UNICUSANO

Università degli Studi Niccolò Cusano - Telematica Roma

*Corso di Perfezionamento e aggiornamento professionale*
*in*
*"Data analyst"*

# IL DATA WAREHOUSE NELLA BUSINESS INTELLIGENCE


**Candidato**
***ANDREA MECCHIA***

**Relatore**
***Chiar.mo Prof. GIUSEPPE MICELI***


**ANNO ACCADEMICO 2020/2021**

## INDICE

# CAPITOLO 1 - Introduzione

## 1.1 Scopo della tesi

Nel presente lavoro descriverò una delle tecnologie consolidate di business intelligence: il data warehouse. Si tratta di una particolare architettura software che permette di effettuare diverse operazioni su dati tra loro eterogenei per tipo, fonte e formato e trasformarli in informazioni utili a certe decisioni. In un data warehouse, le fasi che coinvolgono i dati possono essere raggruppate nei seguenti passaggi:

- Selezionare;
- Classificare;
- Immagazzinare;
- Interrogare;
- Compiere operazioni.

Si potrebbe dire meglio, forse, che per data warehouse si intende un insieme di metodologie e strumenti software per estrarre informazioni dai dati. Nell'esporre gli argomenti seguirò un percorso logico che procede dal generale al particolare.

Infatti, per illustrare correttamente la scopo, la logica, la tecnica di costruzione e il funzionamento di un data warehouse, non si può prescindere da una visione di insieme dei dati, del loro uso e dello scopo con cui vengono utilizzati oggi. Successivamente parlerò dell'integrazione necessaria del data warehouse con il resto dei sistemi informativi aziendali e in particolare con i moderni software gestionali ERP. Procederò parlando dei dettagli tecnici informatici, cominciando con l'esporre le caratteristiche essenziali delle moderne basi dati relazionali. Non si può prescindere dal soffermarsi, seppur brevemente, su questo argomento, in quanto le basi dati operazionali e il data warehosue sono intimamente connessi a livello di dati e architettura informatica.

Passerò quindi alla descrizione dettagliata di un data warehouse e di tutte le sue componenti principali. Continuerò parlando delle linee guida da seguire e degli errori da evitare quando si approccia un progetto di data warehouse. Terminerò con una rapida elencazione dei software e delle tecnologie presenti oggi sul mercato per realizzare o interrogare un data warehouse.

## CAPITOLO 2 - I big data e la business intelligence

### 2.1 Cosa sono i big data

Per capire che cos'è il data warehosue bisogna introdurre il concetto di business intelligence, ma per farlo, comincerò da quello di Big Data, perché oggi viviamo nell'era dell'informazione che guida le decisioni. In un mondo sempre più competitivo, le nicchie di mercato, per qualunque tipo di produzione di beni e servizi, si fanno sempre più rare e affollate. Un mercato e la sua nicchia, per essere tali devono essere profittevoli e la ricerca dell'efficienza e dell'efficacia delle decisioni è la chiave di questa profittabilità.

Le aziende sono favorite nelle corsa verso il successo se sanno utilizzare le informazioni per prendere decisioni tempestive e consapevoli, che arricchiscano la loro organizzazione. Tuttavia, selezionare, immagazzinare e gestire una mole di dati come quella a attuale è decisamente difficile ed è diventata la sfida primaria dalla quale dipendono il buon esito delle attività produttive. Tutti questi dati, devono diventare informazione e l'informazione deve diventare saggezza decisionale nell'approcciarsi all'azione imprenditoriale. Oggi si parla infatti di Business intelligence; l'Intelligence è la conoscenza ed è utilizzata per capire quanto è già successo e per prevedere ciò che accadrà. *"La conoscenza può essere estratta dai dati in modo: passivo, cioè, attraverso criteri di analisi suggeriti dal decision maker (in questa categoria ricadono i metodi di analisi statistica classica, i sistemi di interrogazione e reporting inclusa l'analisi multidimensionale - OLAP) oppure, attivo, cioè, con l'ausilio di modelli matematici di apprendimento induttivo e di ottimizzazione (in questa categoria ricadono la statistica inferenziale e i modelli di apprendimento induttivo, le tecniche di data mining)."* I Big data, possono essere definiti, come *"ampi insiemi di dati che possono essere identificati sulla base della mole, della velocità e della presenza di differenti sottoinsiemi, da cui vengono estratte le informazioni."*.[1] Sono caratterizzati dai seguenti elementi:

- Volume;
- Varietà;
- Velocità;
- Veridicità;
- Valore.

I Big Data sono voluminosi in termini di byte. Si tratta di dati di diverso tipo (di qui la variabilità). Per quanto riguarda la velocità essa "*è riferita alla rapidità con la quale i dati si*

[1] Andrea Moretto Wiel, Lorenzo Montibeller, Big Data, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 16.

*generano, si raccolgono, si aggiornano e si elaborano; diventando poi informazioni che si possono tradurre in real-time analysis e instant-decision making."*[2]. Per quanto riguarda la veridicità *"si riferisce, invece, alla bontà delle informazioni contenute al loro grado di rappresentatività. Tale caratteristica si muove dunque nella direzione della qualità dei dati presenti nei dataset, affinché le decisioni data-driven siano integre da un punto di vista"*.[3]

I Big Data si differenziano dai Little Data che *"sono insiemi caratterizzati da una quantità modesta di dati al loro interno ma che, al tempo stesso, sono anch'essi caratterizzati da notevole velocità e varietà. Rappresentano l'oggetto di analisi quando si parla di data-driven innovation in ambito di piccole e medie imprese."*[4] I Big Data possono provenire anche da fonti esterne all'azienda, mentre nel data warehouse, come dirò in seguito, i dati perlopiù, sono interni all'azienda (anche se non sempre). Quando si parla di conoscenza derivante dai dati possiamo schematizzare questo processo raffigurandolo in questa piramide della conoscenza (DIKW)[5]


La base è rappresentato dai dati, seguono l'informazione, la conoscenza, la saggezza intesa come consapevolezza. La saggezza per ora resta prerogativa umana, tuttavia la ricerca sull'intelligenza artificiale è molto attiva e sta cercando di fare evolvere le tecniche e le tecnologie legate alla cosiddetta "machine learning" (apprendimento automatico), che rappresenta ad oggi, l'aspetto più spinto e futurista dell'approccio alla gestione dei dati. L'ambizione dei ricercatori è quella di portare l'apprendimento alla saggezza.

[2] Andrea Moretto Wiel, Lorenzo Montibeller, Big Data, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 16.
[3] Op. cit. pagina 16.
[4] Op. cit. pagina 18.
[5] Fig. Acronimo di "data, information, knowledge and wisdom". Andrea Moretto Wiel, Lorenzo Montibeller, Big Data, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 14.

## 2.2 Cos'è la business intelligence

Qual è il modo corretto per avere a che fare con i dati? Saperli estrarre e usare efficacemente? Detto altrimenti, in che modo i dati possono essere selezionati, organizzati, utilizzati per estrarre una conoscenza utile in termini di competitività e crescita economica? In breve, come già accennato, dobbiamo saper trasformare i dati in informazioni che a loro volta devono diventare consapevolezza; questo percorso di affinamento può essere riassunto in modo più dettagliato in questa figura piramidale. [6]


Guardando la piramide, *"alla cui base si pongono i dati che hanno natura pubblica e che consistono nella parte più elementare dell'informazione. Si tratta del caso di bilanci delle aziende concorrenti, di nuovi prodotti proposti sul mercato o di nuove tecniche di vendita, oppure ancora, di marchi o brevetti gestiti dal concorrente ma potrà trattarsi, anche, della corretta individuazione di probabili minacce al proprio business o, ancora, di nuove opportunità da sfruttare prima di altri competitor."*. [7] Dalla base si sale attraverso successive trasformazioni. Per esempio il secondo gradino potrebbe riguardare dati di bilancio riclassificati. Salendo i gradini, le informazioni diventano sempre più auto-consistenti e portatrici di valore in sé, il più importante dei quali è quello predittivo, al fine di migliorare il processo decisionale. In particolare: *"I livelli più in alto e che compongono l'area della Business Intelligence sono quelli che definiamo con il termine Intelligence a cui si giunge per effetto dell'analisi di dati e informazioni e che assume importanza per il suo contenuto anticipativo rispetto al comportamento che adotterà il concorrente, frutto dell'analisi previsionale."*[8] e ancora: *"L'analisi strategica deve tendere, perciò, ad influire sul futuro*

[6] Fig. piramidale tratta da Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 6.
[7] Op. cit. pagina 6.
[8] Op. cit. pagina 7.

*indicando le metodologie per regolare il corso degli eventi durante una crisi, assegnando obiettivi a lunga scadenza con l'esame delle minacce attuali ed emergenti.".*[9]

La business intelligence (BI) è il tentativo di "*produrre conoscenza da utilizzare nei processi decisionali.". [10]* In termini applicativi, la business intelligence è quell'insieme di strumenti software che servono per attuare questo processo di conoscenze e il data warehouse, vi rientra a pieno titolo. Si tratta di strumenti che supportano l'attività decisionale. Da questo ne consegue che il compito di una data analyst è quello di *"gestire le informazioni per la formazione della strategia, consentendo al management di poter fare affidamento con informazioni e soprattutto intelligence e non solo con meri dati."*[11]

Da quanto detto finora è intuibile che la raccolta e la gestione dei dati non può essere episodica ma deve essere sistematica, coordinata e finalizzata, almeno nelle grandi organizzazioni. In questo caso si parla di "ciclo di intelligence" che consta delle seguenti fasi:

- Esperienza;
- Ottimizzazione;
- Interazione;
- Analisi.

In base a quanto sopra è utile precisare che, per quanto riguarda l'acquisizione dei dati, "*non esiste una soluzione di continuità tra la risposta alle domande e la nuova richiesta di informazione derivante dall'elaborazione precedente".* [12] e in generale, *"Il ciclo di BI consiste, quindi, in una sequenza logica di attività che trova inizio nel Dato e porta all'estrazione e all'utilizzo della conoscenza, e alla verifica della bontà delle decisioni assunte.".* [13] Non bisogna pensare a questa serie di passi come un percorso lineare, ma come un procedere circolare, iterativo, dove i vari passaggi, costituiscono le parti di un'unica esperienza. Il ciclo di intelligence in un contesto aziendale può essere declinato nelle fasi seguenti:

- Pianficazione;
- Raccolta;
- Gestione;
- Interpretazione;

[9] Op. cit. Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 7.
[10] Op. cit. pagina 8.
[11] Op. cit. pagina 9.
[12] Op. cit. pagina 9.
[13] Op. cit..pagina 10.

- Diffusione.

La fase di raccolta infatti non è la prima, come si sarebbe indotti a pensare, ma è la seconda, proprio perché prima di procedere è importante definire con chiarezza gli scopi per i quali i dati dovranno essere raccolti e gestiti. La seconda fase è quella appunto della raccolta, che dovrà essere sistematica ed efficace. In questa fase si individuano le fonti dati e si predispongono le basi dati e i repository per lo stoccaggio dei dati stessi. La gestione dell'informazione è il flusso di azioni da intraprendere per raccogliere e rendere disponibile i dati, in essa *"trova evidenza l'architettura della struttura di raccolta dell'informazione".*[14]

L'interpretazione e l'analisi dei dati è il cuore stesso della trasformazione dei dati in informazioni e si risolve con la produzione di una prima reportistica. Infine, la diffusione riguarda la reportistica da consegnare al management aziendale responsabile delle decisioni, per aiutare la formulazione delle decisioni stesse.

