatti, è sempre utile rappresentare



la data dell'ultimo aggiornamento del report in modo tale da notare immediatamente malfunzionamenti o errori.

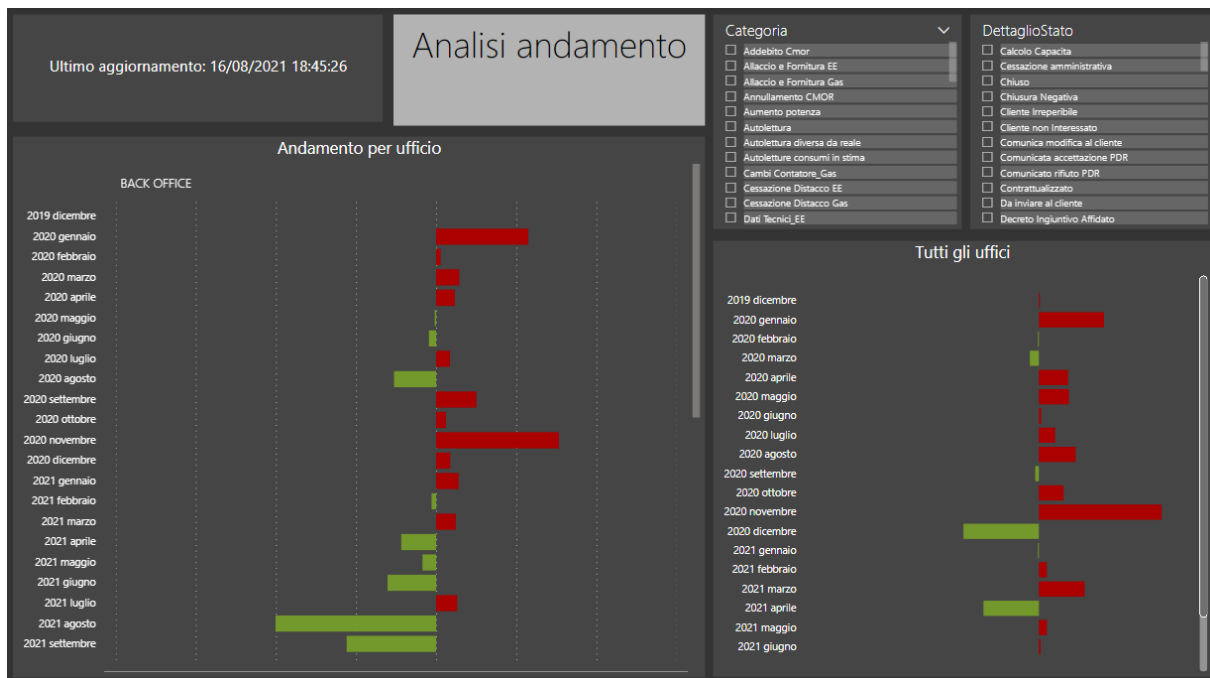


Figura 4.14: report che descrive l'andamento dei ticket.

Un altro esempio di report che è stato creato è presente in figura 4.14. Attraverso questo cruscotto è possibile evidenziare quanti siano i ticket aperti per ciascun ufficio rispetto al mese precedente. In particolare la barra diventa di colore rosso quando il conteggio del mese è maggiore rispetto a quello precedente, viceversa quando il rettangolo assume il colore verde. Si può notare che, confrontando i due report, si è cercato di mantenere lo stesso stile per tutti i report che sono stati creati. Così facendo è stato possibile abbassare la curva di apprendimento per lo strumento Power BI da parte degli utilizzatori, sia che abbiano un ruolo manageriale, sia che siano degli operativi.

#### 4.4.9 Testing

Per concludere, i report che sono stati creati sono stati testati con delle estrazioni dirette sul database e con le analisi svolte dalle figure manageriali. In particolare, si è andati a controllare in maniera incrementale tutti i passaggi dell'ETL, partendo dal numero di ticket inseriti. Attraverso estrazioni dirette sul database è stato possibile constatare che le estrazioni sono state effettuate correttamente. Di seguito, è stato controllato che le operazioni di trasformazione sui dati siano corrette: per testare ciascuna di esse, è stata

eseguita una *stored procedure* alla volta in modo tale da poter essere analizzabile separatamente. In particolare si è posto molto l'accento sul controllo che va a non considerare i ticket considerati "errati". Con questo termine si intendono tutti quei ticket che hanno il dettaglio stato uguale a **Chiusura errata o doppia** e si trovano in tale situazione a causa di aperture errate o duplicate. Con l'aiuto della responsabile dell'ufficio quello che si è fatto è stato controllare che il numero sia corretto dei ticket estratti, per poi andare a cancellare quelli errati e controllare che siano stati tenuti quelli corretti. In questo modo, seppur il procedimento richiede molto tempo, è stato possibile controllare che tutte le estrazioni e le operazioni che trasformano i dati vengano svolte correttamente.

