



**UNIVERSITÀ DI PISA**

Facoltà di Ingegneria

Corso di Laurea Specialistica in

Ingegneria Informatica per la Gestione d'Azienda

**Business Intelligence**

**Enterprise Commercial Tool**

**Relatori:**

Prof. Roberto Chiavaccini

Prof. Gigliola Vaglini

Dott. Alfonso Baccari

**Candidato:**

Brian Trajani



# Ringraziamenti

---

# Sommario

---

Ringraziamenti .....	3
Premessa .....	8
Introduzione.....	9
Capitolo 1 .....	11
<b>BUSINESS INTELLIGENCE</b>	
1 - La Business Intelligence nel contesto competitivo attuale.....	11
1.1 - La BI come supporto decisionale .....	14
1.2 - La BI come strumento di <i>Knowledge Discovery</i> .....	15
1.2.1 - La funzione di <i>Data Warehousing</i> della BI .....	15
1.2.2 - La funzione di Knowledge Discovery della BI.....	17
1.2.3 - Strategie aziendali <i>Knowledge Based</i> .....	20
1.2.4 - Il modello del <i>Knowledge Creation Process</i> .....	22
1.3 - La BI come fonte di vantaggio competitivo.....	25
1.4 - La BI e l'evoluzione della struttura aziendale.....	28
1.5 - La BI e la necessità di nuove competenze.....	30
1.6 - Caratteristiche di un sistema di BI .....	32
Capitolo 2 .....	34
<b>I SISTEMI DSS</b>	
2.1 - Dai processi tradizionali ai sistemi per l'analisi dei dati.....	34
2.1.1 - I punti chiave di un sistema DSS .....	36
2.1.2 - Applicazioni di un DSS.....	37
2.2 - Dal gestionale al Decision Support System .....	39
2.3 - Sviluppo di un sistema DSS .....	41
Capitolo 3 .....	47
<b>IL MODELLO MULTIDIMENSIONALE</b>	
3.1 - Elementi di un modello multidimensionale .....	47

3.1.1 - Dimensioni .....	48
3.1.2 - Gerarchie ed aggregati .....	49
3.1.3 - Fatti e misure .....	50
3.2 - Implementazione del modello multidimensionale .....	51
3.2.1 - Database Multidimensionale .....	51
3.2.2 - Lookup & Relationship & Fact .....	54
3.2.3 - I database relazionali .....	55
3.2.4 - Star Schema vs Snowflake Schema .....	57
3.3 - Operazioni multidimensionali .....	59
Capitolo 4 .....	60
DATA WAREHOUSE	
4.1 - Requisiti del Data Warehouse .....	60
4.2 - L'architettura di un Data Warehouse .....	61
4.2.1 - Il Data Mart .....	62
4.2.2 - I metadati .....	64
4.3 - Processo ETL .....	65
4.3.1 - Extract .....	65
4.3.2 - Transform .....	65
4.3.3 - Load .....	66
4.4 - L'ottimizzazione di un Data Warehouse .....	67
4.4.1 - Pre-aggregazione dei dati .....	67
4.4.2 - Indicizzazione e chiavi numeriche .....	68
4.4.3 - Partizionamento .....	68
4.5 - Dal database OLTP al Data Warehouse .....	69
4.6 - Gli strumenti di accesso ai dati .....	70
Capitolo 5 .....	72
QLIKVIEW	
5.1 - QlikView introduzione .....	72
5.2 - Caratteristiche di Qlik View .....	72
5.3 - Architettura unica .....	79
5.4 - Security in QlikView .....	83

5.5 - La piattaforma QlikView Business Discovery.....	85
5.6 - QlikView migliora il BI .....	86
5.7 - Business Discovery .....	87
5.8 - Più professionisti IT sono in grado di fornire la BI self-service .....	88
5.9 - Limiti di QlikView .....	90
5.10 - QlikView Best practices.....	91
5.11 - Sviluppo veloce e rapido time-to-value .....	94
5.11.1 - Caricamento dei dati.....	94
5.11.2 - Creazione di applicazioni con la tecnica point and click .....	95
5.11.3 - Estensione e miglioramento .....	95
5.11.4 - Visualizzazione .....	96
5.12 - Stima dimensionamento .....	97
 Capitolo 6 .....	 99
<b>IL CASO DI STUDIO: Commercial Tool</b>	
6.1 - Descrizione Funzionale .....	99
6.2 - Scopo del progetto.....	100
6.3 - Analisi dei requisiti .....	101
6.4 - Origine dei dati.....	104
6.4.1 - Modello logico dei dati e di analisi:.....	104
6.5 - Estrazione dei dati .....	106
6.6 - Nuvola dati .....	107
6.7 - Analisi dello sviluppo progettuale .....	108
6.7.1 - Stima delle attività.....	108
6.7.2 - Project management .....	110
6.7.3 - La Metodologia Agile .....	112
6.8 - Sviluppo del commercial tool .....	114
6.8.1 - Solution Analysis .....	114
6.8.2 - Modellazione del ETL1 .....	115
6.8.3 - Modellazione del ETL2 .....	118
6.8.4 - Nuvola Dati .....	120
6.9 - Presentation del commercial tool .....	122
6.9.1 - Home Page .....	124

6.9.2	- Summary Report Monthly .....	125
6.9.3	- Summary Report Yearly .....	126
6.9.4	- Year & BDG compare .....	127
6.9.5	- Order Details .....	128
6.10	- Schedulazione dei caricamenti .....	129
6.10.1	- Schedulazione dei task .....	129
Capitolo 7 .....		131
Conclusioni		
7.1	- Miglioramenti percepite dal cliente.....	131
7.2	- Esperienze e competenze acquisite .....	132
Bibliografia.....		134

# Premessa

---

Questo elaborato, o meglio il suo contenuto stilato durante il mio periodo di *stage* formativo in **ICONSULTING S.p.A.** in qualità di tesista e consulente di “**Business Intelligence**”, è nato dalla necessità di dover supportare le operazioni della forza vendita e di analisi di una società multinazionale di macchine agricole e macchine di movimento terra mediante l’appropriata tecnica, ovvero la realizzazione di un nuovo strumento commerciale.

Sfruttando la tecnologia messa a mia disposizione dall’azienda che mi ha accolto e basandomi sulla tecnologia a disposizione della società proprietaria del database, dopo una fase di software selectionsi è deciso di usare **QlikView**.

Il documento fornisce inizialmente una base teorica sulla **Business Intelligence**, sul **DecisionSupport System (DSS)**, sulla **Modellazione Multidimensionale** dei dati e sui **Data Warehouse**.

Poi fornisce una panoramica tecnica su **QlikView**, ovvero il software utilizzato per realizzare il commercialtool.

Infine, nell’ultimo capitolo si descrivono i passi che hanno portato alla realizzazione pratica del commercialtool, in particolare partendo dai requisiti da noi raccolti e dalle esigenze degli utenti chiave che abbiamo identificato, andiamo a progettare i componenti di una soluzione software utilizzando lo strumento considerato più idoneo QlikView.



# Introduzione

---

Per la necessità di dover supportare le operazioni della forza vendita e di analisi di una società multinazionale di macchine agricole e macchine di movimento terra, in questo elaborato ci troveremo di fronte a milioni di record relativi al portafoglio ordini e vendite effettuati negli ultimi 5 anni nei principali mercati europei.

Questi volumi di dati rendono chiaramente poco utili i metodi tradizionali di analisi. Tali metodi, infatti, possono creare dei rapporti informativi sui dati analizzati, ma non sono in grado di analizzarne il contenuto e focalizzare l'attenzione sulle informazioni interessanti che essi contengono.

Dalla naturale competitività del mercato attuale nasce l'esigenza di utilizzare una generazione di tecniche e strumenti capaci di assistere gli utenti in modo intelligente ed automatico nell'analizzare queste "montagne" di dati allo scopo di trovare conoscenza utile.

L'obiettivo è di costruire un sistema di supporto alle decisioni, in grado di fornire, in tempi brevi, informazioni strategiche agli utenti.

Le informazioni che successivamente vengono rese consultabile tramite un serie di dashboard, ovvero un insieme di oggetti grafici e tabellari contenenti indicatori di performance (KPI) lungo le dimensioni di business, in grado di presentare in modo intuitivo i dati trasformati, formattati ed organizzati in base alle specifiche aziendali.

Il sistema è stato sviluppato in tecnologia QlikView, uno degli strumenti di Business Intelligence attualmente più utilizzati per la realizzazione di dashboard.

L'elaborato inizia con il concetto di Business Intelligence spiegando quali sono gli elementi fondamentali e i principi.

Nel capitolo 2 si inizia a far conoscere il DSS spiegando quali sono gli elementi che lo compongono e il ciclo di vita.

Nel capitolo 3 si introduce la modellazione multidimensionale, le caratteristiche tecniche, le operazioni di analisi che si possono effettuare sul modello e i vari tipi di architetture OLAP (On Line Analytical Processing).

Nel capitolo 4 si parlerà dei Data Warehouse, del ETL (Extract, Transform and Load), dell'OLTP (On Line Transaction Processing) e degli strumenti di accesso ai dati.

Dopo aver affrontato la sezione teorica composta da questi primi capitoli, nel capitolo 5 si fornisce una sorta di manuale tecnico dedicato alle specifiche del software utilizzato.

Nel capitolo 6 descriveremo in dettaglio il caso di studio, dalla descrizione funzionale alla costruzione del modello dati e infine la realizzazione di tutti i report.

Nel capitolo 7 descriveremo in dettaglio le conclusioni su quanto visto nell'elaborato.

Infine si metterà insieme il tutto con un'applicazione atta ad analizzare i dati del portafoglio ordini e di vendita al dettaglio modellando i dati in uno Star-Schema e creando i report sui dati della nuvola aggiornata quotidianamente e più volte al giorno.

# Capitolo 1

## BUSINESS INTELLIGENCE

---

### **1 - La Business Intelligence nel contesto competitivo attuale**

”All’interno dei sistemi aziendali, gli attori che prendono le decisioni sono immersi in una serie di flussi informativi, ricevono informazioni che riflettono lo stato del sistema in cui operano e utilizzano tali informazioni per prendere le decisioni”.

L’ambiente in cui si trova ad operare l’impresa moderna è turbolento. I sistemi aziendali, di per sé complessi, grazie all’aiuto delle tecnologie informatiche possono trovare indubbio giovamento in termini di correttezza e tempestività decisionali. Fondamentale risulta inoltre la capacità dell’organizzazione di adattarsi a un’era che passa dall’informazione alla conoscenza, attuando politiche che permettano di ottenere un’organizzazione snella che percepisca il valore fornito dalla tecnologia e dalla metodologia comune.

Il contesto competitivo attuale ha assistito, nell’arco degli ultimi quindici anni, ad un notevole cambiamento delle sue dinamiche evolutive.

Si è passati da un ambiente piuttosto statico, o che si evolveva comunque secondo linee di crescita prevedibili, a un contesto in cui le variabili in gioco sono estremamente mutevoli sia in termini qualitativi che quantitativi sia di rapidità di cambiamento.

I fattori che hanno determinato questa evoluzione sono molteplici e differenti:

- l'affermarsi della globalizzazione (favorita da fenomeni di carattere politico, economico e sociale) ha aperto le frontiere di nuovi mercati accrescendo le dimensioni del contesto competitivo;
- si è avuta una corrispondente crescita della domanda per le imprese multinazionali, ma si è anche assistito a una crescente competizione tra imprese operanti nello stesso settore;
- lo sviluppo eccezionale delle tecnologie informatiche e la loro relativa diffusione su scala mondiale hanno accorciato le distanze tra mercati che in passato, con molta probabilità, non sarebbero mai entrati a contatto;
- l'attuale instabilità dei mercati ha determinato un aumento della variabilità della domanda sia in termini quantitativi sia qualitativi;
- a queste condizioni l'azienda moderna deve rispondere rendendo flessibili la sua struttura aziendale e le capacità dei singoli componenti, in modo da spingere al massimo (fino al limite teorico della personalizzazione) la gamma e la varietà dei prodotti e servizi forniti;
- la turbolenza del contesto e la sua elevata competitività hanno determinato la riduzione del ciclo di vita dei prodotti e la conseguente necessità, da parte delle aziende, di ridurre il corrispondente *time to market*.

In un contesto di questo tipo la Business Intelligence (BI) riesce a intercettare e a rispondere alla gran parte delle necessità con cui ogni azienda viene a contatto. Proprio per la trasversalità dei suoi interventi ai vari livelli aziendali e per l'impatto che fornisce a livello di efficienza organizzativa, la BI si configura come il settore in grado di dare le migliori risposte alle aziende che si trovano a dover "navigare" nelle acque turbolente del mercato moderno.

Secondo una definizione coniata dai pionieri negli anni '80:

*“La Business Intelligence è il prodotto dell'analisi di dati quantitativi di business. E' destinata a produrre riflessioni atte a consentire ai responsabili aziendali di operare decisioni consapevoli e informate, oltre che a stabilire, modificare e trasformare le strategie e i processi di business in modo tale da trarne vantaggi competitivi, migliorare*

*le performance operative e la profittabilità e, più in generale, raggiungere gli obiettivi prefissati”.*

Un'altra definizione coniata da Howard Dresner, analista del gruppo Gartner, nei primi anni '90:

- un insieme di processi aziendali per raccogliere e analizzare informazioni strategiche;
- la tecnologia utilizzata per realizzare questi processi;
- le informazioni ottenute come risultato di questi processi;

Coerentemente con quanto appena affermato, negli ultimi anni la BI si sta evolvendo verso l'analisi a 360 gradi delle performance di tutta l'impresa e tende ad essere lo strumento principale per orientare le scelte del management verso obiettivi comuni.

Testimonianza di questa affermazione sono i dati riguardanti i tassi di crescita del settore.

Si parla attualmente di una crescita media europea di circa l'11% annuo fino al 2010. Anche se, come in molti altri settori, l'Italia procede un po' a rilento, il fenomeno interessa anche la penisola.

La vitalità del settore è confermata anche dalla corsa scatenatasi, da parte dei colossi del SW, per l'acquisizione delle aziende di BI. Questo settore in forte crescita darà ai grandi produttori di SW la possibilità di sopperire al naturale rallentamento del loro settore di riferimento dovuto alla maturità del corrispondente ciclo di vita.

Andiamo ad analizzare più dettagliatamente qual è l'impatto della BI sulle dinamiche caratteristiche del mercato odierno.

## 1.1 - La BI come supporto decisionale

La già citata competitività dell'attuale contesto e la relativa turbolenza non fanno che accrescere enormemente i rischi legati alle decisioni prese dal *management*.

La rapidità del contesto attuale può far sì che decisioni prese in maniera errata o poco tempestiva abbiano come conseguenza il delinearsi di una situazione difficilmente recuperabile per l'azienda.

Ciò è dovuto oltre che all'elevata competizione del contesto attuale, anche alla presenza di una domanda sempre più esigente, informata e preparata, in grado di scegliere il proprio fornitore secondo criteri oggettivi e in grado di influenzare i comportamenti d'acquisto altrui ad esempio attraverso passaparola, web ecc.

Inoltre in un contesto in cui il vantaggio competitivo viene giocato anche sul filo di lana dei servizi complementari, della reputazione e dell'affidabilità storica di un marchio, una decisione errata e in controtendenza col resto del mercato può avere effetti irreparabili.

L'adozione di tecniche di BI, in questo senso, serve a fornire un supporto il più oggettivo e scientifico possibile al *management* nel loro processo decisionale, in maniera tale da fornire al decisore un quadro che sia il più svincolato possibile da eventi e variabili casuali.

Inoltre disporre di dati tempestivi e aggiornati può dare la possibilità ai manager di prevedere eventuali trend evolutivi del mercato e attuare gli opportuni provvedimenti in tempo, prima che le strategie pensate diventino obsolete.