Ho detto che il ciclo di intelligence non può essere lasciato al caso ma deve essere un'attività aziendale consolidata che può essere affidata ad esperti interni o esterni all'organizzazione. In genere, nelle grandi realtà viene identificato personale interno con competenze specifiche che sarà anche referente verso fornitori esterni di software acquistato e\o commissionato ad hoc. In quest'ultimo caso, il personale interno dovrà anche fare da mediatore tra il fornitore esterno e il management dell'azienda stessa, destinatario finale dei risultati delle applicazioni di business intelligence. In altri situazioni aziendali gli esperti sono esterni. In ogni caso compito degli esperti del settore sarà quello di fornire supporto alle azioni seguenti di management: [15]

- Identificare i trend di cambiamento nei contesti competitivi in cui opera o intende operare l'azienda;
- Individuare in anticipo la strategia e l'azione che sarà posta in essere dai concorrenti;
- Individuare in anticipo le opportune reazioni che l'azienda potrà opporre;
- Giocare d'anticipo.

In altre situazioni ancora *"sono gli stessi manager a curare l'attività operativa di raccolta dei Dati ed elaborarli, applicando i risultati direttamente al processo decisionale di propria competenza. In questi casi, piuttosto che un vero e proprio report, si avrà una serie di appunti*

[14] Op cit. Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 12.
[15] Op cit. pagine 12 e 13.

*operativi. Tuttavia, si tratta di sistemi di intelligence che non garantiscono il giusto livelli di diffusione delle informazioni nel contesto aziendale di riferimento.”*.[16]

Un accenno doveroso deve essere fatto rispetto ai concetti di fonte dati e di analisi dei dati utilizzati nel contesto della business intelligence, perché si tratta di due elementi essenziali per il funzionamento di sistemi come questi. Per fonte dati si intende qualunque fonte e qualunque tipo di file, tuttavia il concetto va esteso anche a fonti analogiche e non digitali, come la carta stampate; si può includere anche materiale fotografico e di qualunque altro genere. Naturalmente, spesso si ha a che fare con vari tipi di file, come data base relazionali, di cui parlerò più diffusamente oltre, oppure file “piatti”, come file di testo, fogli di calcolo, file csv e altri di cui dirò meglio in seguito.

Per quanto riguarda l’analisi e l’interpretazione dei dati è utile chiarire che “*l’analisi «è un metodo di studio e di ricerca consistente nello scomporre un tutto nelle sue singole componenti allo scopo di esaminarle e definirle. L'azione di analisi è quel fatto o documento che indica i risultati o le conclusioni di tale lavoro”.*[17] Inoltre, “*l'analisi non può avere solo il significato di "studio, esame accurato" e deve fare, invece, indispensabile riferimento al "metodo" su cui si basa.”.* [18] Ribadisco che il dato diventa intelligence quando, sin dalla fase di raccolta del dato stesso si pensa ad una specifica finalità per l’utilizzo delle informazioni. L’attività di analisi integra tra loro dati e informazioni complesse, svelandone le interrelazioni, mostrando deduzioni che portano ad altre conclusioni, che diventano esse stesse nuove informazioni utili per prendere decisioni.

L’analista inoltre individua alcuni indicatori per monitorare fenomeni o situazioni e identifica, riassume e schematizza situazioni tipo o comportamenti ripetuti. Le informazioni che apportano “intelligence” ad un organizzazione non sono soltanto quelle riservate al management apicale, ma anche quelle dirette al middle management. Le dimensioni di un’azienda e l’orizzonte temporale in cui si situano le decisioni fanno sì che l’attività di analisi possa essere di due tipi:

- Strategica;
- Operativa.

[16] Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello “Data Analyst” Unicusano, Roma, 2017. Pagina 13.
[17] Op. cit. Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello “Data Analyst” Unicusano, Roma, 2017 pagina 14.
[18]Op. cit. pagina 14.

## 2.3 Strumenti della business intelligence

Nell'ambito della business intelligence (BI), la conoscenza e l'utilità predittiva si ottiene fondamentalmente con queste tre metodologie:

- Esplorazione delle informazioni;
- Data mining;
- Analisi what-if.

La prima consiste in analisi statistiche sui dati per avere dati di sintesi, la seconda esplora grandi quantità di dati al fine di individuare correlazioni tra fenomeni diversi e la terza è una simulazione predittiva che cerca di monitorare l'andamento di sistemi complessi. L'analisi what-if in particolare *"misura come i cambiamenti su un insieme di variabili indipendenti hanno influenza su un insieme di variabili dipendenti, utilizzando un sistema di simulazione. In pratica, l'analisi what-if offre la possibilità di avvalersi di sistemi predittivi, utili per verificare l'impatto che determinate scelte possono avere sul sistema."*.[19]

Nella business intelligence per arrivare ai risultati attesi è necessario passare per i seguenti passaggi:[20]

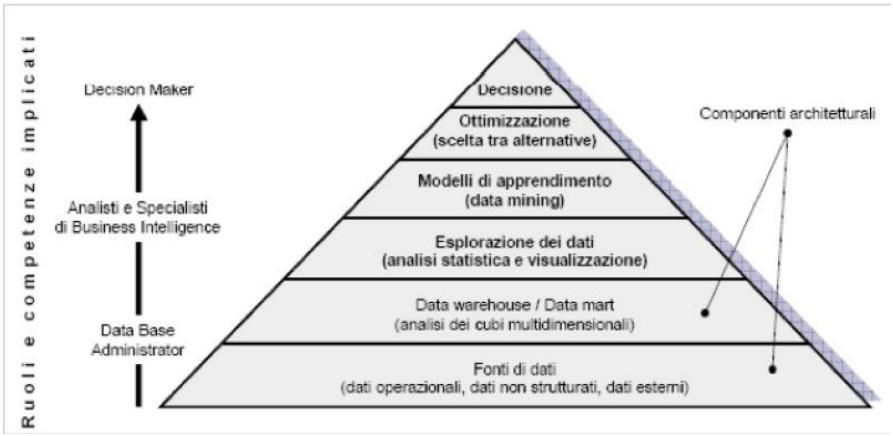
- Comprendere l'origine dei dati e di valutarne la veridicità e attendibilità;
- Riconoscere le specifiche problematiche aziendali e finalizzare l'analisi dei dati alla crescita del business;
- Generare applicazioni automatizzate idonee a selezionare le decisioni più opportune in situazioni complesse.

Il data warehousing fa parte della più ampia famiglia di tecniche, strumenti, realizzate con apposito software, che dagli 80 in poi hanno assunto un'importanza e un valore sempre crescenti.

Nell'economia attuale si parla di "Data economy". I dati non solo rappresentano conoscenza ma sono sorgente di valore e sono oggetto di scambio. Le imprese italiane acquistano dati che riguardano il loro mercato di appartenenza oppure riguardanti i comportamenti dei consumatori. L'acquisto è effettuato presso aziende specializzate, chiamate Data provider. Possiamo considerare il data warehouse come all'inizio del ciclo dell'intelligence e, a mio avviso, ne costituisce uno dei pilastri portanti e irrinunciabili per

[19] Op. cit. Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagine 7-8.
[20] Op. cit. pagine 8.

un'organizzazione che vuole avere vantaggi competitivi dalle informazioni. L'immagine di seguito riportata riassume efficacemente quanto detto.[21]




## CAPITOLO 3 - Processi aziendali e sistemi informativi

### 3.1 La gestione aziendale per processi

Ogni azienda è un insieme di uomini, mezzi e informazioni che può essere visto sotto un duplice aspetto:

- Statico-organizzativo;
- Dinamico-trasformativo.

L'aspetto dinamico-trasformativo è costituito dai cosiddetti processi aziendali, che non solo rappresentano una serie di procedure ma incorporano una dimensione del divenire, poiché l'azienda è un sistema aperto che comunica con il mondo esterno, dal quale è influenzata e al quale deve adattarsi. Il modello per processi è una rappresentazione dell'attività dell'azienda i cui una serie di valori (e specularmente di informazioni) entrano nel processo produttivo per trasformarsi poi nell'output desiderato dal cliente finale.

Più in dettaglio, un processo è una serie di attività, collegate tra loro in modo logico, che accettano un input e producono un output finale, che ha un'utilità per un cliente esterno o per l'organizzazione stessa. I processi vengono classificati e "mappati" e fanno parte di un modo moderno di approcciare produzione e commercializzazione di beni e servizi in un'ottica di qualità certificabile; un processo si compone di sei elementi:

- Materie;
- Macchinari;

[21] Op. cit.. Giuseppe Miceli, Fondamenti e tecniche di Business Intelligence, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017 pagina 32.

- Metodi;
- Misure;
- Ambiente.

Il concetto di processo è molto utile nella progettazione del data warehouse, il quale è un potente strumento informativo, che non può prescindere dell'analisi dei processi aziendali.

### 3.2 I sistemi operazionali e software ERP

Il data warehouse fa parte dei software della business intelligence, che a sua volta, fanno parte della più ampia famiglia dei sistemi informativi aziendali. Il data warehouse infatti prende i dati dai cosiddetti sistemi transazionali, detti anche operazionali. Tra questi sicuramente un ruolo di primissimo piano è ricoperto dai software ERP (Enterprise Resource Planning). I software ERP sono dei software gestionali con in quali è possibile gestire tutte le operazioni che avvengono nei processi aziendali. Faccio subito un esempio per entrare nel vivo; con il software ERP posso emettere una fattura di vendita di prodotti, che una volta convalidata definitivamente attiva una serie di operazioni come quelle seguenti:

- Genera automaticamente la scrittura in partita doppia nella contabilità;
- Aggiorna lo stato dell'ordine di vendita;
- Scadenza l'impegno di incasso secondo le condizioni applicate al cliente;
- Segnala al magazzino la merce in uscita.

Una sola operazione coinvolge più reparti dell'azienda e percorre tutto il ciclo attivo (quello delle vendite appunto). A titolo esemplificativo ma non esaustivo, i processi che possono essere gestiti con un software ERP sono le seguenti: [22]

**Amministrazione**

- Contabilità generale;
- Gestione tesoreria;
- Gestione cespiti.

**Controllo di gestione**

- Contabilità analitica;
- Budget;
- Analisi degli scostamenti.

**Gestione del personale**

[22] Alberto Quagli, Paola R. Dameri, Iacopo E. Inghirami (a cura di) I SISTEMI INFORMATIVI GESTIONALI, FrancoAngeli, 2005 pagina 26.

- Assunzione, risoluzione e altre vicende del rapporto di lavoro;
- Elaborazione di paghe e contributi;
- Gestione degli orari e della turnistica.

**Ciclo attivo (vendite)**

- Emissione delle fatture di vendita;
- Statistiche di vendita;
- Gestione post-vendita;
- Gestione agenti e provvigioni.

**Ciclo passivo (acquisti)**

- Emissione ordini di acquisto;
- Registrazione fatture di acquisto;
- Confronto con gli ordini di acquisto;
- Statistiche sugli acquisti.

**Magazzino**

- Movimentazione delle merci;
- Gestione dei lotti;
- Gestione delle scorte.

**Gestione della produzione**

- MRP II;[23]
- CRP;[24]
- Gestione manutenzione;
- Pianificazione della produzione.

**Altri moduli**

- Gestione documentale;
- Gestione intranet aziendale;
- CRM;
- Gestione della qualità.

Un software ERP deve garantire:

- Un'integrazione di tra scritture contabili;
- La sicurezza dei dati;
- Velocità di elaborazione.