# Capitolo 5

## Conclusioni

### 5.1 Risultati ottenuti

In considerazione agli obiettivi che erano stati posti in questo progetto di tesi, occorre sottolineare quali sono stati i risultati ottenuti. I progetti di reingegnerizzazione dei Sistemi Informativi prevedono sicuramente una forte incertezza da parte dell'azienda in esame, soprattutto legata al concetto di **cambiamento**. Spesso per pigrizia, per non uscire dalla *comfort zone* o per non doversi trovare una serie di complicazioni aggiuntive, le aziende scelgono di non attuare tali progetti che, a lungo termine, porteranno sicuramente a numerosi vantaggi, soprattutto in termini di gestione ed efficienza dei processi. Nel caso specifico di questa impresa, i limiti dell'attuale organizzazione e degli asset digitali ha portato a veder superata questa riluttanza e, sicuramente, il progetto è stato abbracciato a tutti i livelli dell'azienda. Questo clima ha portato (e porterà) a poter lavorare con un obiettivo comunque ed al raggiungimento dello stesso con serenità. Naturalmente, non sarà sicuramente facile ed immediato portare a termine tutti i miglioramenti che sono stati evidenziati. Soprattutto, l'azienda dovrà predisporre in breve tempo delle risorse umane che siano in grado di governare questo cambiamento. Su questo, ma anche su tutti gli altri risultati in termini di diagnosi che sono stati creati, l'azienda ha ritenuto che sia stata fatto un ottimo lavoro e che la strada da percorrere sia la stessa proposta.

Scendendo nello specifico attraverso la seconda parte di progetto di tesi, gli obiettivi che sono stati prefissati sono altresì ambiziosi. Infatti, soprattutto per le problematiche legate alla natura degli asset digitali, non era per nulla scontato che si raggiungessero tali risultati. Nonostante che si tratti di un prototipo molto semplice e con limitate possibilità

in termini di analisi, si è potuto dimostrare quanto sia importante al giorno d'oggi possedere all'interno del proprio organico persone che possano governare i processi aziendali ed, allo stesso tempo, gli asset digitali per poter analizzare correttamente i dati. Infatti, il progetto che ha permesso la creazione del cubo di BI porta con sé una reingegnerizzazione di tutti i processi che vengono toccati. Per fornire degli esempi pratici relativi a questo progetto di tesi, in cui tutto l'ufficio di *Customer Management* ha portato giovamento, si può considerare l'esempio delle richieste in **starvation**. Con questo termine, preso in prestito dall'ambito informatico legato ai Sistemi Operativi, intendo tutti quei ticket che non risultano essere stati gestiti correttamente e che si trovano ad essere aperti anche dopo parecchio tempo. Dopo aver mostrato, graficamente, la numerosità di questi e la tipologia, si è deciso di analizzare quali siano le cause che porta a certe situazioni. Così facendo è stato possibile evidenziare la mancanza di regole specifiche di chiusura per determinati ticket con una conseguente inefficienza in quanto a posteriori occorre perdere del tempo per chiuderli uno alla volta. A riguardo di questo esempio, va sottolineata anche l'importanza di disporre di tool di data visualization che permettano di mostrare in maniera immediata tutte le informazioni che il processo di Data Warehousing mette a disposizione.

## 5.2 Possibili sviluppi futuri

I possibili sviluppi futuri sono già stati descritti all'interno del progetto di tesi, in quanto è stata definita la strategia di reingegnerizzazione da seguire per il futuro. La sfida che si pone di fronte all'azienda è sicuramente importante e la gestione dei nuovi processi sarà fondamentale per il raggiungimento di tutti gli obiettivi.