Diventa importante a questo proposito, il ricorso al *reporting* come strumento di divulgazione tempestivo delle informazioni di performance aziendale.

Le peculiarità che contraddistinguono il *reporting* sono in linea con le richieste provenienti dal mercato e sono:

- tempestività e periodicità;
- storicità;
- multidimensionalità;
- integrazione;
- fruibilità.

## 1.2 - La BI come strumento di *Knowledge Discovery*

Un'altra delle caratteristiche dello sfondo competitivo che chiamano in causa l'adozione di tecniche di BI è l'evoluzione dell'impresa verso una prospettiva di *learningorganization*.

L'impresa non è più, come nella visione classica, un'entità esecutrice di routine organizzative formalizzate e burocratizzate. È questa l'età in cui cresce il peso delle informazioni e della conoscenza (di pari passo con la crescita dei servizi a esse correlati) nella determinazione del "valore" di un'azienda, l'età in cui si assiste al prevalere del peso delle risorse intangibili su quelle tangibili.

Emergono due macro-necessità in un contesto di questo tipo che la BI può soddisfare:

- elaborare ed acquisire il maggior quantitativo possibile di informazioni, che crescono esponenzialmente in virtù dell'esplosione delle tecnologie informatiche e del numero di *competitors* presenti sul mercato (funzione di *Data Warehousing*);
- veicolare queste informazioni per giungere a creare conoscenza (funzione di *Knowledge Discovery*).

### 1.2.1 - La funzione di *Data Warehousing* della BI

*"The amount of data that requires processing and analysis in a large database exceeds human capabilities, and the difficulty of accurately transforming raw data into knowledge surpasses the limits of traditional databases. Therefore, the full utilization of stored data depends on the use of knowledge discovery techniques."*

Da sempre il quantitativo di dati immagazzinati (tramite sistemi automatici e non) nelle più svariate applicazioni pratiche, ha sopravanzato di gran lunga la capacità umana di analizzarli. Questo divario tra dati immagazzinati ma non analizzati rischia di crescere in maniera importante negli ultimi anni per due principali ragioni:

- il boom delle tecnologie informatiche ed in particolare delle capacità di memorizzazione;
- la diffusione di queste tecnologie presso tutti gli attori presenti nell'attuale contesto competitivo.

È pur vero che solo le accresciute capacità di memorizzazione e di elaborazione hanno reso possibile l'adozione delle prime tecniche di analisi dei dati e di sintesi automatica in informazioni.

Ad ogni modo anche ai nostri giorni la capacità di elaborazione ed analisi dei dati non regge il passo della corrispondente crescita delle capacità di memorizzazione.

Si vengono quindi a creare dei veri e propri "cimiteri" di dati sepolti nei DB aziendali, cosa non proprio auspicabile in un'era in cui la conoscenza diventa competenza distintiva e quindi fonte di vantaggio competitivo per le aziende. In termini pratici quindi la maggior parte delle informazioni presenti nei DB sono "nascoste", ovvero non vengono utilizzate in maniera proattiva e, per di più, rappresentano un costo di gestione per l'azienda, il cui peso è tanto più rilevante quanto maggiori sono le possibilità che questi dati possano restituire conoscenza.

***"Le aziende utilizzano mediamente il 7% delle informazioni che possiedono al fine di informare solo il 5% del personale".***

L'analisi efficiente ed efficace di questi dati, non può essere eseguita utilizzando i tradizionali DBMS relazionali.

Essi infatti sono orientati all'organizzazione delle informazioni disponibili seguendo una logica di normalizzazione dei dati che da un lato ne consente la memorizzazione efficiente, ma dall'altro comporta la loro estrema frammentazione ed il conseguente appesantimento delle operazioni necessarie per eseguire interrogazioni più o meno complicate.

Si ha bisogno, nel contesto attuale, di una differente logica di consultazione dei dati, che consideri una porzione dei dati, quelli realmente "interessanti".



L'analisi multidimensionale, che costituisce la base teorica di modellazione di un'applicazione BI, soddisfa proprio queste specifiche. Tramite essa infatti si procede alla realizzazione di una base dati che indaga i soli "fatti" e le sole "dimensioni" davvero importanti per l'analisi.

Le tecniche di analisi multidimensionale producono uno schema di riferimento volontariamente denormalizzato, contenente dati anche non atomici e che predilige la facilità d'utilizzo per l'utente alla sua efficienza in senso assoluto.

La pre-aggregazione dei dati ad un livello non atomico invece, consente di eseguire interrogazioni anche complesse con tempi di risposta decisamente migliori rispetto a quelli forniti dai sistemi transazionali.

### **1.2.2 - La funzione di Knowledge Discovery della BI**

Una definizione piuttosto diffusa adottata per la *Knowledge Discovery* è la seguente: *"The nontrivialextraction of implicit, previouslyunknown and potentiallyuseful information from data"*.

La corretta progettazione di una base dati in logica multidimensionale mette a disposizione dell'azienda che l'adotta una nuova modalità di consultazione dei dati presenti nei suoi DB.

Questa nuova prospettiva di visione le consente di avere a disposizione informazioni meglio correlate tra loro e quindi di disporre di maggior conoscenza potenziale.

Ma perché la conoscenza è diventata una leva di vantaggio competitivo?

Facendo sempre riferimento al contesto competitivo attuale, il valore di un'impresa non coincide più col valore del suo capitale finanziario ma è superiore ad esso e questa differenza di valore è spiegata dal cosiddetto "capitale intellettuale", ovvero dalla valorizzazione delle risorse intangibili possedute.

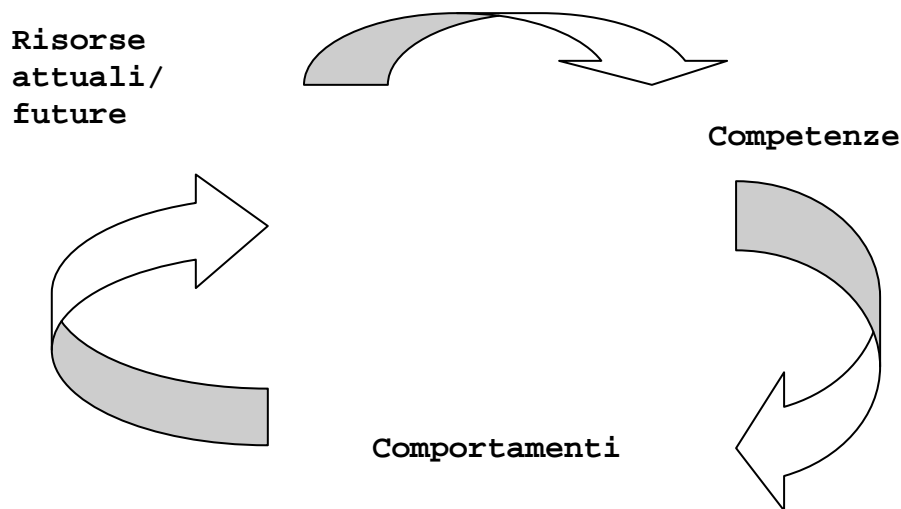
L'azienda moderna non è più solo “quello che vende” ma anche e soprattutto “come realizza quello che vende”.

Diventa chiaro come il processo di sviluppo aziendale si leghi in questo modo in maniera indissolubile al processo di sviluppo autopoietico delle risorse detenute.

Le risorse infatti, possono essere definite come l'insieme dei fattori tangibili e intangibili che l'impresa controlla e utilizza nei suoi processi operativi per realizzare profitto ma anche per autoalimentare la generazione di altre risorse.

Si configura quindi un processo circolare di accumulazione delle risorse che porta ogni impresa a evolvere in modo specifico rispetto ai suoi *competitors*.

Ogni risorsa è quindi il risultato di un processo basato su risorse precedenti (e sui comportamenti che ne derivano) e, al tempo stesso, l'origine delle risorse future che caratterizzeranno l'impresa.



**Fig. 1 Il processo di sviluppo autopoietico delle risorse**

È infatti il modo in cui le risorse vengono “metabolizzate” all'interno della struttura organizzativa aziendale che determina il suo percorso di crescita futuro.

Questo processo di “metabolizzazione” ha come output la generazione di nuove competenze aziendali (distintive e non), le quali guidano i comportamenti futuri nei vari ambiti d'azione (produttivo, commerciale, finanziario, organizzativo ecc.) e, di conseguenza, l'accumulazione delle risorse allo step successivo.

Lo sviluppo di un'azienda viene sempre più determinato quindi da una sorta di DNA insito nella cultura aziendale e trasmesso a tutti i componenti.

Questo DNA altro non è che la conoscenza generata in ambito aziendale attraverso un processo di sedimentazione delle risorse accumulate dall'organizzazione al suo interno.

La conoscenza viene definita da Vicari come *“l'insieme di schemi cognitivi sufficientemente stabili e diffusi all'interno dell'impresa. Questi schemi cognitivi sono alla base dei comportamenti aziendali, in particolare del modo in cui sono utilizzate le risorse disponibili, e del processo di accumulazione del patrimonio di risorse”*.

In un'ottica di questo tipo, col prevalere dell' “apporto umano” su quello “meccanico” ai fini degli obiettivi aziendali, ogni risorsa può essere interpretata in termini di conoscenza.

Il coordinamento e la valorizzazione di queste risorse diventano strumento fondamentale per il raggiungimento degli obiettivi e, di conseguenza lo stesso processo di creazione di valore per l'impresa può essere interpretato come processo di generazione, utilizzazione, accumulazione e rinnovo della conoscenza.

Come verrà approfondito successivamente, l'utilizzo di tecniche di BI è il primo passo necessario per “dare un significato” alle enormi quantità di dati “imprigionate” nei DBMS aziendali e che senza le tecniche di *Knowledge Discovery* non potrebbero generare conoscenza.

Attraverso le sue due principali funzioni, la BI permette all'impresa di disporre di un quantitativo maggiore di informazioni da cui generare conoscenza e, soprattutto fornisce una differente modalità di analisi di questi dati, più centrata sulle informazioni di reale interesse in ambito aziendale e quindi più direttamente orientata alla generazione di conoscenza come fonte di vantaggio competitivo.

A riprova dell'importanza di questo filone di studio, ci si sofferma ulteriormente sull'approccio strategico aziendale basato sulla conoscenza (*Knowledge Based*).

### 1.2.3 - Strategie aziendali *Knowledge Based*

La crescente importanza della conoscenza nell'ambito delle risorse possedute dall'azienda ha fatto sì che venisse elaborato un approccio alle strategie aziendali dedicato.

Secondo questo approccio la conoscenza dell'impresa (o conoscenza organizzativa) è considerata come una risorsa fondamentale per il conseguimento e soprattutto per il sostenimento di un vantaggio competitivo.

Non è solo la conoscenza in sé a determinare una forma di vantaggio competitivo per l'azienda, ma anche e soprattutto gli effetti che essa ha sulle procedure, le competenze ed il "*modus operandi*" aziendale.

La conoscenza è una risorsa specifica e inimitabile di un'azienda (in quanto output di un processo autonomo di sviluppo di risorse e competenze) e per questo garantisce un vantaggio competitivo duraturo (a patto che s'intervenga per modificarla attivamente in risposta alle mutate condizioni ambientali).

Questo approccio individua due "tipologie" di conoscenza, a seconda delle modalità di manifestazione nel contesto organizzativo:

- la conoscenza "superficiale" comprende gli aspetti più visibili, quelli codificati nel sistema organizzativo e sedimentati nel suo tessuto. Si tratta di schemi cognitivi diffusi all'interno dell'azienda, e che rendono prevedibile il comportamento di ciascun componente in risposta a determinati stimoli o per gestire specifiche situazioni. La diffusione interna di questa conoscenza è fondamentale per favorire lo sviluppo di un linguaggio comune che agevoli le relazioni tra i vari attori aziendali e stimoli l'adozione di comportamenti uniformi e coerenti.

Queste caratteristiche intersecano perfettamente il concetto di cultura, obbiettivo e valori tipici di un'azienda.

La conoscenza superficiale (e la sua diffusione) è il primo passo per il raggiungimento di condizioni di efficienza interna.

Fossilizzarsi troppo su questa sorta di DNA comportamentale consolidato può però diventare controproducente per l'azienda moderna, specie nel medio – lungo termine. Il rischio è che si frenino i processi di creazione di nuova conoscenza e che l'azienda diventi “insensibile” alle nuove richieste emergenti dall'ambiente esterno.

Per evitare ciò l'azienda agisce sulla conoscenza “profonda”;

- la conoscenza “profonda” riguarda la capacità dell'azienda di saper fronteggiare al meglio le mutate dinamiche dell'ambiente esterno, generando nuovi schemi cognitivi, nuove modalità del “saper fare” e diffondendole efficacemente al suo interno. L'abilità aziendale di creare questo tipo di conoscenza dipende dalla sua capacità di apprendere dall'interazione con l'ambiente e dalle esperienze passate. È a questo livello che si esplicita il significato di *learning organization*: l'azienda moderna evolve, impara e cresce proprio come un organismo vitale. Le decisioni prese, le esperienze passate determinano il futuro, il destino e il successo dell'azienda.

Quindi attraverso la conoscenza “profonda” l'impresa apprende dalle nuove situazioni e genera nuova conoscenza “superficiale”, quindi nuove procedure organizzative orientate all'efficienza interna.

In definitiva l'output del processo di generazione della conoscenza è l'individuazione continua di nuove routine organizzative che permettano di rispondere meglio e in maniera distintiva alle mutate necessità provenienti dall'ambiente esterno.

Raccomandazione fondamentale nella generazione della conoscenza è la condivisione delle “scoperte” effettuate al fine di integrare al meglio le varie risorse aziendali, accrescere la loro coesione e la coerenza dei processi interni.

L'utilizzo di una piattaforma comune fornisce un metodo di lavoro comune alle persone di un team, migliora la collaborazione, la competitività sul mercato della soluzione fornita e permette quindi di dare in output al processo decisioni più sicure.

A questo proposito può essere interessante indagare su quali conoscenze siano effettivamente trasmissibili e condivisibili e su quali no, e di conseguenza capire anche i livelli della conoscenza su cui la BI può impattare.

Analizziamo quindi il modello del *knowledgecreationprocess* di Nonaka.

#### **1.2.4 - Il modello del *Knowledge Creation Process***

Sviluppato inizialmente da Polanyi ed in seguito re-interpretato da Nonaka, questo modello suddivide la conoscenza generata in ambito aziendale tra conoscenza tacita ed esplicita:

- *la conoscenza esplicita* si configura come quella forma di conoscenza che risulta codificabile e trasmissibile in linguaggi formali e sistematici e che può quindi essere articolata in una forma strutturata (documenti, programmi, presentazioni, reports ecc.);
- *la conoscenza tacita* è intuitiva e non verbalizzabile e quindi difficilmente comunicabile e formalizzabile e comprende sia elementi cognitivi sia tecnici. La dimensione tecnica comprende sia il contenuto informativo sia gli aspetti tecnici specifici (*skills*), mentre quella cognitiva consiste di modelli mentali, percezioni, valori e credenze.

Se la conoscenza diventa vantaggio competitivo solo quando viene diffusa, condivisa ed opportunamente rielaborata, si capisce come la continua trasformazione e conversione della conoscenza sia la modalità migliore per svilupparla ed accrescerla in accordo col modello di sviluppo autopoietico delle risorse in competenze e quindi in conoscenza.

Lo studio di Nonaka parte da questa assunzione ed analizza le possibili modalità di conversione della conoscenza in ambito aziendale, prendendo le mosse dalla classificazione della conoscenza in tacita ed esplicita.