[23] Acronimo di Manufacturing Resources Planning (Pianificazione delle risorse di produzione).
[24] Acronimo di Capacity requirements planning. (Determinazione delle risorse uomo-macchina).

Il software ERP è tale se gestisce l'azienda a tutto tondo e viene utilizzato nelle grandi aziende che hanno anche più sedi operative e centinaia o migliaia di addetti. Tuttavia, negli ultimi anni, il mercato ha visto la comparsa di soluzioni appositamente sviluppate per la media impresa, soprattutto nel contesto italiano. L'architettura software di un sistema ERP è caratterizzata dalla seguente suddivisione logica:

- Il computer client che è usato dall'utente;
- Un livello server che ospita l'applicazione;
- Un server di database.

I software ERP prima di entrare in funzione, in genere vengono "parametrizzati". La parametrizzazione è il caricamento e\o la personalizzazione di alcune informazioni preliminari, senza le quali, il software non può essere utilizzato. Queste informazioni vengono salvate in apposite tabelle della base dati (tabelle parametriche), dedicate a questo tipo di esigenza, che a volte, possono essere anche numerose. Per questo motivo le operazioni di parametrizzazione non sono svolte, come in un normale gestionale, dall'utente finale, che magari aggiunge o modifica dei codici IVA o delle causali contabili, ma richiedono un vero e proprio progetto specifico di "customizzazione" cioè di personalizzazione. Come tutti gli altri progetti informatici (compresi quelli di data warehouse), anche quello di parametrizzazione, termina con alcune fasi di post-produzione:

- Documentazione;
- Test;
- Rilascio;
- Formazione agli utenti finali.

Quando invece, nonostante la parametrizzazione, l'azienda cliente non ottiene in modo completo una personalizzazione soddisfacente, si dovrà procedere lavorando sul codice sorgente dell'ERP per modificare funzionalità esistenti o crearne di nuove. Il caso tipico (ma certo non l'unico) di software ERP personalizzabile sia con la parametrizzazione che con modifica al codice è il SAP R\3. Una delle caratteristiche dei software ERP è la loro capacità di apportare conoscenza, all'interno dell'organizzazione che li usa. Questa conoscenza è data dal cosiddetto "business model", cioè funzionalità che traggono origine da "best practice" di vari settori produttivi.

Un'azienda cliente ha un vantaggio nell'usare un software ERP: la possibilità di imparare, di arricchire il proprio Know how; la parametrizzazione quindi comporta sia la personalizzazione di funzioni, sia l'implementazione di un business model. In un certo senso quindi, una parte di "intelligence" è già compresa nei sistemi operazionali ERP, perché viene

incorporata come tipologia di esperienza. Utilizzare un business model, apporta cambiamenti ad alcuni modi di lavorare, costringe a ripensare procedure organizzative, spingendo al miglioramento. Questi cambiamenti vengono affrontati in modo più efficiente all'interno di grandi organizzazioni, dove le persone sono abituate di più a lavorare seguendo procedure formalizzate e prestabilite, piuttosto che basate sulle abitudini personali. Questo non vuol dire che l'implementazione di un software ERP, come del resto anche un data warehouse, non comporti resistenze interne anche notevoli, anche a livello di top del management. Inevitabilmente, soluzioni software potenti ed estese come gli ERP e il data warehouse, sono accelleratori formidabili di conoscenza e per essere realizzati hanno bisogno, per forza di cose, di cambiamenti anche profondi ma necessari.

Una particolarità dei sistemi ERP italiani è la presenza nel nostro mercato dei cosiddetti software definiti come "light ERP"[25]. All'estero i sistemi ERP sono stati pensati inizialmente come un'estensione dei software gestionali dedicati alla produzione. In Italia, invece i software ERP sono stati pensati come evoluzione dei software gestionali dedicati alla contabilità. Questa differenza di visione è dovuta a due fattori fondamentali che sono tipici del contesto produttivo italiano:

- Le dimensioni aziendali;
- La complessità della normativa fiscale italiana.

Nel primo caso, l'Italia è un paese che, come noto, basa la sua ossatura produttiva sulle piccole e medie impresa, nel secondo caso, la peculiarità dei nostri numerosi adempimenti amministrativi e fiscali richiede una particolare attenzione. La nostra contabilità è basata sul sistema del reddito, mentre negli altri paesi vige spesso il sistema patrimoniale, quindi la personalizzazione di un software ERP verso la nostra realtà potrebbe essere difficoltosa. La stessa cosa dicasi per la gestione automatizzata delle ritenute d'acconto che negli altri paesi non vengono applicate.

### 3.3 Il data warehouse e i sistemi di business intelligence

Come ho già accennato e come dirò più diffusamente avanti, il data warehouse è un modo particolare di trattare le informazioni, soprattutto in queste macro-fasi:

- Selezione delle fonti dati;
- Normalizzazione dei dati e correzione di errori;
- Interrogazione dei dati e reportistica.

[25] Op. cit. Alberto Quagli, Paola R. Dameri, Iacopo E. Inghirami (a cura di) I SISTEMI INFORMATIVI GESTIONALI, FrancoAngeli, 2005 pagina 11.

Esistono software ERP che hanno un loro modulo di data warehousing; laddove questo modulo non è presente o previsto, e se il data base che utilizza il software ERP lo permette, è possibile creare un data warehouse o comunque una forma di analisi multidimensionale dei dati, ricorrendo a strumenti appositi. È il caso per esempio di Microsoft SQL Analysis Service che è lo strumento gratuito del data base Microsoft SQL Server. Più in seguito nella trattazione spiegherò meglio la differenza tra un data warehouse e l'analisi multidimensionale dei dati. Quest'ultima infatti è una tecnica di analisi ma il data warehosue è un'architettura software con specifiche caratteristiche.

Da quanto detto finora è evidente che anche le PMI devono e dovranno sempre di più dotarsi di strumenti di business intelligence, tra i quali spiccano, per primi, quelli di data warehouse e di analisi multidimensionale dei dati. Secondo quanto descritto da Giuseppe Scribani già nel 2000, si può formulare un'equazione che descriva il grado di business intelligence di un'azienda. L'equazione è la seguente: QBI = MI + MS + MC[26]. La definizione estesa degli addendi è la seguente:

- MI: Managerial Index;
- MS: Market Structure;
- MC: Market Contingency.

Detto altrimenti, il grado di business intelligence presente in azienda dipenderebbe da tre fattori:

- Il grado di managerialità (maturità e consapevolezza del management);
- La struttura del mercato (più o meno facile da modellizzare);
- Situazione contingente di mercato (nuovi concorrenti, nuovi prodotti, nuovi processi produttivi).

Come dirò più avanti nella trattazione, un progetto di data warehouse, spesso deve scontrarsi con vincoli esterni e resistenze interne all'azienda. Deve coinvolgere più soggetti con ruoli determinati e necessita diverse. Ci deve essere sempre e fino in fondo, una grande consapevolezza sull'utilità e il vantaggio competitivo che un progetto del genere può apportare a tutta l'organizzazione.

[26] Op. cit. Alberto Quagli, Paola R. Dameri, Iacopo E. Inghirami (a cura di) I SISTEMI INFORMATIVI GESTIONALI, FrancoAngeli, 2005 pagina 200.

# CAPITOLO 4 - Il modello relazionale delle basi dati

## 4.1 Introduzione

Le basi dati relazionali (Relational database management system - RDBMS) sono una diffusa architettura software per immagazzinare, interrogare e manipolare dati. Quasi tutti i software gestionali ne fanno uso. Non si può parlare di data warehouse senza spiegare i concetti essenziali delle basi dati relazionali come tabelle, chiavi, attributi, normalizzazione, relazioni, perché capire questi elementi sarà utile per proseguire con la descrizione del data warehouse. Il modello logico relazionale mette in rapporto diversi dati dopo averli opportunamente suddivisi e memorizzati in tabelle organizzate per colonne (campi) e righe (record); i campi prendono il nome di attributi e il funzionamento di questa architettura si basa sulla matematica degli insiemi.

## 4.2 Il modello relazionale e i suoi vantaggi

L'idea vincente alla base di questa particolare modalità di trattare i dati risiede di fatto in alcuni vantaggi operativi di grande utilità che si possono riassumere in due tipi:

- Non ripetizione dei dati;
- Navigazione tra i dati (quindi combinazione e calcolo tra loro) in molte forme diverse.

Per ottenere questo risultato occorre innanzitutto creare tabelle con informazioni essenziali rispetto al cosiddetto "dominio del problema", scomponendo le informazioni il più possibile e raggruppandole in gruppi coerenti (le tabelle appunto). Per esempio, la tabella degli ordini non conterrà informazioni sui clienti ma un riferimento, per ogni ordine, alla tabella dei clienti che, a sua volta, conterrà informazioni riguardanti questi ultimi, come la denominazione, l'indirizzo, la partita iva e così via. Per creare una base dati efficiente dal punto di vista logico e delle prestazioni servono quindi due accorgimenti:

- Scomporre i dati;
- Classificarli in gruppi omogenei (le tabelle).

Oltre a quanto detto sopra occorre strutturare ogni tabella in modo che ogni record sia identificato univocamente da una "chiave". La chiave è composta da uno o più campi della tabella, tali da costituire un valore univoco di riferimento. Le chiavi non sono solo primarie e univoche ma, in generale, la chiave è un campo che viene scelto per poter fare ricerche. Le tabelle sono collegate tra loro attraverso le chiavi. Questa struttura permette la navigazione e l'estrazione dei dati di interesse attraverso dei percorsi che l'utente può scegliere usando un apposito linguaggio: lo Structured Query Language (SQL).

### 4.3 La normalizzazione delle basi dati

Tornando al processo di scomposizione e classificazione, si ottengono risultati soddisfacenti rispettando le cosiddette “forme normali”. In altre parole, un data base relazionale normalizzato è garanzia di efficienza ed efficacia; le forme normali sono cinque.

- **Prima forma normale**. In una base dati il contenuto di una colonna deve avere un solo valore.
- **Seconda forma normale.** In una base dati tutte le colonne di una riga possono essere identificate da una “chiave” che svolge funzioni di ricerca e ordinamento.
- **Terza forma normale**. In una base dati tutte le colonne che non sono chiavi sono mutuamente indipendenti.
- **Quarta e quinta forma normale**. Non mi soffermo in quanto, nella pratica, vengono spesso trascurate perché a fronte di un rigore usato per **eliminare la ridondanza dei dati,** corrisponde un degrado delle prestazioni.

Ogni forma normale, comprende la precedente, la normalizzazione procede quindi per accumulazione di requisiti. Un semplice esempio di normalizzazione in prima forma è quello degli indirizzi. Quando si scrive città e provincia, non si dovrebbero ripeterne i nomi, sui record ad ogni occorrenza, bensì creare delle chiavi esterne che si colleghino a tabelle diverse che contengono gli elenchi di città e provincie. Tornando invece all’esempio citato prima, un ordine può riguardare un cliente, un cliente può essere di un certo tipo e appartenere ad una certa regione, per cui nella tabella dei clienti ci sarà il riferimento (chiave esterna) alla tabella delle regioni e ci sarà anche la chiave esterna come riferimento alla tabella delle tipologie di clienti. Ancora un esempio: uno studente (tabella degli studenti) può essere iscritto ad uno più corsi (tabella dei corsi). Laddove quindi possiamo scomporre, lo dobbiamo fare, per mettere poi in relazione le informazioni che potremo navigare.

Il linguaggio SQL prevede anche la possibilità di fare operazioni matematiche e alcuni RDBMS consentono di usare il SQL per scrivere veri e propri programmi. Si pensi al Transact SQL (T-SQL) di SQL Server o al PL-SQL di Oracle, entrambi potenti linguaggi di programmazione dei dati.