Per quanto riguarda la parte di prototipazione del Data Mart, sarebbe molto interessante mettere in produzione quanto è stato fatto. Per raggiungere tale obiettivo, occorrerà sicuramente progettare con una tecnologia più avanzata l'ETL; infatti, il linguaggio di MySQL risulta avere delle notevoli limitazioni, soprattutto a livello di espressività e di performance. Inoltre, un ulteriore sviluppo futuro potrebbe essere quello di prevedere, come descritto in figura 4.4, la creazione di altri Data Mart con le informazioni legate ai processi di fatturazione e credito. In conclusione, il principale sviluppo futuro che dovrà essere compiuto è rappresentato dall'aumento delle informazioni presenti all'interno dei

database sorgenti. Come descritto nel corso di questo progetto di tesi, alcune informazioni risultano essere parziali (come per esempio i dati di profilazione dei clienti). Al giorno d'oggi, per poter competere nel mercato che si presenta molto competitivo, occorre avere quante più informazioni possibili e, conseguentemente, avere le conoscenze per gestirle e lavorarle.



# Ringraziamenti

Vorrei ringraziare il Professore Matteo Golfarelli per l'impegno profuso e le conoscenze trasmesse nell'ambito di questo progetto di tesi e durante tutti i suoi corsi che ho avuto il piacere di seguire. Ringrazio l'Alma Mater Studiorum - Università di Bologna per aver messo a disposizione con qualità un mix di possibilità, servizi e strumenti per il raggiungimento dei miei obiettivi. Inoltre, va ringraziata anche l'azienda che mi ha dato la possibilità di testare sul campo tutte le conoscenze acquisite durante il percorso didattico, ma anche di crescere dal punto di vista personale attraverso un clima cordiale ed ospitale. Infine, un mio caloroso ringraziamento va a tutti coloro che hanno contribuito con un messaggio, una parola, un consiglio, un supporto a rendere felici gli anni trascorsi in Università. Grazie amici, grazie colleghi di Università, grazie colleghi di tirocinio, grazie cesenati, grazie coinquilini, grazie familiari e parenti, grazie Aurora.





# Bibliografia

- [1] Harvey Maylor. Beyond the gantt chart:: Project management moving on. *European Management Journal*, 19(1):92–100, 2001.
- [2] Michael E Porter and Competitive Advantage. Creating and sustaining superior performance. *Competitive advantage*, 167:167–206, 1985.
- [3] Andriy Lesyuk. *Mastering Redmine*. Packt Publishing Ltd, 2016.
- [4] Gerold Riempp and Stephan Gieffers-Ankel. Application portfolio management: a decision-oriented view of enterprise architecture. *Information Systems and E-Business Management*, 5(4):359–378, 2007.
- [5] Toni M Somers Arik Ragowsky. Enterprise resource planning. *Journal of Management Information Systems*, 19(1):11–15, 2002.
- [6] Vineet Kumar and Werner Reinartz. *Customer relationship management*. Springer, 2018.
- [7] Matteo Golfarelli, Dario Maio, and Stefano Rizzi. The dimensional fact model: A conceptual model for data warehouses. *International Journal of Cooperative Information Systems*, 7(02n03):215–247.
- [8] Mysql - the world's most popular open source database. <https://www.mysql.com/>. Ultimo accesso: 2021-11-20.
- [9] Nishtha Jatana, Sahil Puri, Mehak Ahuja, Ishita Kathuria, and Dishant Gosain. A survey and comparison of relational and non-relational database. *International Journal of Engineering Research & Technology*, 1(6):1–5, 2012.

- [10] Alexander Stanik, Matthias Hovestadt, and Odej Kao. Hardware as a service (haas): Physical and virtual hardware on demand. In *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings*, pages 149–154. IEEE, 2012.
- [11] Google Cloud Storage. <https://cloud.google.com/storage/>. Ultimo accesso: 2021-12-04.
- [12] Stored Programs - MySQL :: Developer Zone. [https://dev.mysql.com/doc/dev/mysql-server/latest/stored\\_programs.html](https://dev.mysql.com/doc/dev/mysql-server/latest/stored_programs.html). Ultimo accesso: 2021-11-25.
- [13] Microsoft Power BI. <https://powerbi.microsoft.com/it-it/>. Ultimo accesso: 2021-11-20.
- [14] Surajit Chaudhuri, Umeshwar Dayal, and Vivek Narasayya. An overview of business intelligence technology. *Communications of the ACM*, 54(8):88–98, 2011.
- [15] T Braun, M Günter, M Kasumi, and I Khalil. Virtual private network architecture. *Charging and Accounting Technology for the Internet (Aug. 1, 1999)(VPNA)*, 1999.
- [16] Sean Turner. Transport layer security. *IEEE Internet Computing*, 18(6):60–63, 2014.
- [17] Openvpn - secure access and network connectivity reimaged. <https://openvpn.net/>. Ultimo accesso: 2021-11-20.
- [18] MySQL Workbench. <https://www.mysql.com/products/workbench/>. Ultimo accesso: 2021-11-20.