- la socializzazione è il passaggio di conoscenza tacita. Può avvenire a livello individuale, attraverso la condivisione di esperienze tramite l'interazione tra gli

individui, ma anche a livello organizzativo mediante lo sviluppo di una cultura che costituisca un collante valoriale comune;

- l'esternalizzazione consiste nell'esplicitazione della conoscenza tacita mediante il ricorso alla produzione di documenti oppure analogie e metafore;
- l'internalizzazione rappresenta l'apprendimento, a livello individuale o organizzativo, di conoscenza esplicita. È legata ad una attivazione intellettuale da parte dell'individuo o, nel caso di un apprendimento organizzativo è facilitata dalla creazione di un *knowhow* aziendale o di routine formalizzate;
- la combinazione infine è il passaggio di conoscenza esplicita tra individui o tra i diversi livelli dell'organizzazione. Può essere facilitato dall'utilizzo di una infrastruttura di IT che favorisca questo scambio di conoscenze anche attraverso la produzione di nuove forme di conoscenza esplicita.



**Fig. 2** Le modalità di conversione della conoscenza

Lo sviluppo della conoscenza organizzativa è alimentato da queste continue conversioni delle diverse forme di conoscenza ai diversi livelli di analisi: individuo, gruppo, organizzazione ed anche relazioni con l'ambiente esterno.

È proprio l'interiorizzazione di conoscenza tramite lo scambio con attori esterni che determina la capacità di un'azienda moderna di attrarre a sé quelle forme di

organizzazione interna che le permettono di avere una cultura organizzativa che le garantisca di rispondere efficacemente alle mutabili richieste provenienti dal contesto.

Questa forma di sviluppo di nuova conoscenza è quella che permette all'azienda di sopravvivere a lungo nello scenario attuale, sviluppando internamente una forma di vantaggio competitivo difficilmente imitabile e quindi intrinsecamente sostenibile.

Un'azienda che riesce a ripetere sistematicamente questa adattabilità alle richieste del mercato mantenendo comunque un elevato livello di servizio, sviluppa una forma di vantaggio competitivo intrinseca e inimitabile.

Questa caratteristica coincide con le richieste di flessibilità, rapidità di produzione ed alto livello di servizio richieste dal mercato odierno, ma si pone ad un livello superiore poiché le racchiude tutte al suo interno.

La BI rappresenta un valido supporto per lo sviluppo di tutte le forme di condivisione della conoscenza presentate ed agisce quindi ai vari livelli.

In tutti i casi sopra elencati, la BI fornisce una piattaforma comune di lavoro ed un metodo comune di lavoro che facilitano la condivisione di documenti, migliorano la coerenza interna dei processi e della cultura aziendale e standardizzano il dizionario dei termini e dei concetti aziendali.

Interventi di questo tipo hanno come effetto l'individuazione di un indirizzo ed un percorso comune a tutte le unità aziendali che, guidato da quelli che sono i valori e la cultura tipici dell'impresa, favoriscono la creazione di un orientamento comune volto allo sviluppo e alla diffusione della conoscenza quale forma di vantaggio competitivo fondamentale nel contesto attuale di riferimento.

In definitiva la prospettiva strategica basata sulla conoscenza considera la conoscenza come risorsa strategica e prende in considerazione i meccanismi e i processi mediante i quali questa si genera, si trasforma, si trasferisce tra individui e organizzazioni.

Riassumiamo le principali considerazioni a riguardo del processo di accumulazione della conoscenza emerse nella trattazione:

- per il successo di questo processo è necessaria la disponibilità di un contesto organizzativo che si caratterizzi per la comunanza di linguaggi e schemi



cognitivi condivisi, che così facendo favorisca la trasformazione delle conoscenze individuali in organizzative accrescendo di conseguenza il valore complessivo dell'impresa;

- l'incremento dello stock di conoscenze aziendali rappresenta un fattore critico per la crescita delle imprese, attraverso la creazione di un DNA aziendale rivolto al soddisfacimento efficiente ed efficace delle richieste del mercato. In questo modo le conoscenze accumulate e sedimentate diventano una forma di vantaggio competitivo durevole e sostenibile da parte dell'impresa;
- un intervento che le aziende possono mettere in atto per favorire la diffusione della conoscenza è la predisposizione di ruoli e soluzioni organizzative di coordinamento che abbiano il compito di accrescere l'efficacia dei processi di gestione della conoscenza.

### **1.3 - La BI come fonte di vantaggio competitivo**

Nel contesto competitivo presentato diventa sempre più difficile per le aziende raggiungere una posizione di vantaggio competitivo (creando quindi valore aggiunto per la domanda finale) attraverso la leva della differenziazione.

O meglio la posizione di vantaggio competitivo raggiunta differenziando è sempre più difficilmente difendibile proprio a causa dell'elevata competizione intersettoriale (che facilita lo sviluppo di processi imitativi) e dello sviluppo tecnologico che favorisce la diffusione di informazioni correlate alla leva competitiva adottata (concetto di trasparenza del mercato).

In un'ottica di questo tipo se da una parte l'impresa non può rinunciare a cercare di differenziare (sostenendo investimenti sempre crescenti in R&S), d'altra parte ha come obbligo la ricerca di soluzioni progettuali e di processo che portino al raggiungimento di una posizione di leadership di costo o comunque al raggiungimento di una condizione di massima efficienza interna.

Attualmente, la capacità di contenere i costi nella massima misura possibile rappresenta una condizione di competitività spesso irrinunciabile.

In questa prospettiva si percepisce l'importanza dell'adozione di tecniche di BI all'interno delle aziende moderne. La BI infatti rappresenta, come vedremo a fondo in seguito, una possibilità di razionalizzazione e reingegnerizzazione delle attività e dei processi interni dell'azienda che ne adotta le soluzioni.

Questa attività fornisce una struttura organizzativa più snella e flessibile all'azienda, in maniera tale che essa riesca a rispondere in maniera più rapida agli stimoli provenienti dal contesto.

Lo sviluppo delle risorse intangibili assume un peso sempre maggiore nell'ottica evolutiva aziendale.

Questo fattore impatta in maniera importante anche sulle performance interne aziendali.

La costituzione di un DNA, una sorta di coscienza aziendale che guida comportamenti e genera e condivide conoscenza produce i seguenti benefici:

- maggiore coerenza dei processi produttivi con le condizioni interne che caratterizzano l'impresa;
- migliore accesso alle risorse necessarie per la realizzazione dei processi produttivi;
- miglior connessione tra i processi produttivi interni all'impresa.

Si può pensare ad una sorta di parallelismo tra le forme di vantaggio competitivo e le tipologie di conoscenza generate all'interno dell'azienda.

Infatti, se da una parte la riduzione dei costi è un'attività irrinunciabile ma che non garantisce la sostenibilità del vantaggio competitivo raggiunto, d'altra parte la differenziazione è più difficilmente sostenibile ma, una volta raggiunta, più duratura e difficilmente imitabile.

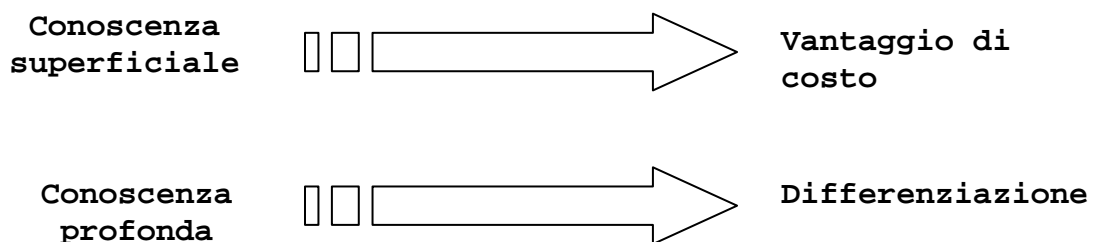
Discorso analogo può essere fatto per le conoscenze generate dall'azienda.

Da una parte abbiamo la conoscenza “superficiale” la cui promozione e diffusione è indispensabile e che rappresenta una sorta di condizione necessaria ma non sufficiente per la sopravvivenza alle attuali condizioni del contesto.

D'altra parte la conoscenza “profonda” rappresenta una capacità distintiva ed inimitabile di un'impresa, attraverso la quale essa riesce a rielaborare in maniera personale gli stimoli provenienti dall'esterno e generare nuova conoscenza “superficiale” che massimizzi le performance interne.

È proprio questo tipo di conoscenza che diventa la caratteristica distintiva di un'azienda, che essa possiede al suo interno e che la rende unica agli occhi dei suoi clienti.

Un ulteriore aspetto che va considerato a proposito, riguarda l'evoluzione dei rapporti cliente – fornitore tra aziende nel contesto competitivo attuale.



**Fig. 3 Parallelismo tra conoscenza e forme di vantaggio competitivo**

Si è assistito in questi anni al passaggio da rapporti di tipo conflittuale in cui la controparte detentrica di un maggior potere contrattuale dettava le leggi della transazione, ad una situazione di tipo collaborativo tra le parti. In quest'ultimo caso si instaura un rapporto tale che le due parti si trovano a lavorare a stretto contatto condividendo informazioni delicate, processi operativi, competenze e quindi conoscenza.

Rapporti di questo tipo consentono all'azienda di allargare il proprio spettro di conoscenze e competenze, confrontandosi col mercato e con le richieste provenienti dalla controparte.

In questa maniera l'azienda aggiorna allo stato dell'arte in maniera continua il suo bagaglio di conoscenza e, per fare ciò, sviluppa internamente quelle routine (formalizzate e non) che portano alla formazione di quella attitudine e quel DNA essenziali per rendere duraturo e sostenibile il vantaggio competitivo derivante dalla formazione della conoscenza.

Questo è sempre più vero quanto più si considera che il mercato evolve, come già detto, verso una situazione in cui le risorse intangibili assumono un peso sempre maggiore ed in cui il focus del vantaggio competitivo si sposta più sui servizi associati ad un prodotto che al prodotto stesso.

La BI incrocia questi processi di formazione del vantaggio competitivo sia garantendo una piattaforma comune per la condivisione e la diffusione della conoscenza, sia fornendo le opportunità di ottenimento di conoscenza dall'esterno attraverso la costituzione di rapporti di partnership per l'implementazione stessa di soluzioni BI.

Un altro aspetto da considerare, ultimo ma non meno importante, riguarda la sempre maggiore completezza e flessibilità richiesta ai vari ruoli aziendali, con conseguente decentramento del potere decisionale.

È anche questo un aspetto derivante dalle mutate dinamiche competitive e sul quale la BI impatta ancora una volta in maniera importante.

## **1.4 - La BI e l'evoluzione della struttura aziendale**

In risposta alla già citata evoluzione dell'ambiente di riferimento, si è assistito ad un corrispondente mutamento della struttura aziendale.

Questa evoluzione ha comportato la trasformazione della moderna azienda da una struttura gerarchica verticale in una orizzontale nella quale il focus si sposta dalla specializzazione nella singola funzione all'ottimizzazione del processo visto nella sua interezza.

Qui per processo s'intende l'esecuzione di attività che nella vecchia concezione riguardavano più funzioni aziendali, ma che adesso vengono viste come un tutt'uno proprio perché concorrono al raggiungimento di un singolo obiettivo.

Se da un lato la struttura verticale prevedeva un controllo rigido e gerarchico che partiva dall'alto e giungeva fino alle singole funzioni, in una struttura verticale si assiste ad una sorta di frammentazione del potere decisionale.

Ogni componente dell'azienda è responsabile (anche in termini decisionali) dell'attività di sua competenza, in altri termini ne diventa il *processowner*.

I singoli componenti aziendali quindi non sono più dei semplici esecutori di mansioni predefinite, ma diventano dei centri di decisione, delle entità che devono essere capaci di decidere per il meglio con gli strumenti a loro disposizione.

Il componente (a qualsiasi livello) dell'azienda moderna quindi, dovrà essere caratterizzato da un bagaglio di competenze che non potrà essere solamente tecnico ma anche, e con un peso sempre maggiore, da un insieme di competenze organizzative e umane che gli permettano di capire le variabili condizioni ambientali che gli si presentano e prendere decisioni di conseguenza.

Ad ogni componente delle aziende "moderne" è richiesta capacità di adattamento a nuove *routine* organizzative, capacità di cambiare rapidamente (se necessario) il proprio *modus operandi*, capacità collaborative e relazionali. La resistenza al cambiamento, tipica di strutture e personale fortemente burocratizzati sono assolutamente deleteri in un contesto competitivo come quello attuale.

Si può sostenere che quella osservata è quindi una trasposizione dell'evoluzione aziendale al ruolo della singola persona. Infatti se da un lato l'azienda evolve nella direzione dell'apprendimento, dello sviluppo delle relazioni e della conoscenza allo stesso tempo per l'individuo prevale il lato umano, relazionale e decisionale sull'aspetto puramente tecnico.

Questa frammentazione del potere decisionale non cancella la presenza del controllo gerarchico, ma ne varia il tipo di intervento.

Si passa dall'adozione di routine di controllo formalizzate e burocratizzate ad un controllo maggiormente basato sul dialogo, sull'interazione coi vari *processowner* e

sull'individuazione di linee di guida di comportamento nel medio – lungo termine che solo l'esperienza di un *manager* può prevedere.

In una situazione di questo tipo diventa importante per l'azienda moderna disporre di strumenti di supporto alle decisioni tempestivi, efficienti e soprattutto diffusi e profilati.

La frammentazione del potere decisionale infatti richiede l'adozione di soluzioni rapide e che siano il più possibile condivise o quantomeno integrate con le decisioni prese dagli altri *processowner*.

Emerge ancora una volta l'importanza dell'adozione di tecniche di BI, che forniscono strumenti comuni per la condivisione delle informazioni e quindi delle scelte prese in una visione più generale, e non limitata alla sola attività di propria competenza.

## **1.5 - La BI e la necessità di nuove competenze**

Si è accennato all'inizio di questo capitolo, al grande sviluppo su scala mondiale che interessa il mondo della BI. Quando si parla di soluzioni BI nella stragrande maggioranza dei casi si fa riferimento a progetti eseguiti da società di consulenza specializzate.

Questo boom delle consulenze targate BI segue di pari passo l'evoluzione delle dinamiche competitive attuali. Si ha in questi anni infatti una crescente deverticalizzazione delle funzioni aziendali attraverso il ricorso all'*outsourcing*.

L'esplosione delle richieste (indotte meno dall'azienda) da parte degli utenti finali ha comportato una corrispondente esplosione delle competenze necessarie per presidiare il contesto.

Le aziende non riescono più a detenere completamente al loro interno tutte le competenze di cui hanno bisogno, quindi devono rivolgersi all'esterno scegliendo di continuare a presidiare quelle che sono le loro *core competences*, ovvero quelle competenze che gli garantiscono una forma di vantaggio competitivo.

Questo discorso vale ovviamente, anche per il mondo della BI.

Sono poche attualmente le aziende che possono sostenere internamente i costi legati alla realizzazione di un progetto di BI. Una scelta di *insourcing* di questo tipo richiederebbe investimenti elevati in termini di formazione, costi fissi iniziali (ad esempio per l'acquisizione degli strumenti SW utilizzati) e potrebbe portare a risultati solo in un'ottica di medio – lungo termine.

D'altra parte il ricorso ad una consulenza BI è un passo importante e delicato per la necessità di adottare un approccio collaborativo tra le parti in un'ottica di medio – lungo termine e per la necessità di individuare il partner “giusto”.

Analizziamo nel dettaglio i vantaggi e gli svantaggi, a livello organizzativo, derivanti dall'adozione di una soluzione BI in *outsourcing*:

- l'azienda cliente si trova a dover condividere col fornitore informazioni quantomeno delicate riguardanti i dati aziendali presenti sui propri DB, ma anche riguardanti le logiche e i processi aziendali;
- l'adozione di un'ottica collaborativa dà il via ad una transazione del tipo *winwin*, ovvero gioco a somma positiva, nella quale l'obiettivo finale è il raggiungimento di un ottimo globale;
- il cliente usufruisce di prestazioni migliori rispetto a quelle che riuscirebbe a ottenere internamente. Ciò è dovuto al fatto che l'azienda consulente è specialista del settore e può sviluppare in esso economie di scala, di scopo e di esperienza;
- i risultati non sono immediati a causa della necessità di allineare processi, modalità di comunicazione e di lavoro differenti;
- in un'ottica di medio – lungo termine, quando il rapporto tra le parti diventa consolidato e vengono superate le difficoltà di coordinamento iniziali, il fornitore è in grado di adeguare sempre più i propri servizi alle richieste del cliente;
- la scelta di un partner che si riveli errato (per incapacità tecniche, perché non rispetti standard previsti da contratto ecc.) comporta dei costi di uscita molto elevati, sia per gli investimenti in costi fissi già sostenuti, sia per la difficoltà di trovare un nuovo partner che sia competitivo fin da subito.