Come detto precedentemente le basi dati relazionali sono il cardine del funzionamento di moltissimi software e i dati che contengo servono per alimentare i data warehouse (anche se i dati che alimentano il data warehouse possono venire anche da fonti diverse) e vengono definiti come dati “operazionali”. Concludo questa breve introduzione alle basi dati relazionali approfondendo alcuni concetti accennati sopra.

### 4.5 Entità

In termini più specifici, un'entità è una tabella della base dati ma è difficile pervenire ad una definizione logica più generale, se non dicendo che si tratta di qualunque cosa di cui si vogliono salvare le informazioni. Le entità sono rappresentative di eventi o di oggetti o di ambiti di riferimento che fanno parte della realtà. Nell'ambito delle basi dati relazionali la realtà del problema da rappresentare viene anche definita come "dominio del problema". Il processo di scomposizione e organizzazione dei dati che segue le cinque forme normali può aiutare a definire perimetro e contenuto delle entità.

### 4.6 Attributi e chiavi

Gli attributi sono i campi nelle tabelle, ovvero potremmo definirli come i fatti che popolano un'entità. Il concetto di fatto tornerà in seguito nella trattazione quando parlerò del data warehouse. Per scegliere quali attributi devono avere le entità bisogna fare riferimento a cosa si vuole conoscere e a quale sia il significato dei dati rispetto ad un certo contesto e l'uso che se ne farà. La progettazione della base dati deve consentire di avere un sistema relazionale che non necessiti di modiche, quando si devono gestire nuove informazioni. Per ottenere questo risultato, il progettista deve avere ben chiaro il dominio del problema e deve pensare a situazioni potenziali che potrebbero verificarsi in futuro, prevedendo al contempo il maggior numero di eccezioni.

### 4.7 Relazioni

Sono il cuore pulsante del sistema. Senza di esse, organizzare i dati in forma normale sarebbe come avere un'automobile senza benzina. In termini pratici, una relazione intercorre tra una tabella primaria e una tabella o più tabelle secondarie. Per esempio, una fattura ha un'intestazione e delle righe; i campi che si trovano nell'intestazione formano la tabella primaria e i campi che si trovano nelle righe formano la tabella secondaria. L'unione (cioè la relazione) può essere impostata tra la "chiave primaria" della tabella principale e un campo chiamato "chiave esterna" che ha lo stesso tipo di dati e lo stesso nome della chiave primaria, presente nella tabella secondaria. La relazione è di uno a molti, cioè una testata di fattura può avere una o più righe di fattura. Una relazione deve avere una sua integrità altrimenti le operazioni sui dati darebbe esiti non affidabili. Questa "integrità referenziale" è garantita da particolari condizioni come le seguenti:

- Il campo correlato nella tabella primaria deve essere la sua chiave primaria;

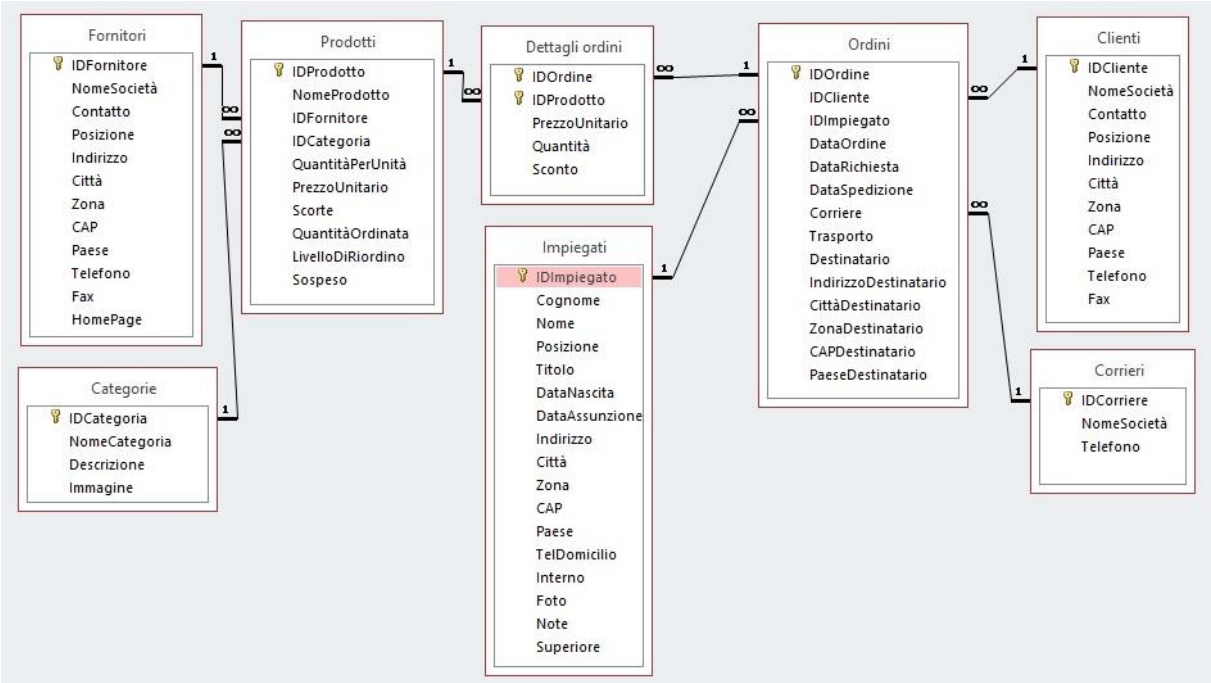
- Il campo correlato nella tabella secondaria deve chiamarsi allo stesso modo e avere lo stresso tipo di dato;
- Non si possono aggiungere righe (tuple) nella tabella secondaria che non contengano valori già previsti nella chiave primaria della tabella primaria;
- Non si possono cancellare record correlati nella tabella primaria, senza aver cancellato quelli della tabelle secondaria.

Le moderne basi dati consentono di modificare o eliminare record "a cascata" cioè modificando o eliminando un record correlato nella tabella primaria, si modifica o si eliminano i record correlati nella tabella secondaria. Soprattutto per quanto riguarda l'eliminazione bisogna essere molto prudenti, perché la cosiddetta "delete on cascade" cancella i record correlati in automatico senza la conferma dell'utente. Questa situazione deve essere ben gestita, soprattutto all'interno di software gestionali che, se non ben testati, potrebbero impartire ordini di cancellazione in situazioni impreviste. La relazione di integrità risponde anche ad un criterio di "cardinalità". cioè il numero di volte che una certa istanza di un'entità partecipa alla relazione. Possiamo avere le seguenti situazioni:

- Relazione obbligatoria una sola volta: (1,1);
- Relazione obbligatoria almeno una volta: (1,n);
- Relazione opzionale una sola volta: (0,1);
- Relazione opzionale più volte: (0,n).

Concludo questa breve disamina sulle basi dati facendo un commento finale allo schema che riporto di seguito.[27]

[27] Fig. diagramma relazioni tratto dal database "Northwind" di Microsoft.




Guardando la figura, si può vedere come le tabelle siano legate con delle relazioni uno a molti, dove il lato uno è rappresentato dal numero 1 e il lato molti dal simbolo dell'infinito (il numero 8 rovesciato). Partendo dall'alto a destra, un fornitore può fornire uno o più prodotti, mentre una categoria può riguardare uno più prodotti. Un impiegato può fare uno o più ordini e a sua volta un ordine può avere una o più righe di dettaglio dell'ordine stesso. Un prodotto può essere contenuto in una o più righe di un ordine. Un cliente può fare uno o più ordini, così come un corriere può spedire uno o più ordini.

# CAPITOLO 5 – Il data warehouse

## 5.1 Introduzione

Prima di ogni cosa è utile dire che un data warehosue viene utilizzato, di norma, in grandi organizzazioni, sia per la mole dei dati da gestire, sia per i costi di realizzazione. Tuttavia, ad oggi, sistemi alcuni sistemi per data warehouse possono essere creati anche per le medie aziende, visto che la tecnologia disponibile è avanzata nel tempo e, di conseguenza, i costi sono diminuiti. Un data warehouse è un sistema (ovvero un architettura software) che serve per analizzare fatti di gestione aziendale da diversi punti di vista, chiamati dimensioni. Possiamo sintetizzare forse grossolanamente, ma efficacemente, affermando che i fatti da analizzare sono descritti dalle dimensioni.

L'analisi multidimensionale è il cardine di questa struttura dati che proprio grazie all'incrocio e al cambiamento dei diversi punti di vista (e ad altre operazioni sugli insiemi), ci

fa conoscere un fatto-fenomeno da diverse angolazioni e ci permette di scoprirne caratteristiche inedite tali da fornire informazioni per prendere decisioni. Esiste quindi un server OLAP (Online Analytical Processing) e diversi client OLAP utilizzati dagli utenti per eseguire le loro interrogazioni.

Voglio precisare subito un concetto che ribadirò più avanti nella trattazione: il data warehouse viene realizzato in funzione dell'utente e della sua capacità di porre domande, anche impreviste, al sistema (certo entro un certo limite), diversamente da quanto accade con le applicazioni gestionali comuni dove le interrogazioni (Query) sono già impostate. L'utente del data warehouse ha quindi una libertà maggiore nel personalizzare le sue interrogazioni e operazioni da eseguire sui dati. Non solo, per interrogare i dati l'utente non deve conoscere il linguaggio SQL tipico delle basi dati relazionali.

### 5.2 Logica multidimensionale

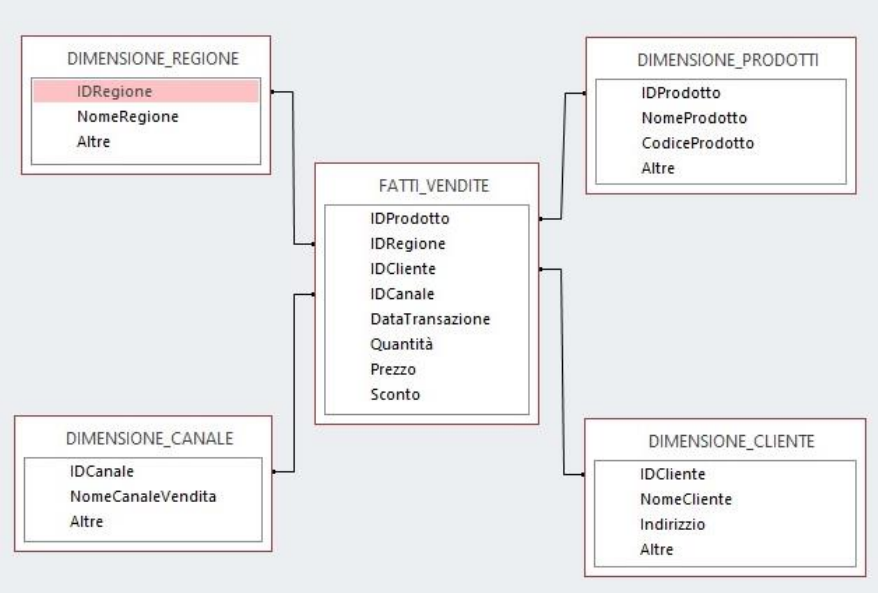
A questo punto della trattazione devo spiegare meglio il concetto di logica e analisi multidimensionale dei dati. Ho già accennato che i fatti aziendali possono essere esplorati da diversi punti di vista. Faccio un esempio: le vendite. L'importo delle vendite è un fatto, le dimensioni da cui osservarlo possono essere il luogo, il tempo, i prodotto venduto. Ancora più in dettaglio, il fatto potrebbe essere una riga di scontrino o di fattura e le dimensioni di analisi potrebbero essere le seguenti:

- Data dell'operazione;
- IDProdotto (dimensione prodotto);
- IDNegozio (dimensione negozio-punto vednita)
- IDPromozione (dimensione attività promozionale);
- Numero transazione POS;
- Quantità venduta;
- Ammontare complessivo della vendita;
- Margine di ricarico complessivo.

I campi che cominciano con "ID" si intendono come chiavi esterne alle rispettive tabelle delle dimensioni.[28] Fatti e dimensioni sono gli elementi che esaminerò in dettaglio più avanti, per ora introduco un concetto importante del data warehouse: il cubo. L'analisi multidimensionale viene immaginata come un cubo, ovvero, spesso come un insieme di cubi,

[28] Non si tratta di una nomenclatura obbligatoria. Gli stessi campi, nella pratica, sono anche nominati con il suffisso "Cod" (Codice). Esempio: CodProdotto, CodNegozio, oppure si possono usare altre convenzioni. Ci sono poi le chiavi surrogate che sostituiscono i codici operazionali come verrà detto più avanti.

definiti “ipercubi”. I cubi sono formati da una tabella principale, chiamata tabella dei fatti e tabelle collegate chiamate appunto tabelle delle dimensioni. La struttura, nel suo insieme richiama una stella, da qui il nome “star schema”. Un schema (anche se semplificato) può essere ricavato dalla seguente figura.[29]




### 5.3 Data warehouse e data mart, differenze

Per data warehosue si intende il sistema completo aziendale di analisi multidimensionale. Si parla invece di data mart quando ci si riferisce ad una applicazione specifica di data warehosue, per esempio per le vendite, la contabilità, la produzione e così via. Tuttavia, bisogna evitare l’errore di pensare che il data mart sia ristretto per forza all’attività di specifici reparti e uffici. Infatti, i dati di un settore aziendale possono anche venire da altri settori e far parte della visione a processi dell’azienda di cui ho già parlato nel capitolo dedicato. Per realizzare una data mart\ data warehisue bisogna avere ben chiaro lo scopo a cui serve una simile architettura, anche perché spesso per realizzarla occorrono tempo e denaro.

### 5.4 La caratteristiche dal data warehouse

Il data warehouse deve avere alcune caratteristiche:

- Deve essere utile per migliorare le decisioni;
- Rendere le informazioni facilmente accessibili;
- Presentare le informazione in modo coerente, cioè adatto allo scopo;
- Deve essere ad adattabile al variare delle esigenze;

[29] Fig. immagine di esempio realizzata da Andrea Mecchia.

- Deve essere sicuro;
- Deve avere prestazioni di accesso ai dati adeguate.

Per realizzarlo correttamente occorre evitare alcuni errori che posso definire come "classici":

- Badare troppo alla tecnologia e non agli scopi;
- Non farlo capire e apprezzare agli utenti che dovranno usarlo;
- Partire con progetti troppo lunghi e complessi.

Come accennato sopra, il data warehouse si basa su due concetti fondamentali:

- I fatti;
- Le dimensioni.

A queste due possiamo aggiungere le misure cioè la misurazione numerico-quantitative dei fatti. Per esempio, una tabella dei fatti delle vendite (molto semplificata) che registra transazioni di vendita in diversi negozi, presenti in differenti città potrebbe essere composta dai seguenti campi:

- IDNegozio;
- IDCittà;
- IDRegione;
- IDProdotto;
- IDProduttore;
- Data di vendita;
- Importo;
- Quantità.

Importo e quantità sono le misure. Il data warehosue viene alimentato con dati di gestione aziendale provenienti da varie fonti e che si presentano con vari formati. I dati possono essere:

- Dati interni aziendali;
- Dati esterni e di mercato.

Questi dati vengono definiti "operazionali" e vengono immagazzinati, "puliti" e resi disponibili per le interrogazioni da parte degli utenti alla ricerca di risposte.

### 5.5 Importazione dati e area di staging

I dati operazionali vengono immagazzinati temporaneamente in un'area di staging per essere trasformati ed essere usati. Quest'area di staging dei dati è importantissima trattandosi

delle porte di ingresso al sistema; si colloca tra i sistemi sorgenti dei dati e l'area di presentazione di cui parlerò più avanti.

In quest'area i dati subiscono le seguenti operazioni:

- Importazione;
- Trasformazione.

Gli utenti finali non possono accedere a quest'area dove invece vengono svolte operazioni di pulizia dei dati, quali per esempio:

- Correzione di errori ortografici;
- Gestione di omissioni;
- Adattamenti del formato;
- Altre operazioni.

Spesso le stringhe presentano spazi all'inizio o alla fine, i numeri a volte sono arrotondati in modi diversi, si deve scegliere invece un solo tipo di arrotondamento. Un altro esempio sono gli indirizzi; come scrivere la parola piazza? Potremmo trovare delle stringhe che riportano "p.zza", "Piazza", oppure se pensiamo al numero civico, esso può trovarsi separato con la virgola: "Via …., xx", oppure preceduto dal numero: "Via …n.xx) e altre situazioni. Un altro esempio può essere fornito dalle operazioni sulle date, quando vogliamo trasformare il formato di una data in formato italiano gg/mm/aaaa, in formato americano, yyyy/mm/dd o altre operazioni simili. Un altro esempio ancora può essere dato dalla sostituzione di codici prodotto con descrizioni più comprensibili.

Le operazioni di alimentazione e pulizia dei dati vengono definite ETL (Extract Transform Load). Spesso, come già detto, i dati da ripulire sono contenuti in file "piatti" e non in data base relazionali. Per file piatti intendo per esempio i seguenti tipi di file:

- Txt;
- Csv;
- Excel;
- Word;
- Xml;
- Html;
- Json;
- Altri possibili formati.

Tuttavia può accadere che vengano create delle strutture relazionali per organizzare l'area di staging. L'esperienza ha sconsigliato questo approccio perché troppo lento e complesso.

Infatti, utilizzare strutture relazionali significherebbe, di fatto, caricare i dati due volte: una nell'area di staging, l'altra nel data warehouse vero e proprio. Come è facile immaginare, il dispendio di risorse potrebbe non giustificare questa scelta. Tuttavia, la questione non è affatto di semplice soluzione. Leggere un file piatto, riga per riga, implica necessariamente dover identificare la sua struttura, per poter ben individuare i dati e il loro significato. Si ricorre quindi spesso a tabelle esterne che ci permettono di organizzare preliminarmente i dati in formato più strutturato che poi faciliterà la lettura dei dati stessi.

L'area di staging rende disponibile i dati secondo delle regole aziendali prestabilite che formano oggetto di specifica condivisione con il management aziendale. L'area di staging dei dati viene caricata e a fine operazione gli utilizzatori finali dovranno essere avvertiti del fatto che i dati sono normalizzati e disponibili. In seguito parlerò dell'importanza di avere uno sponsor del progetto e di assicurarsi la collaborazione degli utenti finali e di altri soggetti. Proprio le definizioni dei dati e il loro formato sono il riflesso degli obiettivi e delle scelte progettuali. I dati nell'area di staging servono per il popolamento della tabella dei fatti ma anche di quelle dimensionali. Per queste ultime è importante mantenere una tabella con i riferimenti alle chiavi surrogate di cui dirò più avanti nel testo. Per quanto riguarda il popolamento delle tabelle delle dimensioni, l'area di staging dei dati permette queste tre operazioni:

- Estrazione dati;
- Pulizia dei dati;
- Assegnazione e mantenimento delle chiavi surrogate.

Per quanto riguarda il popolamento della tabelle dei fatti possiamo avere i seguenti passaggi che riguardano l'area di staging dei dati:

- Estrazione dati;
- Separazione dei fatti al livello di granularità necessario;
- Trasformazione dei dati;
- Inserimento delle chiavi surrogate;
- Costruzione o aggiornamento delle tabelle di aggregazione dei fatti;[30]
- Caricamento dei dati;
- Avviso agli utenti dei nuovi dati caricati.

[30] Sono tabelle aggregate, realizzate di norma fuori dalla tabella dei fatti per facilitare l'aggregazione o la somma dei fatti al fine di rendere migliori le prestazioni del data warehosue.

### 5.6 Metadati

Un altro elemento essenziale per il corretto funzionamento dei data warehouse sono i metadati. Possono essere definiti come dati che descrivono altri dati. Un esempio tipico è la scheda di un catalogo di una biblioteca che riporta diverse informazioni che riguardano un libro. Informazioni quali, per esempio:

- Titolo;
- Autore;
- Genere;
- Editore;
- Edizione;
- Numero di volumi disponibili;
- Posizione dei volumi nella biblioteca.

Sono quindi dei dati usati per classificare altri dati operazionali in categorie e facilitarne l'uso. Per esempio, possono essere utilizzati per l'accesso ai dati importati e normalizzati, come i nomi e definizioni aziendali, con cui vengono identificati le colonne dell'area di presentazione.

### 5.7 Presentazione dei dati

L'area di presentazione è lo spazio adibito all'interrogazione diretta dei dati che non avviene usando il linguaggio SQL come per un data base normalizzato. La semplicità d'uso e il diverso scopo del data warehouse lo rendono molto diverso. La modellazione delle base dati relazionali cerca ti togliere la ridondanza dei dati che invece qui torna utile agli scopi conoscitivi.

Bisogna considerare che un data warehouse è composto da milioni, se non miliardi di record. La normalizzazione costringerebbe a percorsi di interrogazione dei dati che potrebbero essere troppo impegnativi in termini di risorse e tempi di risposta, tali da rendere perfettamente inutile l'intero sistema. Quando popolare il data warehouse? Non esiste una regola precisa, il popolamento può essere giornaliero, settimanale, quindicinale, mensile tutto dipende dai seguenti fattori concomitanti:

- Esigenze degli utenti;
- Disponibilità dei dati;
- Tecnologia utilizzata.

Le operazioni che l'utente deve poter fare sono le seguenti:

- Raggruppare i dati (roll up);

- Disaggregare i dati (drill down);
- Tagliare i dati (slice & dice);
- Ri-orientare il cubo (pivot).

Nel primo caso si raggruppa ad un livello superiore (per es. passare da giorni a settimane, da settimane a mesi, oppure sommare valori per gruppi). Nel secondo caso fare il contrario e ritornare al dato più atomico disponibile. Nel terzo caso occorre selezionare solo una parte dei dati in base ad un criterio (per es. tutte gli acquisti in un certo punto vendita) Nell'ultimo caso sarà possibile effettuare un cambiamento globale di punto di vista (per es. da fornitori per materiale a materiale per fornitori). Tutte queste operazioni devono poter esser svolte da utenti finali che, appunto, non hanno cognizioni di basi dati e conoscenza di SQL.

### 5.8 I fatti

Il cuore dell'applicazione è la tabella dei fatti. In questa tabella vengono registrati i fatti nel modo più dettagliato (atomico) possibile. Come già accennato sopra possiamo riassumere e definire il fatto come una misura aziendale. Le righe che rappresentano i fatti, devono riportare dati che non possono essere più scomponibili, questo perché nelle interrogazioni i fatti (e le misure) vengono aggregati. Sui fatti si possono effettuare operazioni come somma, media, conteggio. Il livello di dettaglio dei fatti viene definito granularità o grana del data warehouse. Quest'ultimo aspetto è essenziale per il corretto funzionamento del sistema. Esistono tre tipi di granularità nelle tabelle dei fatti:

- Transazioni;
- Istantanee periodiche;
- Istantanee accumulate.