## 1.6 - Caratteristiche di un sistema di BI

Tra i sistemi di supporto alle decisioni, i sistemi di Business Intelligence sono probabilmente quelli su cui negli ultimi anni si è maggiormente focalizzata l'attenzione sia nel mondo accademico che industriale. Un sistema di BI può essere definito come il processo di trasformazione dei dati in informazioni e infine in conoscenza, nato nel mondo industriale agli inizi degli anni '90 per soddisfare le richieste dei manager di analizzare efficientemente ed efficacemente i dati di impresa in modo da comprendere meglio la situazione del proprio business e quindi migliorare il processo decisionale. Le esigenze che ne hanno decretato la nascita derivano dalle carenze rilevate in modo più ricorrente da parte degli utenti finali dei sistemi informativi tradizionali:

- i knowledge-worker possono accedere ad una quantità notevole di dati tuttavia non possiedono le tecnologie per ottenere, nella forma desiderata, i dati necessari;
- in contesti aziendali medio-grandi, dove coesistono molteplici basi di dati relative a differenti aree del business, capita frequentemente che i risultati prodotti all'interno delle varie aree siano inconsistenti tra loro;
- l'utente finale non riesce a condurre l'analisi in maniera flessibile ed estemporanea, bensì deve incaricare il personale tecnico di formulare l'interrogazione ad hoc che effettui i calcoli necessari;
- la manipolazione delle informazioni al massimo livello di dettaglio, come sono presenti nelle basi di dati operazionali, non consente di focalizzare l'attenzione sulle informazioni veramente significative;
- i dati operazionali sono spesso affetti da errori, inconsistenze e talvolta sono addirittura mancanti.

I fattori distintivi che contraddistinguono un sistema di BI coincidono con le soluzioni ai problemi riscontrati:

- accessibilità ad utenti con conoscenze limitate di informatica;
- integrazione dei dati sulla base di un modello standard dell'impresa;
- flessibilità di interrogazione;



- sintesi dei dati per permettere analisi mirate ed efficaci;
- rappresentazione multidimensionale per fornire una visione intuitiva del modello di business;
- correttezza e completezza dei dati.



Figura Tipica architettura di un sistema di BI

## Capitolo 2

# I SISTEMI DSS

---

Lo sviluppo e la conseguente diffusione di sistemi informatici in tutte le aziende (avvenuta negli ultimi venti anni) ha portato queste ad avere a disposizione un'enorme mole d'informazioni. Dapprima a questi dati non era stata data molta importanza strategica (erano usati soprattutto per tenere traccia dei diversi processi aziendali e non per l'analisi) mentre poi si è scoperta la loro reale importanza nel processo decisionale. Infatti, dopo mezzo secolo di forte espansione economica, sostenuta da una domanda sempre crescente che le aziende tentavano di inseguire, ora l'offerta è superiore alla domanda; in un simile contesto economico ogni azienda deve lottare per difendere i propri clienti e per attaccare quelli altrui; è fondamentale captare in anticipo rispetto ai concorrenti i segnali del mercato rispondendo in modo personalizzato ("one to one market") senza tuttavia dover rinunciare ai benefici delle economie di scala.

A questo scopo sono nati i sistemi DSS (Decision Support System), cioè sistemi focalizzati sul processo decisionale, diretti ai top manager e agli esecutivi, progettati per avere caratteristiche di flessibilità, adattabilità ed in grado di fornire risposte in tempi rapidi.

### **2.1 - Dai processi tradizionali ai sistemi per l'analisi dei dati**

I sistemi tradizionali, detti anche transazionali o OLTP (On-line transaction processing) utilizzando una metafora sono quelli che fanno girare le "ruote" di una organizzazione. Infatti questi processi governano la produzione e la gestione dei singoli dati aziendali provenienti dalle varie attività caratteristiche di una impresa (ordini, fatturazione, spedizioni, vendite, ecc.).

Possiamo definire un sistema transazionale come:

”Un sistema in grado di facilitare ed automatizzare la costruzione di applicazioni, la loro esecuzione e l’amministrazione. Le applicazioni transazionali sono tipicamente costruite su una rete di diversi dispositivi e consistono, principalmente, di semplici operazioni di interrogazione ed aggiornamento compiute sui dati memorizzati all’interno di un database che ha il compito di modellare un generico sistema della nostra realtà mantenendo, al suo interno, una rappresentazione dello stato reale di questo sistema. Spesso questi sistemi sono fisicamente distribuiti in territori geografici distinti, coinvolgono tecnologie hardware e software eterogenee, devono essere altamente affidabili, devono essere costantemente disponibili ed hanno stringenti vincoli in termini di performance.”

Il fine di questi sistemi è quindi quello di garantire la memorizzazione (in modo persistente) di ogni attività dell’impresa: dati sui clienti, dati sugli ordini, dati sulle offerte fatte e ricevute ecc. Sono nati con lo scopo di sostituire il troppo lento e voluminoso flusso cartaceo e quindi sono stati progettati per ottimizzare un flusso dati con tante operazioni ognuna composta da pochi dati (come l’inserimento di un cliente o di un ordine).

Le singole operazioni software (che poi si traducono in query) rispecchiano la loro controparte reale: quindi ci saranno operazioni d’inserimento ma anche di ricerca e di correzione o meglio d’aggiornamento. Tutte queste operazioni hanno una caratteristica in comune: vanno a toccare una piccolissima parte del database alla volta.

Usando la stessa metafora precedente i processi per l’analisi dei dati, detti anche analitici, permettono agli utenti di guardare e controllare ”come girano le ruote” di una organizzazione. Attraverso i sistemi DSS è possibile compiere analisi sui dati che provengono dai vari sistemi OLTP sparsi in azienda. Queste analisi consistono in aggregazioni di dati atomici (che sono dati al minimo livello di dettaglio) secondo varie prospettive. Un sistema di supporto alle decisioni è un sistema in grado di fornire chiare informazioni agli utenti in modo che essi possano analizzare dettagliatamente una situazione e prendere facilmente le opportune decisioni sulle azioni da compiere.

In altre parole un sistema DSS è un sistema informativo dedicato ad aiutare i responsabili del business in tutta la gestione d'impresa in modo da permettere decisioni più veloci, più mirate e più efficaci. Questo sistema deve quindi avere un unico obiettivo: contribuire ad incrementare le performance economiche di una impresa attraverso la fornitura di informazioni strategiche in modo semplice e veloce. Dovendo cercare di descrivere la situazione di un'azienda le query di questi sistemi coinvolgono gran parte dei dati del database e sono meno frequenti.

Dopo aver visto gli obiettivi di questi due diversi 'mondi' analizzeremo quali caratteristiche dovrà avere un sistema software per portarli a compimento.

### 2.1.1 - I punti chiave di un sistema DSS

Abbiamo visto che il termine DSS non indica un prodotto ma la strategia di memorizzare dati consolidati su un sistema informativo, dedicato ad aiutare i responsabili del business in tutta la gestione d'impresa. Vediamo allora quali sono i cinque termini chiave di questa strategia.

- **Politica.** Significa determinare quali informazioni devono essere memorizzate nel nostro sistema, stabilire l'intervallo di aggiornamento di queste informazioni, stabilire le regole di salvataggio e di sicurezza dei dati;
- **Trasformazione.** Prima di poter caricare i dati di livello base nel sistema, questi devono essere puliti e certificati in modo da rendere il valore dell'informazione memorizzata il più significativo possibile per l'utente finale. Questa trasformazione può comportare la ristrutturazione, la ridefinizione, e il ricalcolo delle informazioni base;
- **Memorizzazione.** Le sorgenti dei dati che popolano il DSS sono i vari sistemi OLTP aziendali. I dati devono essere memorizzati in modo da massimizzare la flessibilità, la maneggevolezza e l'accessibilità globale del sistema. I dati memorizzati nel sistema DSS sono storici per natura e rappresentano le misure chiave delle performance dell'impresa nel passato;

- **Analisi.** I tipi di analisi che devono essere supportati sono la "what-if analysis" e complesse computazioni su grandi volumi di dati. Queste analisi devono supportare un ambiente multiutente e devono fare uso di funzioni analitiche in grado di lavorare su dati storici e di produrre proiezioni di dati. La tecnica più flessibile e più efficace per implementare questi tipi di analisi è sicuramente l'analisi multidimensionale. Infatti, per rispondere ai tipici quesiti di un business, è necessario analizzare i dati secondo differenti prospettive. Ad esempio, un venditore di automobili sarà interessato ad analizzare le sue vendite per modello, per colore, per intervalli di tempo, etc. Utilizzando queste diverse prospettive dei dati sarà possibile rispondere a domande del tipo: "Qual' è il trend del volume di vendita lungo un certo periodo di tempo, per un dato modello e per un determinato colore?". Le differenti prospettive di analisi dei dati vengono chiamate dimensioni del business. Parleremo più approfonditamente dell'analisi multidimensionale nel capitolo successivo;
- **Accesso.** Con questo termine si intende la capacità di vedere, selezionare e manipolare i dati disponibili. Attraverso alcuni desktop tools (fogli di calcolo, query tools, strumenti di reportistica) è possibile compiere facilmente queste operazioni, mediante una interfaccia grafica che permette di navigare facilmente i dati contenuti nel DSS.

### 2.1.2 - Applicazioni di un DSS

Vediamo ora brevemente quali sono le tipologie di applicazioni che possono essere servite da un sistema DSS.

- **Applicazioni OLAP.** L'acronimo OLAP significa On-Line Analytical Processing. Con questo termine si intende una categoria di applicazioni e di tecnologie per collezionare, gestire, elaborare e visualizzare i dati in modo multidimensionale per scopi di analisi, di pianificazione e di gestione. Le funzioni di analisi e pianificazione infatti, permettono non solo la gestione dei dati storici ma anche di ottenere le proiezioni future di queste misure. Questi sono strumenti di estrema importanza per il successo di una impresa. Grazie a queste applicazioni molti requisiti di un business possono essere risolti efficacemente (budgeting, previsioni,

redditività dei clienti, analisi delle vendite, analisi dei mix di produzione, ecc.). Le caratteristiche che deve possedere un sistema per applicazioni OLAP sono: un robusto modello scientifico analitico (funzioni matematiche), ambiente read/write per supportare la what-if analysis, bassi tempi di risposta, gestione di grandi volumi di dati, grande flessibilità;

- **Applicazioni personali.** Comprendono applicazioni come fogli di calcolo, programmi statistici, tools grafici. Queste applicazioni permettono di manipolare e presentare i dati, sono sviluppate e pensate per un ambiente standalone e per accedere ad un piccolo sottoinsieme dei dati memorizzati nel sistema DSS;
- **Query standard, query ad hoc e reportistica.** Il modo più semplice per fare analisi è poter disporre di query già confezionate ed eventualmente parametrizzabili. (Es. visualizza tutti i clienti che l'anno scorso hanno acquistato il prodotto X). In alternativa è possibile costruire query specifiche al presentarsi di situazioni particolari. Richiedono continue riprogettazioni che l'utente spesso non è in grado di fare. Per costruire una query occorre prima formulare un'ipotesi di ciò che voglio ricercare. Il ruolo dell'utente principalmente passivo. La reportistica ha forma nota e aggiorna i valori alla richiesta dell'utente (vendite del mese per categorie di prodotto), in genere la presentazione o la parte cosmetica del report è in forma tabellare o grafica;
- **Data Mining.** E' difficile definire in maniera precisa un'area in continua evoluzione, quale sicuramente quella del data mining. In assenza di una definizione universalmente accettata, considereremo il data mining come: "il processo di estrazione di informazione valida, utilizzabile e precedentemente sconosciuta, da grandi databases e l'utilizzo di queste informazioni per prendere cruciali decisioni di business". Alcune delle parole utilizzate nella definizione aiutano a chiarire in cosa il data mining si differenzia rispetto ad altre discipline simili, quali il query reporting o l'OLAP:
  - *informazione valida:* un data miner, analizzando larghi insiemi di dati, prima o poi troverà qualcosa di interessante. Contrariamente a quanto dicano i super-ottimisti, è necessario controllare che i risultati ottenuti non siano errati;
  - *informazione utilizzabile:* deve essere possibile tradurre la nuova informazione in un vantaggio di business;
  - *informazione sconosciuta:* il data miner ricerca qualcosa che non è intuitivo ma,

anzi, è spesso contro intuitivo (più l'informazione si discosta dall'ovvio, infatti, più è grande il suo valore potenziale).

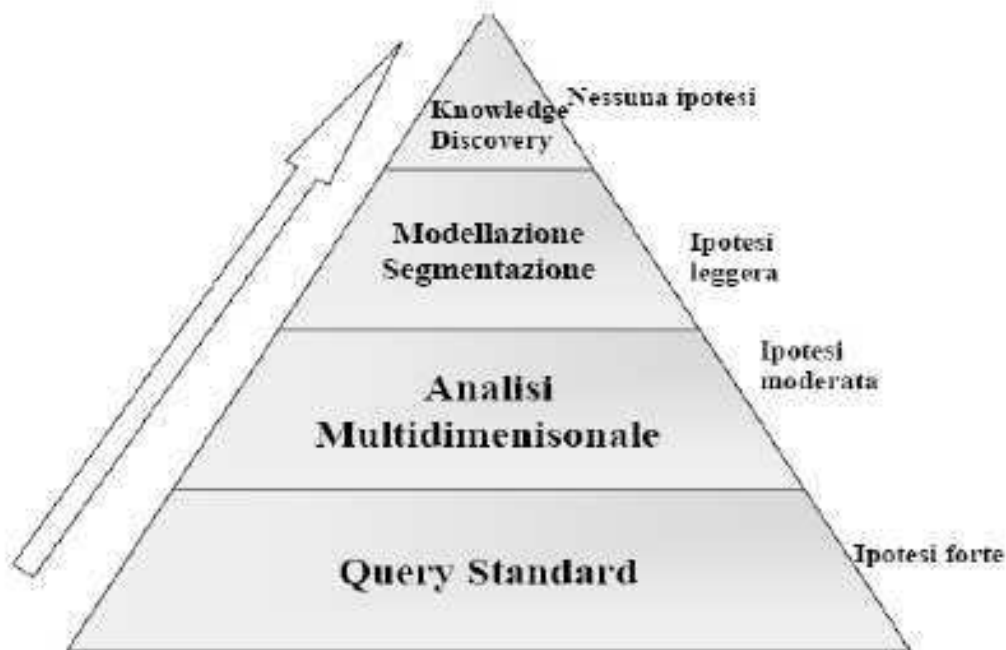


Figura La piramide della conoscenza

## 2.2 - Dal gestionale al DecisionSupport System

Esistono numerosi punti chiave che diversificano i sistemi OLTP dai sistemi DSS. Oltre a quelli già visti in precedenza, le differenze più macroscopiche riguardano :

- **la consistenza:** La consistenza OLTP è microscopica. Infatti, lo scopo di un sistema transazionale, è processare un grandissimo numero di piccole transazioni atomiche senza perderne nessuna, nel minor tempo possibile. In un sistema DSS, invece, la consistenza si misura globalmente. Questo significa che in genere non ci dobbiamo preoccupare delle singole transazioni, ma dobbiamo assicurarci che ogni operazione di caricamento sia composta da un set corretto di dati. Quindi la consistenza in un

sistema DSS è intesa come certificazione della qualità dei dati caricati dai sistemi OLTP;