La granularità a transazione risulta essere quella più utilizzata. Si tratta, come è facile immaginare, di una riga per ogni transazione, per esempio ogni riga in uno scontrino. La seconda granularità fotografa un certo valore in un dato momento: si pensi ai livelli di inventario giornaliero. La terza consiste nell'aggiornamento di righe nella tabella dei fatti, quando i dati cambiano; in genere si tratta di fatti che rappresentano un ciclo di lavorazione consecutivo a stati e, di norma, sono presenti più campi che rappresentano le date in cui avvengono questi passaggi. Come è facile intuire, la scelta della granularità è vitale per gli scopi conoscitivi che ci si propone e può portare al successo o al fallimento del progetto. Non solo, ma come spiegherò in seguito, la granularità è uno degli elementi di cui tenere conto per creare dimensioni comuni a più data mart, quando si cerca di creare un unico data warehouse

globale aziendale. La tabella dei fatti è composta da chiavi esterne alle tabelle delle dimensioni. I fatti possono essere:

- Aggiuntivi;
- Semi aggiuntivi.

I primi possono essere sommati lungo tutte le dimensioni, i secondi solo lungo alcune. Per un'analisi utile i fatti di maggior interesse devono essere:

- Numerici;
- Aggiuntivi.

Un esempio di fatti non aggiuntivi possono essere indici e rapporti. In questo caso, nella tabella dei fatti devono essere memorizzati il numeratore e il denominatore. Anche il costo unitario o il prezzo unitario sono fatti non aggiuntivi, perché a pensarci bene, non si possono sommare diversi valori unitari.

Infatti, come ho detto in precedenza, tra le operazioni che il data warehouse deve consentire senza problemi ci sono:

- Raggruppare i dati (roll up);
- Disaggregare i dati (drill down);
- Tagliare i dati (slice & dice);
- Ri-orientare il cubo (pivot);
- E le relative operazioni matematiche sui sottoinsiemi.

A livello progettuale, prima della scelta delle dimensioni, vanno individuati i fatti. La tabella dei fatti è normalizzata e ha moltissime righe ma relativamente poche colonne, la maggior parte delle quali sono chiavi esterne alle tabelle delle dimensioni. La tabella dei fatti deve essere popolata da fatti utili e reali, perciò vanno evitati record che contengono zeri che indicano che non sono disponibili valori. Questa presenza di dati inutili aumenterebbe senza senso il numero di righe della tabella dei fatti a discapito delle prestazioni e della chiarezza. La tabella dei fatti ha una sua chiave primaria che in genere è composta da un sottoinsieme di chiavi esterne alle tabelle di dimensione. Alcuni potrebbero chiedersi se sia utile prevedere un ID progressivo che funga anche da chiave. La risposta è negativa a meno che non venga prevista la possibilità di caricare righe identiche, tuttavia si tratta di una scelta che a mio avviso va fatta con molta cautela perché, come accade nelle basi dati tradizionali, non scegliere degli attributi che descrivano il record, nasconde spesso qualche problema di progettazione. Un esempio (non esaustivo) di campi per alcune tabelle dei fatti possono essere le seguenti.

**Fattura**

- Data della fattura;
- Codice del prodotto;
- Codice del cliente;
- Data spedizione merce;
- Codice Tipo di spedizione della merce;
- Codice Tipo di offerta promozionale;
- Ammontare del prezzo;
- Ammontare dello sconto;
- E altre.

**Contabilità**

- IDPeriodo contabile;
- IDCausale contabile;
- IDConto;
- IDSede Azienda;
- Ammontare DARE del periodo;
- Ammontare AVERE del periodo.

Per quanto riguarda i codici operazionali farò delle precisazioni quando parlerò delle chiavi surrogate, perché i codici (e gli identificativi) sono chiavi esterne alle tabelle delle dimensioni.

### 5.9 Le dimensioni

Tramite le dimensioni noi classifichiamo, quindi esploriamo i fatti. Le tabelle delle dimensioni, al contrario di quella dei fatti, hanno relativamente poche righe ma molte colonne (attributi). Le colonne sono i punti di vista attraverso i quali guardare ai fatti. Numero e tipo di colonne devono essere coerenti con la granularità della tabella dei fatti e i fatti stessi, appunto, devono essere preferibilmente aggiuntivi su tutte le dimensioni. In un data mart, una tabella di dimensioni può avere anche più di 20 colonne e le tabelle di dimensioni possono essere anche più di 10. I nomi degli attributi devono essere:

- Adeguati alla situazione reale;
- Composti da parole estese e non da abbreviazioni.

Le tabelle delle dimensioni devono avere una loro chiave primaria per creare un vincolo di integrità referenziale con la tabella dei fatti. Per questo motivo, come visto sopra, il data warehouse deve essere immaginato come una sorta di stella (star schema).

Ribadisco ancora un punto che ritengo fondante per la comprensione dell'argomento e per la riuscita di un progetto: il data warehouse è costruito per l'utente e il progettista (che è un tecnico) non deve dimenticarselo mai. Il data warehouse è un cubo in movimento, dove poter far "ruotare" gli attributi dimensionali e fare operazioni sulle misure dei fatti per ricavare informazioni in modo semplice senza costruire istruzioni SQL. Non solo, ma come già detto, l'utente non si avvale di Query predefinite disponibili nel gestionale operazionale ma può (ovviamente entro certi limiti) sperimentare una vasta gamma di interrogazioni da costruire sul momento, senza ricorrere appunto a conoscenze informatiche specialistiche. Questo principio è uno dei test logici del corretto funzionamento del sistema, che a questo punto della trattazione dovrebbe essere più chiaro. Facendo un esempio, una tabella di dimensione che rappresenti un conto di contabilità in un data mart della contabilità generale potrebbe avere i seguenti attributi:

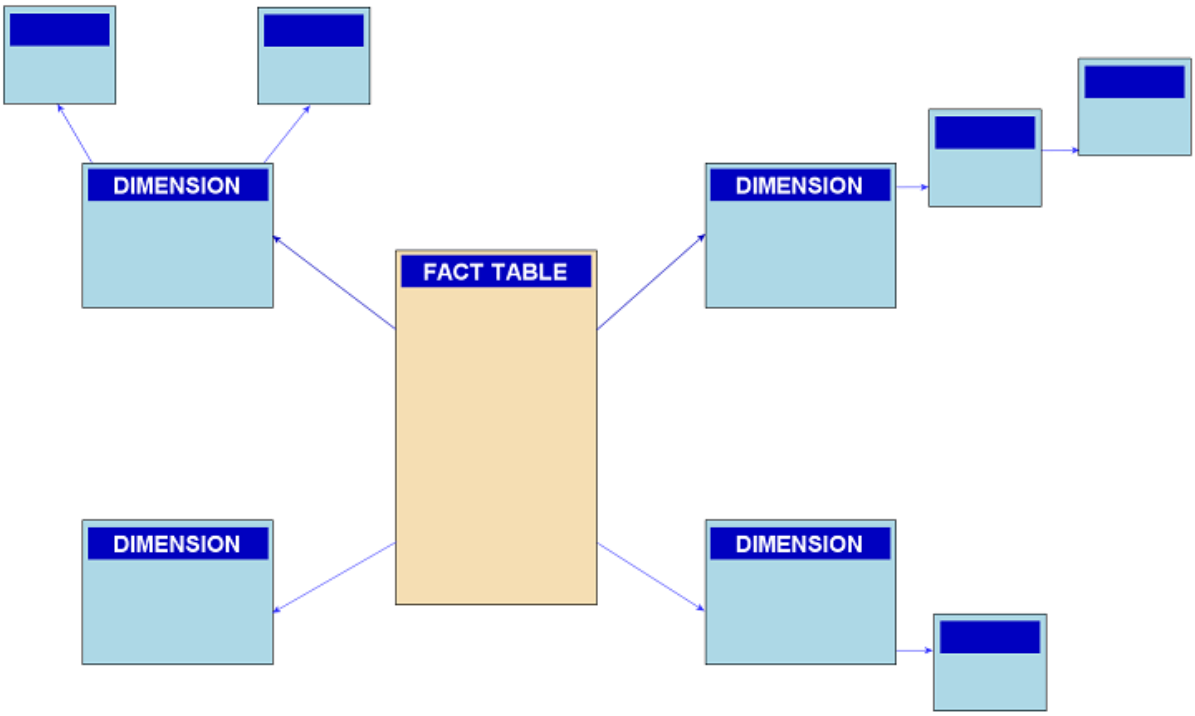
- IDConto;
- Descrizione conto;
- Tipologia conto;
- Categoria conto;
- Livello del piano dei conti.

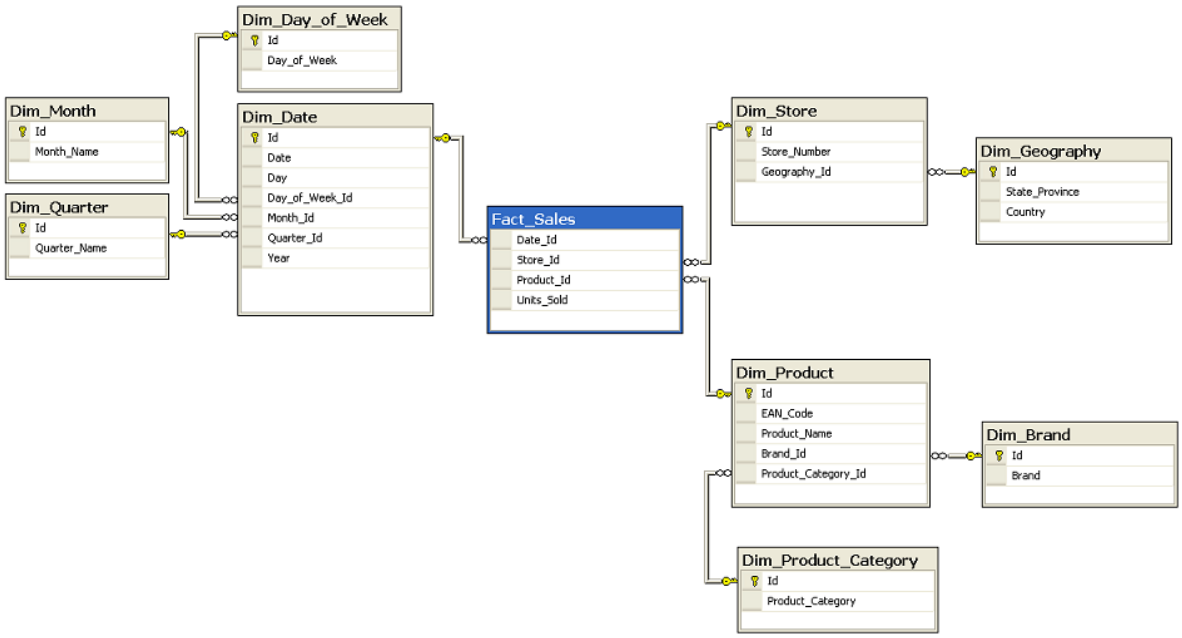
Ad esempio, la tipologia potrebbe assumere i seguenti valori:

- Costo di esercizio;
- Ricavo di esercizio;
- Attività patrimoniale;
- Passività patrimoniale.

La categoria potrebbe essere anche una classificazione più specifica all'interno della tipologia come per esempio le immobilizzazioni materiali, immateriali e altri. Le tabelle dei fatti sono normalizzate, quelle dimensioni non lo sono. Tuttavia il classico schema a stella che vede la tabella dei fatti al centro, collegata con chiavi esterne alle tabelle delle dimensioni può avere anche la variante del cosiddetto "schema a fiocco di neve" (snowflake schema), sul quale vale la pena spendere qualche considerazione. Quest'ultimo schema consiste nel normalizzare le tabelle delle dimensioni, scomponendole in almeno due tabelle per ogni dimensione. Le figure riportate di seguito possono aiutare a chiarire la comprensione.[31] [32]

[31] Fig. star schema tratta dall'indirizzo: https://upload.wikimedia.org/wikipedia/commons/b/b2/Snowflake-schema.png.
[32] Fig. Snow Flake Schema tratta dall'indirizzo: https://commons.wikimedia.org/wiki/File:Snowflake-schema-example.png.




L’esigenza di questa struttura potrebbe favorire la manutenzione delle dimensioni, dove la mancata normalizzazione delle relative tabelle, costringerebbe a modificare moltissimi valori. La normalizzazione, basata su relazioni tra tabelle, invece, risolverebbe egregiamente questo problema, poiché modificando il valore in una sola tabella collegata, la modifica stessa si propagherebbe a tutte le istanze di quel valore presenti nella tabella delle dimensioni. Anche

qui però sorge il solto problema che affligge sempre i progettisti di basi di dati e di data warehosue: trovare un equilibrio tra prestazioni e complessità di navigazione. Ricordo infatti che la navigazione tra i dati di un data warehouse viene fatta da utenti che non sono obbligati a conoscere la logica relazionale. In genere, disegnare uno snowflake schema rende necessario creare molte tabelle per normalizzare le dimensioni. Facendo un ultimo esempio con la dimensione di prodotto, essa potrebbe accogliere le seguenti informazioni:

- IDProdotto;
- Descrizione;
- IDMarca;
- IDReparto;
- IDProduttore;
- Tipo di imballaggio;
- Tipo di conservazione in magazzino;
- Altre.

Come si vede anche dalle immagini precedenti, in uno schema a fiocco di neve normalizzato, gli attributi diventano a loro volta delle chiavi esterne verso tabelle legate, magari a loro volta, con altre tabelle in forma normale. La marca potrebbe essere una dimensione a sé, collegata a sua volta con una dimensione categoria, collegata ancora con una dimensione di reparto. In ultima analisi, come visto parlando delle basi dati relazionali, normalizzare significa fondamentalmente due cose:

- Scomporre;
- Collegare.

Al di là dello snowflake schema, della minore o maggiore normalizzazione, in genere, quante dimensioni servono per creare un data warehouse utile? Quando ci accorgiamo che stiamo procedendo in modo giusto? Dipende dagli scopi, ma generalmente una quindicina di dimensioni sono sufficienti per attraversare i processi aziendali principali. Tuttavia, non è la numerosità in sé un indice di buona progettazione; occorre esaminare la cosa dal punto di vista dell'indipendenza delle relazioni, o se si preferisce, della loro autoconsistenza. In fondo stiamo parlando di tabelle e proprio come visto per le basi dati relazionali, il dominio del problema è quello che va trovato prima di individuare le tabelle utili e mentre si stanno progettando i loro attributi. Quando un progettista si ritrova con un numero di dimensioni che sembrano eccessive, questo è la spia che le dimensioni create non siano completamente indipendenti le une dalle altre. In questo caso si devono mettere insieme più dimensioni in una sola. Per capire se la numerosità è eccessiva bisogna chiedersi se dimensioni apparentemente

considerate come separate tra loro, non stiano formando in realtà una sorta di dipendenza gerarchia. Questo è un altro test logico di progettazione molto importante che deve essere sempre fatto in occasione del design logico del sistema, di cui parlerò più avanti. I contesti aziendali in cui ci sono gerarchie possono essere per esempio:

- Il piano dei conti;
- Le righe di un ordine;
- Le righe di una fattura;
- Un organigramma aziendale.

Se prendiamo per esempio una fattura, la tabella dei fatti potrà avere una granularità a livello di riga di fattura e delle dimensioni separate di:

- Prodotto;
- Offerta promozionale;
- Spedizione.

Un altro aspetto che riguarda le dimensioni sono le cosiddette "chiavi surrogate", le quali meritano un'attenzione a parte. Si tratta di chiavi fittizie (in genere numeri sequenziali), che vengono assegnate per popolare una dimensione. A volte questa soluzione viene preferita alla scelta di altre chiavi per creare le relazioni tra la tabella dei fatti e quella delle dimensioni. Una regola per la modellazione del data warehouse afferma che *"ogni join tra tabelle di dimensioni e di fatti nel data warehouse deve essere basato su chiavi surrogate a numero intero senza significato ed è necessario evitare l'uso di codici di produzione operazionali naturali. Nessuna delle chiavi del data warehouse deve essere intelligente e consentire all'utente di dedurre informazioni sulla riga semplicemente guardando la chiave"*. [33]La scelta delle chiavi surrogate è quella che si rivela vincente in termini di scalabilità dell'architettura. Le chiavi surrogate hanno inoltre il seguente vantaggio: *"proteggono l'ambiente del data warehouse dalle modifiche operazionali. Le chiavi surrogate consentono al team del data warehouse di mantenere il controllo dell'ambiente senza impazzire sulle regole operazionali per generare, aggiornare, eliminare, riciclare e riutilizzare i codici di produzione."*[34]

I vantaggi sono anche di altro tipo infatti *"L'utilizzo delle chiavi surrogate presenta anche vantaggi per le prestazioni. La chiave surrogata è un numero intero più piccolo possibile che garantisce però di ospitare comodamente la cardinalità futura o il numero massimo di righe della dimensione"*. [35]Anche le dimensioni giocano a favore della chiave surrogata in quanto

[33] Ralph Kimball Margy Ross, Data Warehose La guida completa, HOEPLI, Milano, 2003 pagina 59.
[34] Op. cit. pagina 59.
[35] Op cit. Ralph Kimball Margy Ross, Data Warehose La guida completa, HOEPLI, Milano, 2003 pagina 60.

*"Una chiave surrogata più piccola si traduce in tabelle di fatti più piccole e più righe della tabella di fatti."*[36]

## CAPITOLO 6 - Note operative di progettazione

### 6.1 Modifiche al sistema e scalabilità

In precedenza ho parlato della differenza tra data mart e data warehosue. Un progetto può partire da un data mart per soddisfare specifiche esigenze e poi evolvere in un sistema informativo che coinvolge tutta l'organizzazione. Si possono aggiungere attributi alla tabelle dei fatti e si possono aggiungere dimensioni al data mart, a patto di non modificare quanto già creato. Solo in questo modo ci si può accorgere di quali e quanti dati condividere e scambiare tra i vari data mart. Il sistema dei dati dimensionale deve essere perfettamente "scalabile", cioè modificabile senza impatto sul pregresso; per riuscire in questo intento è necessaria una buona progettazione iniziale. Nel prossimo paragrafo spiegherò più in dettaglio l'intero ciclo di vita del data warehosue, mentre di seguito parlerò di dimensioni e fatti comuni a più data mart, ovvero di dimensioni e fatti "conformati".

Quando si progetta un data mart bisogna sempre chiedersi come esso potrà essere inserito all'interno di un data warehouse per tutta l'organizzazione. Anzitutto, la tabella dei fatti del data mart deve avere la giusta granularità, che deve essere coerente con quella degli altri data mart aziendali (anche se ancora non esistenti e solo potenziali), proprio per poter scambiare informazioni in futuro. Individuare dimensione comuni, dunque è la prima strada da percorrere quando ci si dedica alla progettazione di un data mart. Viene d'aiuto al progettista il concetto di data mart consolidato. Come già accennato in precedenza, non si deve pensare ad un data mart come un'architettura software speculare ad ogni funzione aziendale, bensì come qualcosa che si occupa di processi aziendali che possono attraversare diverse funzioni previste nell'organigramma. Resta inteso comunque che è alquanto difficile, pensare a dimensioni comuni immutabili nel tempo, già dalla creazione del primo data mart. Tuttavia è utile quanto meno fare un primo tentativo in questo senso, perciò il progettista si occuperà, in prima battuta, di creare dimensioni e fatti standardizzati. Si prendano per esempio i seguenti tre data mart:

- Acquisti;
- Inventario;

[36] Op cit. pagina 60.

- Vendite.

Sicuramente questi tre processi aziendali avranno informazioni in comune che possono trasformarsi in dimensioni condivise. A titolo esemplificativo potrebbero essere:

- Data;
- Prodotto;
- Negozio;
- Magazzino;
- Produttore;
- Contratto;
- Spedizioniere.

Per individuare correttamente le dimensioni comuni il progettista può avvalersi di una matrice denominata "bus di dati" come quella di seguito indicata in figura.[37]

| | DIMENSIONI COMUNI | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **Data** | **Prodotto** | **Negozio** | **Promozione** | **Magazzino** | **Produttore** | **Contratto** | **Spedizioniere** |
| **PROCESSI AZIENDALI** | x | x | x | | | | | |
| Vendite al dettaglio | x | x | x | | | | | |
| Inventario al dettaglio | x | x | x | | | | | |
| Consegne al dettaglio | x | x | | | | | | |
| Inventario al magazzino | x | x | | | x | x | | |
| Consegne di magazzino | x | x | | | x | x | | |
| Ordini di acquisto | x | x | | | x | x | x | x |

Questa matrice può aiutare ad individuare le dimensioni comuni e la loro priorità rispetto al progetto. Un altro accorgimento per disegnare bene dimensioni comuni è la necessità di dimensioni e fatti conformati. Le dimensioni conformate sono quelle denominate allo stesso modo, che contengono gli stessi tipi di dati e contengono quindi gli stessi valori, di conseguenza hanno lo stesso valore semantico in ogni tabella dei fatti nei diversi data mart in cui sono contenute. Le dimensioni conformate devono essere pensate al più basso livello della granularità dei dati e devono essere approvate e validate dal livello più alto del management, proprio in virtù del loro ruolo che avranno in futuro per altri data mart all'interno dell'organizzazione. Come già visto parlando per i sistemi ERP, anche in questo ambito le dimensione (e i fatti) conformati rappresentano un primo nucleo di quello che potrei definire come "intelligence business model" che costituisce l'altra faccia della medaglia del "business model" operazionale. Un passo molto importante che influenzerà una strutturazione futura.

I fatti conformati devono essere definiti con accortezza perché essi possono essere espressi in un'unità di misura diverse a seconda delle tabelle dei fatti dei diversi data mart in cui sono inclusi.

[37] Op. cit. Ralph Kimball Margy Ross, Data Warehose La guida completa, HOEPLI, Milano, 2003 pagina 80.

Misure dei fatti possono essere grandezze quali:

- Reddito;
- Margine;
- Profitto;
- Costi standard.
- e altri.

I fatti per essere ben conformati devono essere:

- Chiamati allo stesso modo;
- Espressi nella medesima unità di misura.

### 6.2 Progettazione e criticità

Il problema delle dimensioni e dei fatti conformati è solo uno degli aspetti operativi più importante. Approfondirò il ciclo di vita di un progetto di data warehouse, perché le criticità sono varie. Come è intuibile da quanto esposto finora, la progettazione e la realizzazione anche di un solo data mart è operazione molto delicata che deve essere gestita secondo linee guida rigorose, per non commettere errori che potrebbero vanificare sforzi che richiedono tempo e denaro. Ogni progetto deve avere il giusto sponsor all'interno dell'organizzazione nella quale dovrà essere usato di data warehosue. Lo sponsor è la persona o le persone in posizioni apicali che hanno compreso lo scopo, l'utilità reale e il beneficio che l'organizzazione aziendale avrà dal progetto. Prima di tutto quindi è vitale individuare e assicurarsi l'appoggio di chi può far riuscire o far fallire l'iniziativa.