- **le transazioni:** Un sistema OLTP processa migliaia o anche milioni di transazioni al giorno, ciascuna delle quali accede ad una piccola porzione di dati. Un sistema DSS, spesso, processa una sola transazione al giorno, composta però da migliaia o anche milioni di record. Questa transazione, detta Production Data Load, consiste nel caricamento dei dati dalle fonti OLTP verso il sistema DSS. Quello di cui dobbiamo preoccuparci è che lo stato del sistema sia consistente subito prima e subito dopo l'esecuzione di questa grande transazione. Nel caso in cui si debba interrompere un production data load, invece di compiere il classico rollback della transazione, troppo impegnativo per il sistema, ripristiniamo lo stato iniziale con una snapshot presa prima del caricamento;
- **il tempo:** I sistemi OLTP e i sistemi DSS trattano il tempo in modo molto differente. Il miglior sistema OLTP è una fotografia istantanea di una organizzazione aziendale. Ad ogni modo ci sono pochissime organizzazioni e pochissimi individui che vogliono compiere analisi su sistemi costantemente aggiornati. Inoltre, molti utenti non gradiscono che il database sottostante possa cambiare stato mentre la loro query è in esecuzione. Questa caratteristica dei sistemi OLTP è il primo tipo di inconsistenza temporale che vogliamo evitare in un sistema DSS. Un secondo tipo di inconsistenza temporale che affligge un database OLTP è la mancanza di un supporto per rappresentare correttamente la storia passata. In un sistema DSS, possiamo evitare questi problemi rappresentando esplicitamente delle serie temporali. Per fare ciò, bisogna muovere delle snapshot statiche, formate da dati provenienti dalle sorgenti OLTP, verso il sistema DSS. Le snapshot diventeranno gli elementi della nostra serie temporale. Questa operazione viene fatta ad intervalli regolari, tipicamente una volta al giorno;
- **il progetto del database:** La più importante differenza fra un sistema OLTP e un sistema DSS riguarda l'organizzazione dei dati all'interno del sistema o più semplicemente il modello dei dati. Molti dei miracoli ottenuti nelle performance dei sistemi transazionali sono dovuti al modello Entity Relation<sup>3</sup>. Questo modello è



volto ad eliminare completamente la ridondanza dei dati, in modo da permettere ad una transazione, che debba cambiare un qualsiasi dato, di modificare il database in un sol punto. Purtroppo però, questo modello non è altrettanto efficiente per i sistemi DSS. Infatti il modello entity relation è difficile da capire per l'utente e non può essere navigato utilmente dal software del DSS. La soluzione è il modello multidimensionale che vedremo in dettaglio nel prossimo capitolo.

A causa di queste differenze, le tecniche di progettazione proprie dei sistemi transazionali sono poco appropriate per i sistemi DSS.

## 2.3 - Sviluppo di un sistema DSS

In questa sezione, vedremo una panoramica di tutte le attività che sovrintendono la creazione e il mantenimento di un sistema di supporto alle decisioni. L'insieme di queste attività prende il nome di ciclo di vita del sistema DSS. Di seguito sono elencate e descritte queste attività:

- **pianificazione del progetto.** Pianificare significa: definire le attività del progetto, stabilire le risorse da impiegare e lo staff di risorse umane da utilizzare. Successivamente, una volta individuato l'insieme di tutte le attività, bisogna stabilire la sequenza e la relativa durata prevista. Il risultato di questo lavoro, sarà un documento strutturato che identifica ed organizza tutte le attività coinvolte nel ciclo di vita del sistema e le relative parti chiamate in causa. Questo documento sarà di vitale importanza per la gestione del progetto. Notiamo inoltre che la pianificazione è strettamente dipendente dai requisiti del business da modellare. Questa dipendenza viene mostrata dal collegamento bidirezionale esistente fra queste due attività;
- **definizione dei requisiti del Business.** La probabilità di successo di un sistema DSS è tanto più elevata quanto migliore è stata la comprensione dei requisiti del business e degli utenti finali. Senza questa fondamentale comprensione, l'intero progetto si rivelerà un fallimento. L'approccio utilizzato per raccogliere i requisiti

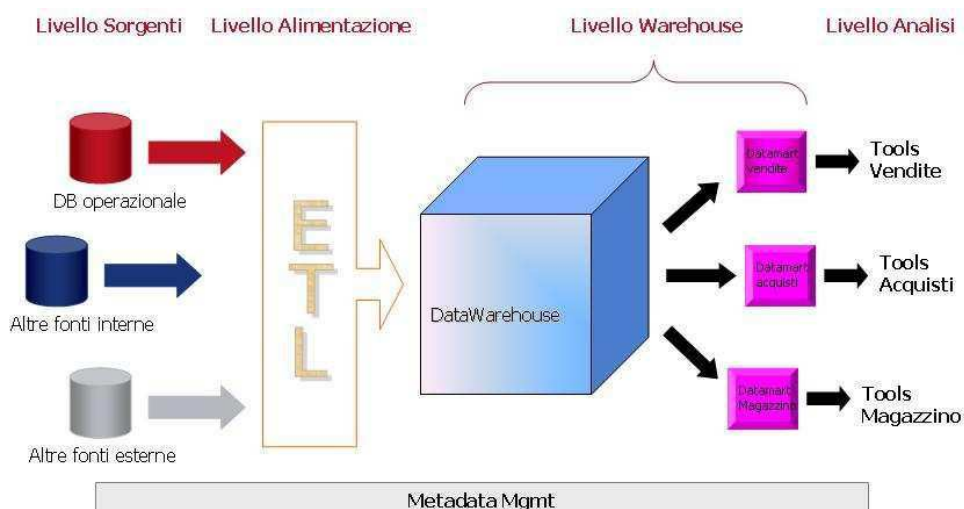
dei processi analitici è significativamente differente rispetto alla raccolta dei requisiti dei processi transazionali. Chi progetta un data warehouse, infatti, deve capire perfettamente i fattori chiave che guidano il business, in modo da inquadrare inequivocabilmente quali siano questi requisiti. Solo a questo punto sarà possibile trasformarli in linee guida del progetto. Il processo di definizione delle specifiche rappresenta le fondamenta per i successivi tre percorsi paralleli di attività:

- *le tecnologie*. Comprende le attività di progetto dell'architettura tecnologica e la selezione dei relativi prodotti commerciali;
  - *i dati*. In questo percorso, rientrano le attività di progettazione concettuale, logica e fisica della base dei dati e la definizione dei processi di caricamento delle informazioni nel data warehouse;
  - *le applicazioni utente*. Il compito di questo ramo è quello di raccogliere le specifiche delle procedure da costruire e di procedere al relativo sviluppo.
- **modello multidimensionale, Progetto Fisico e ETL.**

La definizione dei requisiti permette di definire il modello dei dati necessario per soddisfare le esigenze richieste dagli utenti. Abbiamo visto che la progettazione del modello dei dati differisce notevolmente rispetto all'approccio adottato per i sistemi transazionali. Questoprocesso si compone dei seguenti passi:

- *costruzione della matrice delle dimensioni*. Si inizia costruendo questa matrice le cui righe riportano i processi chiave del nostro business mentre le colonne descrivono la loro dimensionalità, cioè il numero e il tipo delle dimensioni chiamate in causa;
- *analisi delle fonti OLTP*. Successivamente, si conduce una dettagliata analisi dei vari sistemi OLTP che fungono da sorgenti per il nostro data warehouse. Quest'attività è necessaria per stabilire le specifiche del processo di caricamento dei dati;
- *modello Multidimensionale*. Considerando l'insieme dei requisiti raccolti nel punto precedente viene sviluppato il modello multidimensionale del business. Questo modello permette di specificare i fatti, le dimensioni, gli attributi e le gerarchie del business;

- *progetto Logico*. La progettazione logica del database dipende dal tipo di tecnologia adottata per la memorizzazione dei dati. Anche in questo caso, affronteremo il discorso nel capitolo successivo.
- **Il progetto fisico del database** focalizza l'attenzione sulla definizione delle strutture fisiche necessarie per supportare il progetto logico definito nel punto precedente. Gli elementi principali di questa attività sono:
  - la definizione di una convenzione standard per i nomi. L'uso di standard è essenziale per rendere il progetto chiaro e leggibile;
  - la configurazione dell'ambiente del database. Queste attività permettono di effettuare il tuning del database in modo da ottimizzare le sue prestazioni;
  - la definizione delle strategie di indicizzazione e di partizionamento. Anche questa tecnica è volta al raggiungimento delle prestazioni massime del sistema.
- **il processo di ETL**, si compone di tre processi fondamentali: l'estrazione dei dati dalle varie fonti OLTP, la loro trasformazione e il caricamento nel data warehouse. Il processo di trasformazione è particolarmente delicato perché deve garantire la qualità dei dati del warehouse. Questa qualità è fondamentale affinché le informazioni contenute nel DSS siano il più credibili possibile. Generalmente vengono realizzati due processi ETL: il primo utilizzato per popolare la prima volta il data warehouse, il secondo per caricare incrementalmente e ad intervalli regolari le nuove informazioni non ancora presenti nel data warehouse. Vedremo nel capitolo 4 i dettagli di tale processo.



Una volta concluso quest'ultimo processo, prima di rendere le informazioni disponibili all'utente, bisogna procedere al controllo della qualità dei dati caricati nel data warehouse. Questo controllo può essere attuato attraverso l'esecuzione di un particolare report, contenente varie statistiche relative all'intero set dei nuovi dati caricati.

Un progetto DSS richiede l'integrazione di numerose tecnologie. Per progettare l'architettura tecnologica bisogna considerare simultaneamente tre fattori fondamentali:

- i requisiti del business. Stabiliscono le linee guida per quanto riguarda le prestazioni e la sofisticazione del sistema tecnologico;
- l'architettura tecnologica in essere. Questo fattore è fondamentale per capire l'integrazione delle nuove tecnologie con quelle esistenti;
- le strategie pianificate nell'area tecnologica. Dettano, sostanzialmente, i budget disponibili per l'acquisizione di nuove tecnologie.

Utilizzando i risultati del punto precedente, bisogna procedere alla selezione dei vari componenti dell'architettura stabilita. Questo traduce nell'avalutazione dei seguenti punti chiave:

- Le piattaforme hardware, necessarie per l'implementazione del sistema DSS.
- Il database management system, che dovrà contenere e gestire la base dati.
- Gli strumenti ETL, necessari per caricare le informazioni nel DSS.
- Gli strumenti di accesso, utilizzati per visualizzare e manipolare i dati contenuti nel warehouse.

Per ognuno di questi componenti bisogna procedere alla valutazione tecnica delle varie offerte commerciali, in modo da selezionare il prodotto che meglio si adatta alle nostre esigenze.

Una volta che il prodotto è stato selezionato deve essere *installato* e accuratamente testato prima di essere integrato definitivamente nel sistema DSS.

- **Specifiche & Sviluppo utente.** Le specifiche delle applicazioni sono composte sostanzialmente da documenti che descrivono il modello dei report richiesti, i parametri utente e i calcoli richiesti. Queste specifiche permettono di stabilire

un'unica visione della applicazione da sviluppare, sia all'utente finale, sia al team incaricato alla sua implementazione;

- seguendo le specifiche raccolte, lo sviluppo di una applicazione consiste nella costruzione dei vari report richiesti. Spesso, queste applicazioni vengono realizzate utilizzando strumenti avanzati per l'accesso ai dati che permettono di ottenere un alto livello di produttività del team di sviluppo. Inoltre, le applicazioni così costruite, mettono a disposizione semplici strumenti all'utente finale per modificare, secondo le proprie esigenze, i report esistenti;
- **Integrazione.** Con questo termine si rappresenta la convergenza e la perfetta integrazione delle tecnologie, dei dati e delle applicazioni utente, in modo che siano pienamente utilizzabili dai desktop dell'utente. Questa integrazione deve essere considerata anche durante il processo formativo degli utenti finali, in modo da renderli in grado di sfruttare tutte le potenzialità del sistema DSS;
- **Manutenzione e crescita.** L'attività di manutenzione coinvolge una serie di processi:
  - Il supporto e la formazione degli utenti. Queste sono attività fondamentali per permettere l'appropriato utilizzo delle nuove risorse disponibili;
  - La manutenzione delle applicazioni. Poiché i requisiti di un business sono molto dinamici, le richieste effettuate dagli utenti cambiano nel tempo. Di conseguenza, è necessario modificare le applicazioni originali per seguire di pari passo questi cambiamenti;
  - La supervisione del sistema DSS, mantenendo sotto controllo il suo comportamento e le sue prestazioni. La realizzazione di una rete di comunicazione. La facilità di comunicazione, infatti, permette una miglior integrazione fra tutti i partecipanti del progetto.
- seguendo questo tipo di approccio nella costruzione di un sistema DSS, quello che si ottiene è un sistema modulare e aperto in grado di crescere ed evolversi nel tempo, seguendo i nuovi requisiti dettati dall'evoluzione del business e degli utenti;

- **Gestione del progetto.** La gestione del progetto consiste nel controllare che le attività coinvolte nel ciclo di vita procedano rispettando i tempi previsti e rimangano sincronizzate fra loro. Il raggiungimento di questo obiettivo richiede di tenere conto di vincoli contrastanti come:
  - Tempo, costi e qualità, (migliorare la qualità comporta in genere un aumento di costi e tempi; la riduzione dei tempi può causare un aumento dei costi o una riduzione della qualità);
  - Stakeholder con necessità e aspettative in conflitto, (il finanziatore è interessato a costi bassi, gli utenti sono interessati a prestazioni elevate);
  - Bisogni inespressi, bisogni espressi non necessari e aspettative irrealistiche (specie nei progetti innovativi, le aspettative sono spesso non coerenti con il budget; inoltre l'impiego di tecnologie non ancora diffuse, spesso non consente di prevedere il livello di qualità del risultato).

## Capitolo 3

# IL MODELLO MULTIDIMENSIONALE

---

La modellazione multidimensionale è la tecnica di progettazione logica più utilizzata per un sistema di supporto alle decisioni. L'idea base è di vedere i dati come punti in uno spazio le cui dimensioni corrispondono ad altrettante possibili dimensioni di analisi: ciascun punto, rappresentativo di un evento accaduto nell'azienda viene descritto tramite un insieme di misure di interesse per il processo decisionale.

L'analisi di questi "eventi" avviene su dati rappresentati in forma multidimensionale, ovvero organizzati mediante i seguenti concetti:

- **Fatto:** concetto sul quale concentrare l'analisi;
- **Misura:** una proprietà atomica o misura di un fatto da analizzare;
- **Dimensione:** prospettiva rispetto alla quale effettuare l'analisi.

### 3.1 - Elementi di un modello multidimensionale

Di seguito verranno riportati e descritti i diversi elementi di un modello multidimensionale, in particolare:

- dimensioni;
- gerarchie e aggregati;
- fatti e misure;
- il cubo.

Il concetto di dimensione ha dato origine alla metafora del cubo per la rappresentazione dei dati multidimensionali. Secondo questa metafora gli eventi corrispondono a celle di un cubo i cui spigoli rappresentano le dimensioni di analisi. Ogni cella del cubo contiene valori per ogni singola misura.

### 3.1.1 - Dimensioni

Gli analisti sono abituati a ragionare in termini di dimensioni e misure (non schemi, tabelle e record). Possiamo definire il termine dimensione come "una prospettiva di analisi dei dati di un business". Attraverso le dimensioni, sarà quindi possibile visualizzare e studiare gli stessi tipi di dati secondo diversi punti di vista. Le dimensioni sono attributi consistenti di dati, rappresentano in pratica le coordinate di un fatto percorribili ai fini analitici, il numero è la "grandezza" di una dimensione con la complessità e la comprensione del business.

La domanda che ci si potrebbe porre è: come si identifica se un campo numerico è un fatto o un attributo di una dimensione?

Se è una misura che varia continuamente nel tempo allora è un fatto, ad esempio l'analisi di costo di un prodotto nel tempo.

Se è una descrizione discreta di qualcosa che è ragionevolmente costante è un attributo di una dimensione, ad esempio costo di un prodotto come informazione descrittiva.

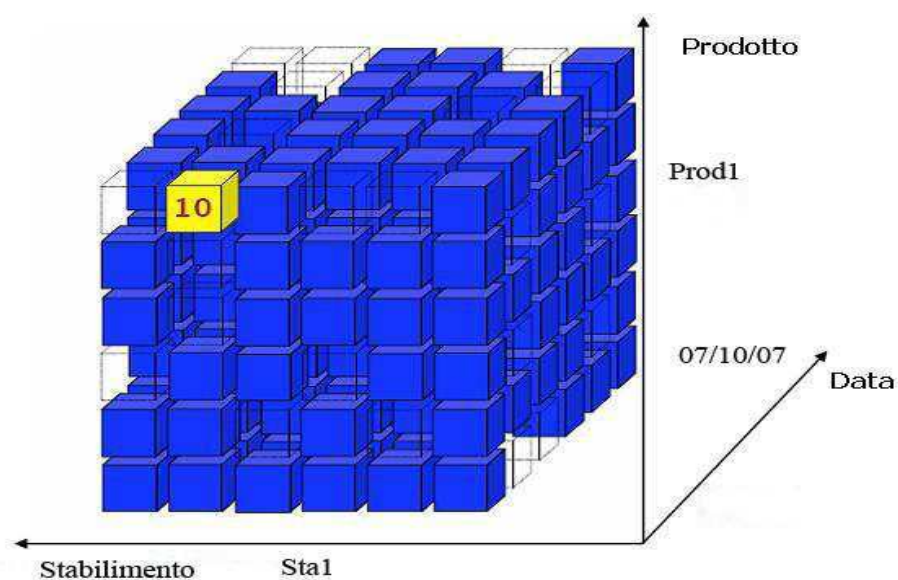


Figura 3.1. Cubo a 3 dimensioni

La parola dimensione nasce dal fatto che, organizzando i dati secondo differenti prospettive, questi possono essere rappresentati come detto mediante un cubo avente



tante dimensioni quante sono le prospettive di analisi stabilite. Chiariamo questo concetto con un esempio.

Consideriamo un banale sistema di analisi della produttività su due differenti prodotti e su stabilimenti distinti. I responsabili del controllo di gestione sono interessati a controllare le quantità prodotta di questi prodotti nei loro tre stabilimenti. Per risolvere questo semplice problema è sufficiente costruire un cubo organizzato secondo tre dimensioni: una dimensione Prodotto, costituita dai due prodotti Prod1 e Prod2. Una dimensione Stabilimento, costituita dagli stabilimenti Sta1, Sta2 e Sta3. Una dimensione Data, costituita dalle date di rilevazione dei Kg.

Il cubo di dati che si ottiene è rappresentato in figura 3.1. Come possiamo notare, ogni cella è identificata univocamente da tre coordinate, costituite da elementi delle rispettive dimensioni, e contiene le quantità prodotte di un determinato prodotto in un determinato stabilimento su un determinato giorno. Possiamo aumentare la complessità del nostro cubo aggiungendo nuove dimensioni.

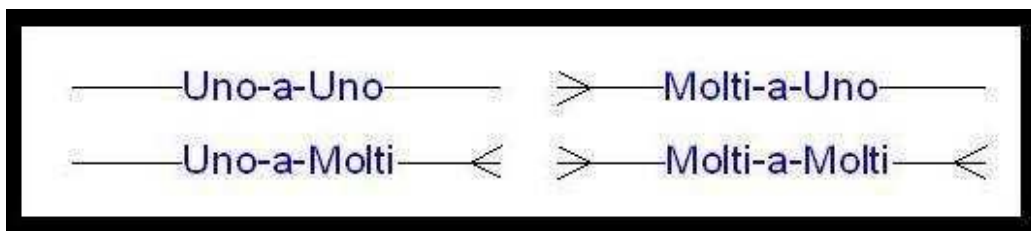


Figura 3.2. Tipologie di relazioni

Nel modello multidimensionale, le relazioni vengono rappresentate mediante linee, aventi come estremo una forbice disposta verso l'attributo che partecipa alla relazione con cardinalità multipla. La figura 3.2 illustra quanto detto.

### 3.1.2 - Gerarchie ed aggregati

L'idea delle gerarchie è di aggregare automaticamente i dati di interesse quando ci si focalizza su un livello: se ci concentriamo su "Mese" i fatti rappresentano i Kg totali per ogni mese. In realtà possiamo concentrarci su diversi livelli della gerarchia in dimensioni diverse: la produzione mensile per stabilimento di ogni prodotto.

Una gerarchia tipica può essere: Famiglia prodotto, Sottofamiglia prodotto e Prodotto, le gerarchie inoltre mostrano i livelli all'interno delle dimensioni, ogni dimensione può avere gerarchie multiple.

### 3.1.3 - Fatti e misure

I fatti sono gli oggetti che influenzano il processo decisionale in quanto sono fatti del mondo aziendale. Ogni singolo fatto può essere organizzato lungo una o più dimensioni. L'insieme delle dimensioni su cui viene organizzato si chiama dimensionalità di quel fatto. Nel paradigma multidimensionale, ogni singolo fatto può essere pensato come un cubo di dati aventi tanti spigoli quante sono le sue dimensioni. Un fatto deve sempre avere degli aspetti dinamici che lo rendano interessante per il processo decisionale. Inoltre può essere caratterizzato da una o più misure. Una misura è una proprietà numerica di un fatto e ne descrive un aspetto quantitativo di interesse per l'analisi. Ad esempio per il fatto Produttività le misure tipiche possono essere la i Kg prodotti o le ore lavorate.

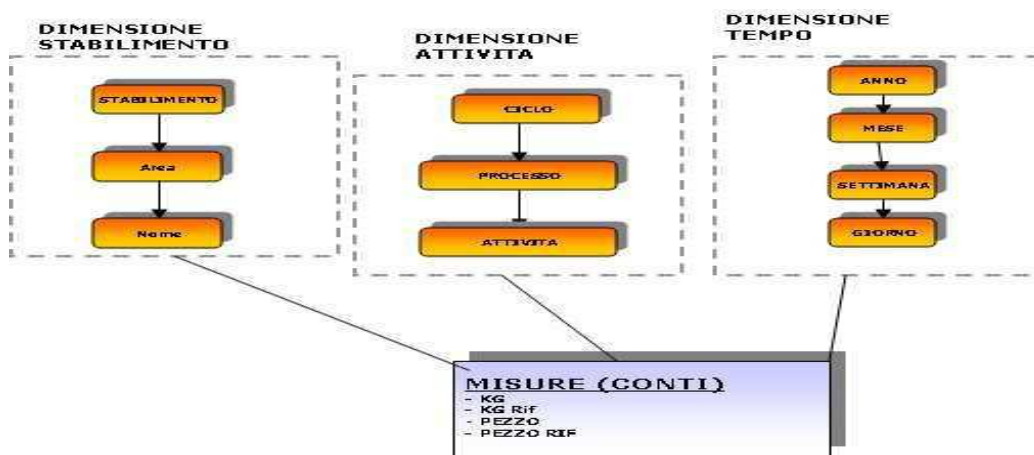


Figura 3.3. Un semplice modello multidimensionale

Nel modello multidimensionale, i fatti vengono rappresentati mediante dei cubi e la loro dimensionalità viene rappresentata da frecce che li collegano con le rispettive dimensioni. In figura 3.3 viene rappresentato un semplice modello multidimensionale. In questo esempio la granularità dei fatti è definita dalla tripla (Stabilimento, Attività, Tempo).

## 3.2 - Implementazione del modello multidimensionale

Una volta definito il modello multidimensionale del business, bisognerà implementarlo fisicamente su un sistema. Questo comporta la scelta della tecnologia di supporto, il disegno logico del database e la sua realizzazione fisica. Nei prossimi paragrafi verranno introdotte le due differenti tecnologie per la memorizzazione dei dati e il relativo progetto logico del database.

### 3.2.1 - Database Multidimensionale

Il concetto su cui si basa questo sistema è quella di realizzare una corrispondenza biunivoca fra come le persone vedono i dati del loro business e come questi dati vengono memorizzati in un sistema informativo. Utilizzando questa tecnologia, la progettazione logica del database è praticamente inesistente, in quanto esiste una relazione uno-a-uno fra il modello multidimensionale e il disegno del database. Infatti, questo sistema software permette di definire gli stessi oggetti che abbiamo visto nel modello multidimensionale: dimensioni, attributi, relazioni, livelli, gerarchie e fatti.

La struttura fondamentale di memorizzazione di un database multidimensionale è l'array scelto per sostituire le tabelle relazionali, da cui per altro si differenziano per alcuni aspetti fondamentali che ora andremo ad analizzare.

---

ATTIVITA'	STABILIMENTO	KG
1.00	1001	6
1.00	1002	5
1.00	1003	4
2.00	1001	3
2.00	1002	5
2.00	1003	6
3.00	1001	4
3.00	1002	3
3.00	1003	2

Figura 3.4. Una tipica tabella relazionale

I dati in un database relazionale vengono memorizzati sotto forma di tabelle, collegate fra loro attraverso chiavi. All'interno di ogni tabella le informazioni sono organizzate in

record. Ogni record corrisponde ad una riga della tabella ed è suddiviso in campi che costituiscono le colonne della tabella stessa. In figura 3.4 è rappresentato un esempio di tabella relazionale.

Come possiamo notare, la tabella contiene nove record, ciascuno dei quali è composto da tre campi : attività, stabilimento, Kg. Per rappresentare le stesse informazioni, un database multidimensionale utilizzerà l'array bidimensionale mostrato in figura 3.5.

Guardando la figura , si può vedere immediatamente che ci sono esattamente due dimensioni formate da tre elementi ciascuna, cosa molto più difficile da notare su una tabella, soprattutto al crescere delle dimensioni e degli elementi. Inoltre, l'array multidimensionale presenta un più alto livello di organizzazione dei dati che permette una visualizzazione ed una manipolazione molto più logica per l'utente finale.



Figura 3.5. Il corrispondente array bidimensionale

Questo tipo di organizzazione permette anche grossi vantaggi in termini di performance. Consideriamo ad esempio un cubo di 3 dimensioni contenenti ciascuna 10 elementi. L'equivalente tabella relazionale deve contenere  $10 \times 10 \times 10 = 1000$  records. Se un utente vuole conoscere il valore di una determinata cella, il sistema relazionale deve scandire, nel caso peggiore, tutti i 1000 records, mentre il sistema multidimensionale deve cercare solo lungo le tre dimensioni per identificare le coordinate del dato. Questo si traduce nella scansione di un massimo di 30 elementi.

Una nota dolente per i sistemi multidimensionali è rappresentata, invece, dalla sparsità. Infatti, se i vari record di una tabella relazionale non godono di interrelazioni, cioè i campi corrispondenti alle dimensioni non sono composti da pochi elementi ripetuti,

quello che si ottiene, mappandola su un database multidimensionale, è un array sparso (molte celle sono vuote) poco prestante in termini di performance e utilizzo di spazio. Questa tipologia di sistemi è stata concepita con l'intento di massimizzare le performance dei processi analitici realizzando una corrispondenza fra come l'utente pensa ed utilizza i dati di un business e come il sistema li gestisce effettivamente.

Esistono numerose caratteristiche proprie di un database multidimensionale (MDDDB), vediamole brevemente.

**Analisi multidimensionali** - In particolare è banale immaginare come le operazioni di rotazione e di selezione avvengano naturalmente su questi database senza la necessità di riorganizzare i dati e con ottime prestazioni.

**Gestione delle gerarchie** - Le gerarchie sono viste come differenti livelli di aggregazione collegati attraverso relazioni uno-a-molti. I dati aggregati ai vari livelli vengono memorizzati nel database multidimensionale in modo da non dover essere calcolati a runtime. Per ottenere lo stesso effetto su un sistema relazionale, occorre rilanciare la query ogni volta che cambia il livello della gerarchia.

**Linguaggio di interrogazione** - L'alto grado di strutturazione che si ottiene con un MDDDB si traduce anche in un semplice ed efficiente linguaggio di interrogazione. Non solo il linguaggio è più intuitivo, ma l'output della query è molto più significativo per l'utente essendo in formato multidimensionale.

**Accesso Multiutente** - Uno dei vantaggi chiave di un MDDDB consiste nella possibilità di gestire meglio l'accesso concorrente alle stesse informazioni da parte di più utenti. Ogni utente è in grado di definire agevolmente la propria vista dei dati e di compiere le relative operazioni senza interferire con gli altri.

**Calcoli Multidimensionali** - Un MDDDB permette di eseguire calcoli aritmetici sui dati in modo semplice ed efficiente.

**La Dimensione Tempo** - In molti MDDDB il tempo è una dimensione predefinita, con gerarchie predefinite. Quindi esistono funzioni preconfezionate che operano su serie temporali in modo molto efficiente.

**Sparsità** - Gli sparse array sono cubi con poche celle popolate. Le celle vuote, cioè quelle che non contengono dati, sono contrassegnate da uno speciale valore NA

(NotAvailable). Un buon MDDDB deve trattare in modo intelligente i valori NA utilizzando tecniche di compressione del DB che permettano di ottenere ottime performance riducendo lo spazio di memorizzazione richiesto.

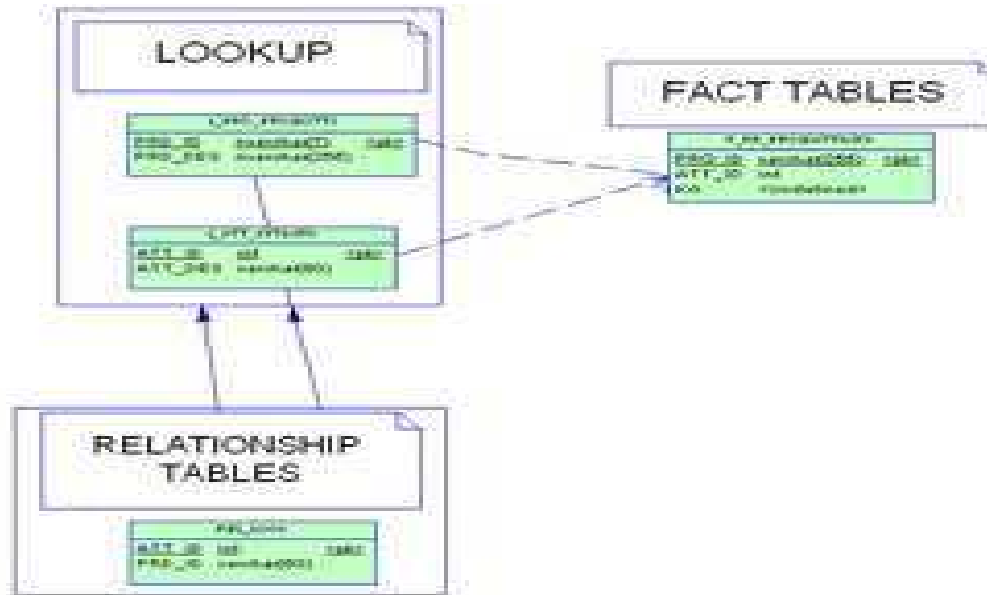


Figura 3.6. La struttura delle diverse tipologie di tabelle relazionali usate per rappresentare gli oggetti di un modello multidimensionale.