Non va trascurata la piena collaborazione di chi dovrà usare il software, che in genere, è il middle e top management. Successivamente bisogna assicurarsi anche la completa collaborazione del personale IT interno e degli esperti interni della materia oggetto del data warehouse.

Più in dettaglio è utile seguire i seguenti passi:

- Assicurarsi uno sponsor aziendale forte;
- Individuare un bisogno specifico ed urgente da soddisfare con la realizzazione di un data mart\data warehosue;
- Capire subito la fattibilità tecnica ed economica del progetto;
- Scegliere le tecnologie, il personale e i fornitori in grado di realizzare il tutto.

Il progetto di realizzazione può suddividersi nelle seguenti macro-fasi logiche che devono essere formalizzate e documentate:

- Individuazione delle motivazioni e dello scopo;

- Pianificazione del progetto;
- Modellazione dimensionale;
- Design fisico;
- Progettazione dell'area di staging dei dati;
- Distribuzione;
- Miglioramento ed espansione.

Queste macro fasi devono essere arricchite in dettaglio da:

- Scelta della tecnologia;
- Modalità pratica di integrazione tra varie tecnologie.

In concreto bisognerà sviluppare software soprattutto per la parte che riguarda gli strumenti ETL e la presentazione dei dati, e\o personalizzare (customizzare) software già esistente. Per quanto la presentazione, i dati dovranno essere fruibili su diversi dispositivi e accessibili anche fuori dall'organizzazione. La questione degli accessi e la sicurezza diventa quindi fondamentale. Dovranno essere individuati gli amministratori per creare utenti, permessi e autorizzazioni. Come ogni progetto informatico di una certa importanza devono essere affrontate le seguenti fasi operative:

- Sviluppo;
- Test;
- Collaudo;
- Messa in produzione;
- Documentazione;
- Formazione agli utenti finali.

Ogni fase ha un suo "ambiente" caratterizzato da:

- Software installato;
- Dati disponibili;
- Persone autorizzate ad accedere ed operare.

Di norma, le persone che effettuano il test dovrebbero essere diverse da quelle che sviluppano. Il collaudo deve essere fatto dagli utenti finali.

La messa in produzione deve essere validata dal responsabile aziendale del progetto, dopo verbale di approvazione di buon esito del collaudo. Le fasi sopra indicate non riguardano solo il progetto nella sua totalità ma anche le singole parti. Infatti, durante la realizzazione, in genere, vengono effettuate dei rilasci intermedi. Successivamente, quando ci sono dei malfunzionamenti in produzione, essi devono essere identificati e isolati. La questione viene

riesaminata in fase di sviluppo e l'errore viene corretto. Dopo la correzione si procede di nuovo a test, collaudo e messa in produzione delle funzionalità interessate dall'errore. Un'importante distinzione concettuale che, a mio avviso, deve essere fatta è tra design logico (o modellazione concettuale) e design fisico. La modellazione concettuale è una progettazione che riguarda il cosa, il design fisico riguarda il come. Nella modellazione concettuale si affrontano problematiche come le seguenti:

- Scegliere fatti e dimensioni;
- Scegliere la granularità;
- Determinare i valori validi;
- Verificare la disponibilità degli attributi.

Prima di passare alla fase del design fisico, quello concettuale deve essere validato ufficialmente dal committente. Avuto l'avallo di quanto fatto, si può passare alla fase del design fisico. In questa fase si possono mettere a punto dettagli come il nome fisico delle colonne, i tipi di dato, le chiavi da utilizzare, la gestione di campi non valorizzati. Nel design fisico si mettono a punto anche dettagli concernenti le prestazioni, il layout dei file e delle eventuali tabelle di aggregazione. Queste ultime sono tabelle che favoriscono le prestazioni di roll-up perché contengono valori pre-calcolati e pre-memorizzati, al fine velocizzare le prestazioni.

### 6.3 Tecnologie disponibili sul mercato

Le tecnologie presenti sul mercato sono molte e una loro descrizione puntuale richiederebbe una tesina come questa (o anche di più). Mi limiterò quindi ad alcune considerazioni generali e a fornire dei rimandi.

Ci sono molti produttori che negli anni hanno sviluppato software che gestiscono tutto o parte delle attività legate al data warehousing. Tra i produttori storici sicuramente c'è la SAS[38] che da anni è leader del marcato della business intelligence. Ci sono poi Oracle[39] e Microsoft[40] che hanno sviluppato loro soluzioni e sono attive in vario modo nel mercato dei data warehouse. Esiste anche un'offerta open source di vari strumenti che possono essere presi in considerazione da medie imprese oppure come tecnologie complementari in progetti più grandi. Nel panorama open source possiamo trovare il progetto italiano SpagoBI[41] che è

[38] https://www.sas.com/it_it/home.html
[39] https://www.oracle.com/application-development/technologies/warehouse/warehouse-builder.html
[40] https://www.microsoft.com/
[41] https://www.spagobi.org/

una suite di business intelligence che offre numerose funzionalità come quelle indicate di seguito:

- Analisi multidimensionale;
- Etl;
- KPI (gestione indicatori aziendali);
- Grafici e presentazioni;
- Strumenti di data mining;
- Strumenti di collaborazione per gruppi di lavoro;
- Integrazione con Microsoft Office;
- Gestione di tabelle del db.

Il software è rilasciato con licenza Mozilla Public Licence v. 2.0.[42] .

Un altro software è Penthao. L'azienda offre tutta una serie di strumenti open source, come Pentaho Business Analytics.[43] Ci sono anche altri software gratuiti di cui una parte sono dedicati alla produzione di reportistica di presentazione e altri sono software di data warehosing più completi. I software dedicati alla reportistica sono i seguenti:

- Google Data Studio;[44]
- Microsoft Power BI Desktop;[45]
- Qlik Sense Cloud Basic;[46]
- Style Scope Agile Edition (AE);[47]
- Tableau Public;[48]

Ci sono poi software che in vario modo includono funzionalità di data warehosing e che, in alcuni casi, contengono altri strumenti di business intelligence quali:

- Metabase;[49]
- BIRT (Business Intelligence and Reporting Tools) Project;[50]
- KNIME;[51]
- Jaspersoft Community.[52]

[42] https://www.mozilla.org/en-US/MPL/2.0/
[43] https://sourceforge.net/projects/pentaho/
[44] https://datastudio.google.com/u/0/
[45] https://powerbi.microsoft.com/it-it/desktop/
[46] https://www.qlik.com/us/products/qlik-sense
[47] https://www.inetsoft.com/products/StyleScopeAE/
[48] https://public.tableau.com/en-us/s/
[49] https://www.metabase.com/start/
[50] http://download.eclipse.org/birt/downloads/drops/
[51] https://www.knime.com/knime-analytics-platform
[52] https://community.jaspersoft.com/

Gli elenchi di cui sopra non sono da intendersi tassativi. L'offerta di tecnologie simili cambia frequentemente. Oggi le tecnologie di data warehosue sono sempre più integrate con i sistemi ERP e gli altri strumenti di business intelligence come la "machine learing", ma capire il funzionamento di un data warehouse, a mio avviso, resterà la base concettuale di partenza dalla quale iniziare il percorso di comprensione dell'intero ciclo di business intelligence. Le soluzioni attuali come per esempio *"Oracle Autonomous Data Warehouse"*, permettono di creare e gestire un data warehouse in cloud, eliminando molte delle complessità che esistevano in passato per la creazione di questo tipo di infrastrutture informatiche. Nel momento in cui scrivo, sul sito dell'applicazione si legge testualmente:
*"Oracle Autonomous Data Warehouse è un data warehouse cloud che elimina praticamente tutte le complessità legate alla gestione di un data warehouse e alla protezione dei dati. Automatizza il provisioning, la configurazione, la protezione, l'ottimizzazione, il ridimensionamento, l'applicazione di patch, il backup e la riparazione del data warehouse. Diversamente dalle altre soluzioni di data warehouse cloud "completamente gestite" che correggono e aggiornano solo il servizio, Oracle Autonomous Data Warehouse offre inoltre scalabilità elastica e automatizzata, ottimizzazione delle performance, sicurezza e una vasta gamma di funzionalità integrate che consentono l'analisi basata sul machine learning, il caricamento semplice dei dati e la visualizzazione dei dati. È disponibile sia su Oracle Public Cloud che nei data center dei clienti con Oracle Cloud@Customer."*[53]

L'integrazione degli strumenti oggi è fondamentale. Microsoft per esempio propone le soluzioni di "Azure". Nel momento in cui scrivo, sulla sua pagina dedicata all' "*Architettura moderna di Data Warehouse"*[54], si possono leggere le componenti dell'architettura stessa che riporto testualmente di seguito (grassetto mio):

- ***"Azure sinapsi Analytics** è il data warehouse cloud rapido, flessibile e affidabile che ti permette di ridimensionare, calcolare e archiviare in modo elastico e indipendente, con un'architettura di elaborazione parallela massiva.*
- ***Azure Data Factory** è un servizio di integrazione dei dati ibrido che ti permette di creare, pianificare e orchestrare i tuoi flussi di lavoro ETL/ELT.*
- ***Archiviazione BLOB di Azure** è un archivio di oggetti altamente scalabile per qualsiasi tipo di dati non strutturati, ad immagini, video, audio, documenti e in modo più semplice e conveniente.*

[53] https://www.oracle.com/it/autonomous-database/autonomous-data-warehouse/
[54] https://docs.microsoft.com/it-it/azure/architecture/solution-ideas/articles/modern-data-warehouse

- ***Azure Databricks*** *è una piattaforma di analisi veloce, semplice e collaborativa basata su Apache Spark.*
- ***Azure Analysis Services*** *è un servizio di analisi di livello aziendale che ti permette di gestire, distribuire, testare e distribuire la tua soluzione di business intelligence in tutta sicurezza.*
- ***Power BI*** *è un gruppo di strumenti di Analisi business che consente di distribuire informazioni dettagliate in tutta l'organizzazione. Collegarsi a centinaia di origini dati, semplificare la preparazione dei dati ed eseguire analisi ad hoc. Produrre report interessanti, quindi pubblicarli in modo che l'organizzazione utilizzi sul Web e nei dispositivi mobili."*

Per quanto riguarda invece il connubio con i sistemi ERP, un esempio di integrazione può essere il modulo BW (Business Warehouse) di SAP R\3.[55] Si tratta di una serie di strumenti che permettono di operare direttamente sulla base dati del noto sistema ERP della SAP. Oggi la versione più recente di SAP è il sistema SAP HANA.

[55] https://www.sap.com/products/data-warehouse-cloud.html

## BIBLIOGRAFIA

Giuseppe Miceli, *Fondamenti e tecniche di Business Intelligence*, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017.

Andrea Moretto Wiel, Lorenzo Montibeller, *Big Data*, Dispensa corso Master di I Livello "Data Analyst" Unicusano, Roma, 2017.

Alberto Quagli, Paola R. Dameri, Iacopo E. Inghirami (a cura di) I SISTEMI INFORMATIVI GESTIONALI, FrancoAngeli, 2005.

Rebecca M. Riordan, *Progettare database relazionali*, MicrosoftPress-Mondadori Informatica, Milano, 2001.

Ralph Kimball Margy Ross, *Data Warehose La guida completa*, HOEPLI, Milano, 2003.