### 3.2.2 - Lookup, Relationship e Fact

Esistono differenti tipologie di tabelle corrispondenti ai vari oggetti del modello multidimensionale :

- **Lookup Tables.** Sono costituite dalla chiave o l'identificatore di un elemento e la relativa descrizione. Queste tabelle permettono quindi di rappresentare gli attributi e i livelli di una dimensione, sono utilizzate principalmente per contenere le anagrafiche.
- **Relationship Tables.** Sono composte dalle chiavi di due o più attributi, in modo da rappresentare le relazioni esistenti fra gli attributi. Queste tabelle sono strettamente necessarie nel caso di relazioni multi-a-molti, mentre generalmente non vengono utilizzate negli altri casi.
- **Fact Tables.** Rappresentano i fatti del business (es: Driver di costo), contengono le chiavi degli attributi delle dimensioni.

Il procedimento di base per lo sviluppo del modello inizia definendo una lookuptable per ogni attributo del modello multidimensionale. Successivamente vengono stabiliti i campi di queste tabelle che consistono, generalmente, in una chiave e in una descrizione opzionale e in altri campi che caratterizzano la dimensione.

Durante la progettazione è fondamentale definire una nomenclatura standard. Ad esempio aggiungendo il suffisso ID per le chiavi primarie (PK) e per le foreignkey (FK), il suffisso Desc per tutti i campi descrittivi, il prefisso Lookup per i nomi delle lookuptable e il prefisso Fact per i nomi delle facttable.

Nel caso di relazioni molti a molti si definiscono le RelationshipTables. Come detto queste tabelle contengono al loro interno le combinazioni valide delle chiavi degli attributi che partecipano alla relazione.

La combinazione delle dimensioni porta alla definizione delle facttable che conterranno tante chiavi quante sono le dimensioni e tanti campi quanti sono i fatti associati a queste dimensioni. La primarykey della facttable sarà una chiave composta dagli identificatori degli attributi partecipanti. Ogni chiave di attributo inserita nella facttable permette di identificare la rispettiva dimensione e il livello di aggregazione dei dati.

Il risultato di questo procedimento è uno schema normalizzato che rappresenta il modello multidimensionale di partenza. Questo schema sarà quindi costituito da una o più lookuptables per ogni dimensione, contenenti le informazioni relative ai vari attributi, e da una o più facttables.

### **3.2.3 - I database relazionali**

Utilizzando un database multidimensionale, non esiste la necessità di definire un progetto logico della base dati perché sussiste una corrispondenza biunivoca col modello multidimensionale.

La situazione si presenta decisamente differente se scegliamo come supporto di memorizzazione un sistema relazionale. In questo caso, infatti, occorre utilizzare una tecnica di progettazione logica espressamente pensata per rappresentare nel miglior modo possibile strutture dati multidimensionali. Il risultato di questo processo sarà uno schema relazionale composto da differenti tipi di tabelle che rappresentano gli oggetti caratteristici del modello multidimensionale: una tabella dei fatti contenente le misure richieste e una tabella per ogni dimensione che conterrà tutti

gli elementi della gerarchia.

Il modello finale che si ottiene è detto Star Schema proprio per la sua caratteristica forma a stella con la tabella dei fatti al centro e le tabelle delle dimensioni attorno, la figura 3.7 ne riporta un esempio.

La caratteristica dello star schema è di essere pesantemente denormalizzato e di presentare, quindi, una grande ridondanza dei dati. L'operazione di denormalizzazione permette però di ridurre il numero di join richiesti per soddisfare una interrogazione.

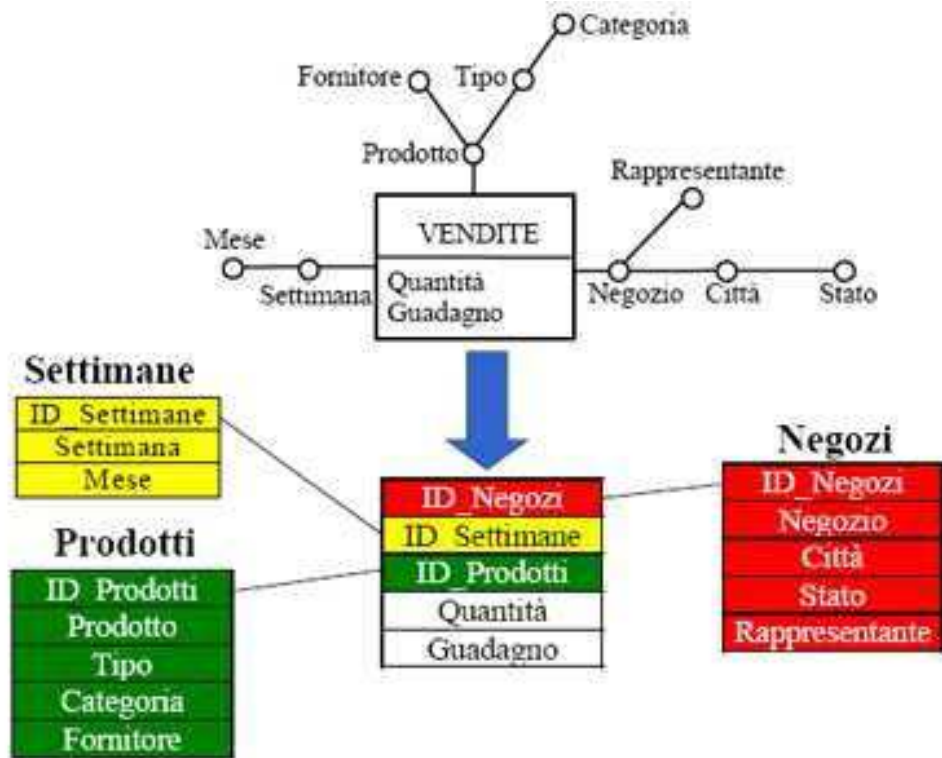


Figura 3.7. Lo star schema

Questo schema viene realizzato con alcune varianti in base alle necessità del business. Innanzi tutto è possibile mantenere nella facttable solo i dati atomici o anche quelli aggregati riducendo i tempi di roll-up ma aumentando il numero di record memorizzati. Inoltre, lavorando sulle dimensiontable si ottengono degli schemi via via più normalizzati detti Snowflake Schema che riducono le dimensioni delle varie tabelle aumentando però il numero di join necessari. In pratica si implementa per ogni livello della gerarchia una diversa tabella; i vari livelli sono collegati con l'uso di foreignkey e solo il livello più basso è legato alla facttable che dovrà quindi mantenere sempre dati non aggregati.



### 3.2.4 - Star Schema vs Snowflake Schema

#### Star Schema:

La caratteristica dello star schema è di essere pesantemente denormalizzato e di presentare, quindi, una grande ridondanza dei dati. Questo schema utilizza una sola lookuptable per ogni dimensione, contenente al suo interno tutte le informazioni relative ai suoi vari attributi. L'operazione di denormalizzazione permette, quindi, di ridurre il numero di join richiesti per soddisfare una interrogazione.

Per quanto riguarda, invece, la facttable, la sua chiave primaria è composta dalle chiavi generiche associate ad ogni dimensione (SETTIMANE\_ID, PRODOTTI\_ID, NEGOZI\_ID). Al suo interno, quindi, non esiste l'indicazione del livello di aggregazione dei dati memorizzati. Questa informazione la si trova solamente nelle tabelle di lookup. E' possibile adottare due diverse tipologie di fact table :

- **in-table Aggregation.** La tabella dei fatti può contenere dati aggregati per ogni possibile combinazione dei livelli di aggregazione;
- **atomic data only.** Una facttable può contenere solamente dati appartenenti allo stesso livello di aggregazione.

#### Snowflake Schema:

La caratteristica principale di questo tipo di schema è quello di fare un basso uso della ridondanza dei dati presentando, quindi, una maggior normalizzazione.

Nello snowflake schema esiste una lookuptable per ogni attributo o livello di una dimensione. Questa tabella conterrà una chiave primaria, per identificare i vari elementi, una descrizione e le foreignkey per rappresentare le relazioni esistenti fra i vari livelli. In questo schema, la facttable contiene sempre dati appartenenti allo stesso livello di aggregazione. Non è quindi possibile avere in tableaggregation. La chiave della facttable è composta dalle singole chiavi dei livelli delle diverse dimensioni. Esisterà una chiave per ogni dimensione che interseca la facttable, la quale permetterà di stabilire il livello di aggregazione dei dati e la dimensione partecipante.

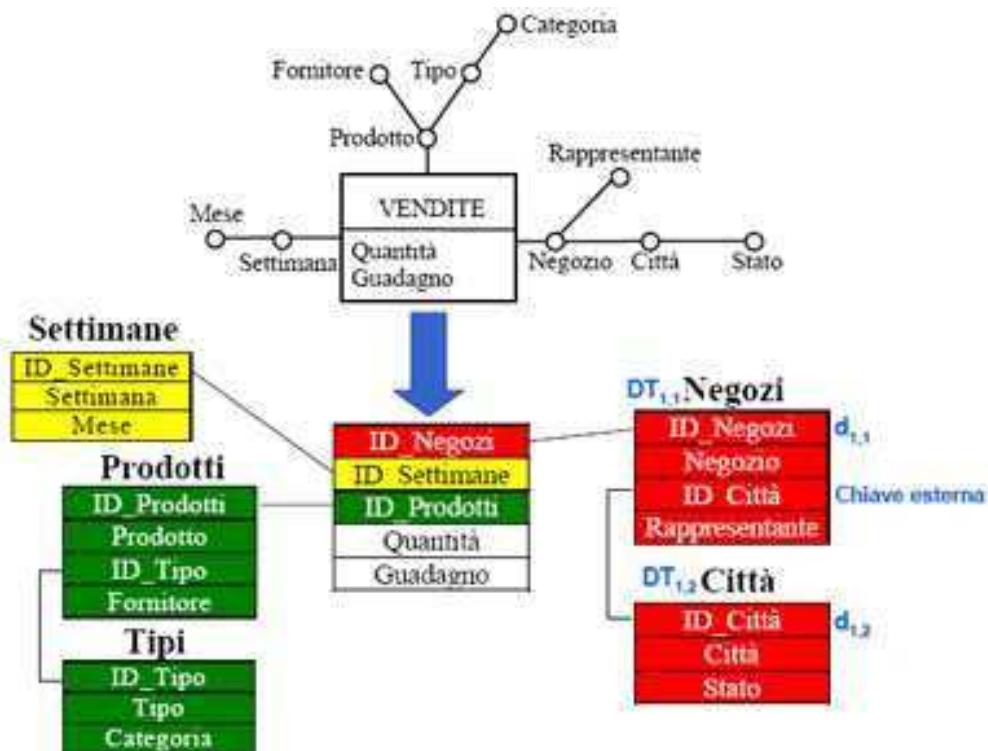


Figura 3.8. Lo snowflake schema

## Star Schema vs Snowflake Schema

In conclusione i vantaggi derivanti dall'adozione di uno star schema sono: basso numero di tabelle, alta comprensibilità dello schema e semplice generazione delle query SQL.

I problemi principali che affliggono questo disegno sono, invece: l'impossibilità di rappresentare le relazioni multi-a-molti e la necessità di cambiare la struttura delle tabelle di lookup al variare del numero di attributi e/o dei livelli all'interno di una dimensione.

Invece i principali vantaggi che si ottengono adottando uno snowflake schema sono: alta flessibilità e scalabilità, supporto delle relazioni multi-a-molti e facttable contenenti dati omogenei.

Gli aspetti negativi di questa soluzione sono invece: alto numero di tabelle, schemapiù complicato e meno leggibile, difficoltà di generazione delle istruzioni SQL per risolvere una interrogazione.

### 3.3 - Operazioni multidimensionali

Per navigare il cubo multidimensionale esistono differenti operazioni che forniscono informazioni su punti di vista differenti:

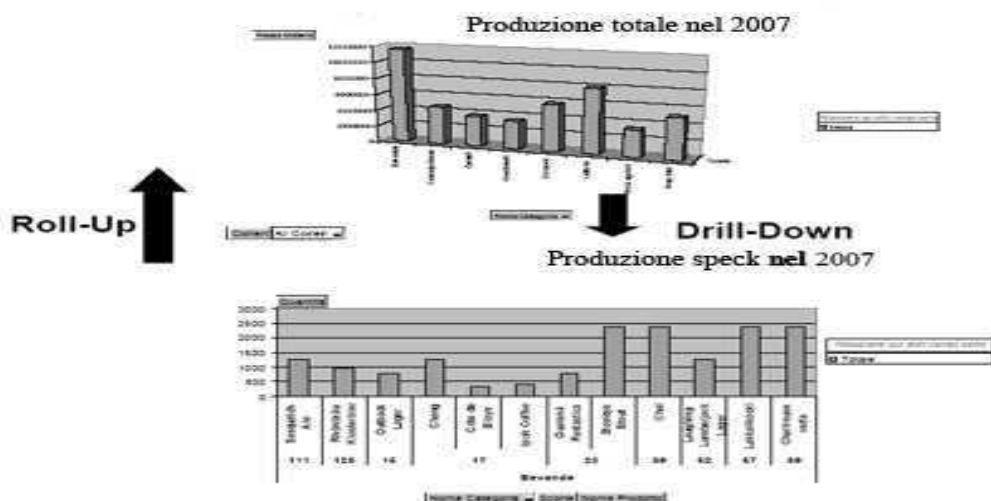


Figura 3.9. Roll-up e Drill-down

- **Roll-up:** aggrega i dati (rispetto all'interrogazione corrente), ovvero mostra dati ad un maggior livello di aggregazione, ad esempio per un particolare prodotto è possibile ottenere i Kg di una singola attività svolta in un determinato stabilimento;
- **Drill-down:** disaggrega i dati, ovvero mostra dati a un minor livello di aggregazione;
- **Drill-across:** combina i dati associati a più fatti.;
- **Slice&Dice:** seleziona e proietta i dati. In pratica consente la selezione e la proiezione dei dati;
- **Pivoting** o **Rotating:** re-orienta i dati. In pratica riorganizza(ruota) il cubo, consentendo un cambiamento nella modalità di presentazione con l'obiettivo di analizzare le stesse informazioni sotto diversi punti di vista.



Figura 3.10. Slice e Dice.

## Capitolo 4

# DATA WAREHOUSE

---

La caratteristica fondamentale che differenzia i sistemi OLTP dai sistemi DSS è il progetto del database. Infatti, un database analitico, è pensato per gestire grandi volumi di dati, tipicamente in sola lettura, al fine di produrre le informazioni necessarie ai responsabili di un'organizzazione, per prendere le decisioni. Chiameremo allora Data Warehouse "il database analitico che costituisce le fondamenta di un sistema di supporto alle decisioni".

### 4.1 - Requisiti del Data Warehouse

Gli obiettivi del DW possono essere ottenuti semplicemente camminando tra gli uffici di una azienda e ponendo attenzione alle problematiche tipiche dei manager. Tipicamente emergeranno problemi di questo tipo: "Abbiamo montagne di dati ma non riusciamo ad accedervi"; "Mostratemi solo ciò che è importante"; "Vogliamo che la gente utilizzi le informazioni per supportare i processi decisionali". I temi del management aziendale vanno trasformati in requisiti del Data Warehouse; possiamo dunque affermare che un DW:

- deve rendere le informazioni di un'azienda facilmente accessibili. Gli strumenti con cui accedervi devono essere estremamente intuitivi, in maniera da garantire facilità di interrogazione anche da parte di un'utente aziendale, tipicamente privo di conoscenze tecniche informatiche;
- deve presentare le informazioni dell'azienda in modo coerente. La qualità dei dati deve essere garantita, ovvero i dati devono essere esaurienti e completi;

- deve essere adattabile alle modifiche ed elastico. Le necessità degli utenti, le condizioni aziendali, i dati e le tecnologie sono tutti soggetti a cambiamenti. Il DW deve essere progettato in modo da poter gestire questi cambiamenti inevitabili.;
- deve proteggere le informazioni in modo sicuro. Data la natura strategica delle informazioni contenute nel DW, è necessario un efficiente meccanismo di security;
- deve servire come base per prendere le decisioni migliori. L'output di un DW corrisponde ad una determinata decisione da intraprendere.

Come messo in evidenza da questi requisiti, il data warehouse richiede molto più di un eccellente Database Admin, in quanto questa è un'area trasversale tra l'informatica e il management aziendale. La figura professionale richiesta deve comportarsi come un ibrido tra DBA e MBA.

## **4.2 - L'architettura di un Data Warehouse**

La figura 4.1 mostra il flusso, l'organizzazione e l'utilizzo dei dati all'interno di un data warehouse. Il processo di Data Integration si occupa di estrarre i dati dalle varie sorgenti OLTP presenti, di trasformarli e organizzarli opportunamente secondo regole ben definite, e di caricarli all'interno del data warehouse. Questo processo viene anche chiamato Extraction, Transformation e Loading.

Il data warehouse rappresenta quindi una sorgente di dati secondaria, popolata dai sistemi OLTP esistenti e da altre fonti dati esterne, che consistono, ad esempio, in compagnie specializzate nella fornitura di dati di vario genere. Quella che a prima vista potrebbe sembrare una duplicazione delle informazioni, è invece una essenziale raccolta ben organizzata di tutte le informazioni presenti all'interno di una organizzazione, al fine di facilitare i processi di analisi e di pianificazione.

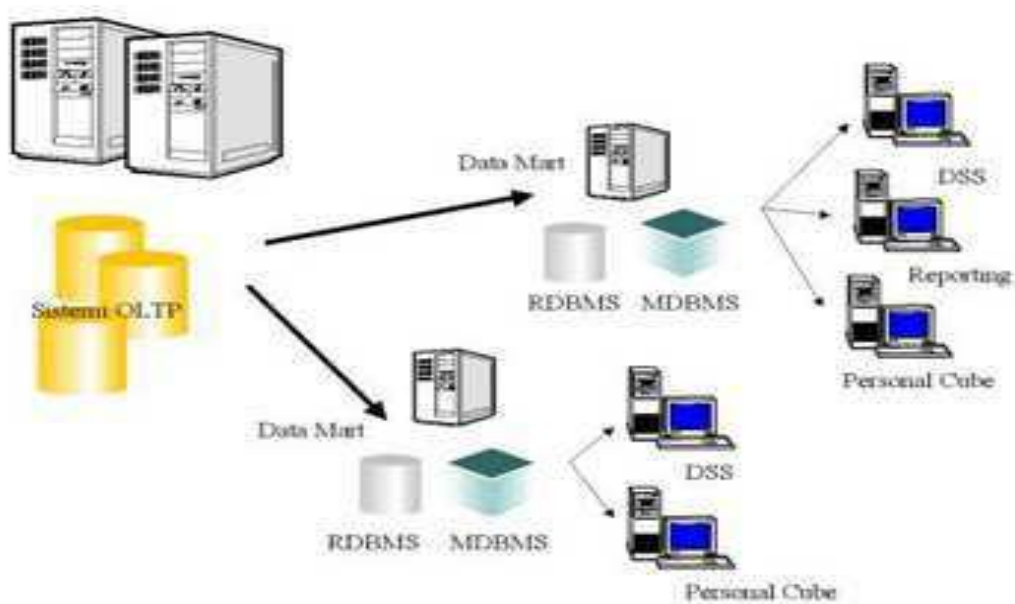


Figura 4.1. L'architettura di un Data Warehouse

## 4.2.1 - Il Data Mart

Un data mart è un raccoglitore di dati specializzato in un particolare soggetto. Un data mart contiene un'immagine dei dati che permette di formulare strategie sulla base degli andamenti passati. Normalmente si colloca a valle di un data warehouse più globale ed è alimentato a partire da esso, di cui costituisce, in pratica, un estratto. Detto in termini più tecnici, un data mart è un sottoinsieme logico o fisico di un data warehouse di maggiori dimensioni. La differenza fondamentale consiste nel fatto che la creazione del data warehouse avviene in maniera generalizzata per poi venire incontro alle specifiche esigenze, mentre il data mart viene generalmente creato per venire incontro ad un'esigenza specifica e già determinata.

La necessità di creare un sistema separato per il data mart rispetto al data warehouse può riassumersi nelle seguenti motivazioni:

- la necessità di utilizzare un diverso schema;
- migliorare le performance;

- garantire una maggiore sicurezza dovendo autorizzare l'accesso ad un insieme minore di dati.

Una generica organizzazione può scegliere di implementare un progetto di DW con data mart, secondo sue differenti approcci: top-down e bottom-up sono strategie di elaborazione dell'informazione e di gestione delle conoscenze, riguardanti principalmente il software e, per estensione, altre teorie umanistiche e teorie dei sistemi. Nel modello top-down è formulata una visione generale del sistema senza scendere nel dettaglio di alcuna delle sue parti. Ogni parte del sistema è successivamente rifinita aggiungendo maggiori dettagli dalla progettazione. Ogni nuova parte così ottenuta può quindi essere nuovamente rifinita, specificando ulteriori dettagli finché la specifica completa è sufficientemente dettagliata da validare il modello. Il modello top-down è spesso progettato con l'ausilio di scatole nere che semplificano il riempimento ma non consentono di capirne il meccanismo elementare.

In contrasto con il modello top-down c'è la progettazione bottom-up, nella quale parti individuali del sistema sono specificate in dettaglio. Queste parti vengono poi connesse tra loro in modo da formare componenti più grandi, che vengono a loro volta interconnessi fino a realizzare un sistema completo. Le strategie basate sul flusso informativo bottom-up sembrano potenzialmente necessarie e sufficienti dato che sono basate sulla conoscenza di tutte le variabili che possono condizionare gli elementi del sistema.

Indipendentemente dall'approccio che viene adottato, per poter considerare il data warehouse come l'unione dei vari data mart che lo compongono, bisogna che quest'ultimi siano progettati seguendo un modello standard, stabilito considerando l'intero business nel suo complesso. Questo standard di progettazione è fondamentale per conseguire l'integrazione dei singoli data mart.

## 4.2.2 - I metadati

Un metadato (dal greco meta- "oltre, dopo" e dal latino datum "informazione" - plurale: data), letteralmente "dato su un (altro) dato", è l'informazione che descrive un insieme di dati. Un esempio tipico di metadati è costituito dalla scheda del catalogo di una biblioteca, la quale contiene informazioni circa il contenuto e la posizione di un libro, cioè dati riguardanti i dati che si riferiscono al libro. Un altro contenuto tipico dei metadati può essere la fonte o l'autore dell'insieme di dati descritto oppure le modalità d'accesso, con le eventuali limitazioni.

In un sistema DSS possiamo considerare due tipi di metadato:

1- **Data integration metadata** e contiene i metadati associati alle informazioni provenienti dalle varie sorgenti OLTP. Questi metadati vengono utilizzati durante il processo ETL. Le informazioni contenute sono:

- nomi e strutture nei sistemi sorgenti;
- tipo dei dati;
- nomi e strutture DWH;
- regole di trasformazione ecc.

2- **DSS metadata**. Questi metadati realizzano il mappaggio delle informazioni contenute nel datawarehouse verso le applicazioni utente, nascondendo all'utente finale i dettagli implementativi del datawarehouse e presentandogli invece una interfaccia di lavoro che opera direttamente con termini propri del business. Le informazioni contenute in questi metadati sono i:

- nomi del business;
- gerarchie e dimensioni;
- le metriche definite ecc.

Questo tipo di metadati viene utilizzato dagli strumenti messi a disposizione dell'utente per accedere alle informazioni del data warehouse.



## **4.3 - Processo ETL**

Talvolta si fa riferimento al Data Integration come all'iceberg del progetto di data warehouse. Anche se l'iceberg non sembra spaventoso dalla nave, spesso non si riesce a cogliere pienamente la sua enormità fino a che non ci si sbatte contro, scoprendo l'enormità della massa che si nasconde sotto le superfici dell'acqua, infatti è ormai dato per certo che i principali limiti delle architetture di BI, riguardino le procedure di aggiornamento batch per l'alimentazione del DW. I sistemi di BI sono in rapida e continua evoluzione, ma ora più che mai risultano vincolati alla fase di Data Integration, in quanto la natura quasi realtime del BPM ha richieste di aggiornamento dei dati sempre più stringenti.

### **4.3.1 - Extract**

Sarebbe fantastico se i sistemi sorgente fossero organizzati in maniera coerente tra di loro, poiché la fase di ETL risulterebbe decisamente semplificata. Purtroppo solitamente la situazione è proprio quella opposta, in cui si hanno una serie di applicazioni stovepipe, che a volte non condividono nemmeno strutture dati base per il business, come clienti o prodotti. Tipicamente le estrazioni riguardano una grossa mole di dati e quindi occorre, mettere a punto un piano di estrazione dei dati che stabilisca quali dati estrarre (strategia incrementale o full refresh) e soprattutto l'orario in cui effettuare tali spostamenti. Ovviamente l'orario preferenziale risulta quello notturno in cui il sistema è affetto da un numero di richieste decisamente inferiore. Estrarre significa leggere e comprendere i dati sorgente, che verranno copiati nella Staging Area.

### **4.3.2 - Transform**

Nella Trasformazione si trattano le questioni di qualità dei dati, si identificano i dati aggiornati, si gestiscono le chiavi surrogate, si creano gli aggregati e si gestiscono gli errori. Questa è senza dubbio la fase che richiede maggiore sforzo. Prima di tuffarsi nell'applicazione delle trasformazioni è fondamentale progettare un piano di staging area, in cui tramite una serie di schemi si descrive le sequenza di operazioni da

applicare, partendo da schemi di livello elevato fino a scendere nello specifico di ogni singola tabella.

E' necessario inoltre valutare la convenienza di acquistare uno strumento di ETL disponibile sul mercato, piuttosto che implementarlo in proprio, tramite script SQL. Ovviamente il ritorno sull'investimento di un tool ETL non sarà immediato al primo utilizzo, ma a causa della curva di apprendimento, fornirà una maggiore integrazione, riutilizzabilità e manutenibilità dei dati a lungo termine.

Vediamo ora più dettagliatamente di quali operazioni si compone questo processo:

- i dati subiscono un processo di pulizia attraverso la correzione di eventuali errori, la risoluzione di conflitti, la trasformazione in un formato standard, ecc;
- i dati subiscono un processo di selezione che permette di scartare le informazioni presenti nei sistemi sorgenti che non sono utili nel data warehouse;
- i dati subiscono un processo di composizione secondo differenti criteri;
- le chiavi dei dati vengono generalmente ridefinite utilizzando dei surrogati in modo da eliminare la dipendenza con le vecchie chiavi dei sistemi sorgenti;
- i dati subiscono un processo di aggregazione per incrementare le prestazioni delle interrogazioni più frequenti.

### **4.3.3 - Load**

Dato che le tabelle dei fatti contengono riferimenti alle dimensioni, vengono prima caricate queste ultime e successivamente quelle dei fatti, in modo da garantire il vincolo di integrità referenziale.

Nella grande maggioranza dei caricamenti, i dati, la cui qualità è adesso assicurata da tutti i controlli effettuati in precedenza, vengono spostati su di un normale database relazionale, che potrà successivamente venire interrogato oppure essere a sua volta usato come input per altre applicazioni; ad esempio può venire utilizzato come sorgente per alimentare un database multidimensionale.

## 4.4 - L'ottimizzazione di un Data Warehouse

In questa sezione affronteremo, molto sommariamente, il problema dell'ottimizzazione di un database analitico progettato secondo le metodologie presentate nelle pagine precedenti. I principi per massimizzare le performance di un data warehouse sono:

- evitare l'aggregazione a runtime dei dati nei casi in cui debbano essere processati un grande numero di record;
- i joins devono coinvolgere il minor numero di tabelle possibile e devono essere molto efficienti;
- le dimensioni fisiche delle tabelle non devono essere troppo elevate. Le tecniche derivanti dall'applicazione di questi principi sono:
  - pre-aggregazione dei dati;
  - indicizzazione e chiavi numeriche;
  - partizionamento.

Vediamo allora in cosa consistono queste tecniche.

### 4.4.1 - Pre-aggregazione dei dati

Permette di migliorare enormemente le prestazioni di query che richiedono dati aggregati. Tipicamente infatti, i dati vengono caricati nel warehouse al minimo livello di dettaglio. Se la query fosse risolta utilizzando solo dati base, bisognerebbe sommare una grossa quantità di records a runtime. Il risultato sarebbe un notevole impegno delle risorse di sistema, deleterio nel caso di ambiente multi-utente. Per evitare questa situazione, i dati possono essere aggregati durante il processo di caricamento. La scelta dei livelli da calcolare preventivamente dipende sostanzialmente dalla loro frequenza di utilizzo. Il risultato di questa strategia sarà la creazione di varie facttable, dette summarytable, contenenti dati a differenti livelli di aggregazione. Il prezzo da pagare per queste prestazioni consiste, però, in un maggior utilizzo di spazio su disco, dovuto alla ridondanza introdotta. Bisogna notare, però, che le dimensioni fisiche delle summarytable diminuiscono al crescere del livello di aggregazione dei loro dati. Le migliori performance si ottengono compilando al 100 per cento il data warehouse. Questo significa creare una summarytable per ogni combinazione dei livelli di

aggregazione. In realtà questa soluzione difficilmente si adotta a causa di due principali motivi: la dimensione del data warehouse aumenta a dismisura e il tempo richiesto per creare le summarytable è troppo elevato. La soluzione ottima nasce allora da un sapiente compromesso fra le performance delle query, lo spazio occupato su disco e il tempo di pre-aggregazione.

#### **4.4.2 - Indicizzazione e chiavi numeriche**

Per velocizzare le operazioni di join, è altamente raccomandato l'utilizzo di chiavi numeriche che sono molto più prestanti delle corrispondenti chiavi testuali. Inoltre bisogna creare molto accuratamente gli indici per le tabelle di lookup e per le tabelle dei fatti contenute nel data warehouse. Attraverso una corretta indicizzazione è possibile aumentare drasticamente le prestazioni in lettura di una query. Questo comporta però, un rallentamento del processo di aggiornamento dei dati, a causa della manutenzione dei vari indici creati.

#### **4.4.3 - Partizionamento**

Questa tecnica permette di contenere le dimensioni di tabelle troppo grandi, suddividendole secondo determinate regole. Ad esempio se consideriamo una tabella denominata "Produttività Mensili" essa può essere suddivisa in 12 tabelle contenenti dati relativi ad un solo mese. I principali vantaggi, che si ottengono adottando questa tecnica, sono:

- il tempo di risposta di una query viene sensibilmente migliorato;
- le operazioni di backup e recovery incrementale sono accelerate;
- il tempo richiesto per caricare i dati in tabelle indicizzate diminuisce;
- problemi che invece nascono a causa del partizionamento sono:
  - operazioni di join e di union aggiuntive sono necessarie per accedere a dati provenienti da differenti sotto-tabelle;
  - maggior difficoltà nella generazione delle query SQL sul set delle tabelle partizionate;

## 4.5 - Dal database OLTP al Data Warehouse

I classici database dei sistemi OLTP differiscono dai database analitici sia nella progettazione, che nelle funzionalità. Le differenze più macroscopiche riguardano:

- l'organizzazione dei dati: abbiamo visto che i dati all'interno di un sistema OLTP sono orientati al tipo di applicazione. Spesso, all'interno di una organizzazione, i sistemi transazionali, che supportano le differenti applicazioni esistenti, sono fra loro eterogenei. Questo significa che i diversi sistemi OLTP presenti all'interno di una organizzazione, contengono differenti tipi di dati;
- all'interno di un data warehouse invece, tutti i tipi di dati vengono integrati in un unico sistema e vengono organizzati secondo le dimensioni (o prospettive) di analisi del business;
- l'integrazione dei dati: I dati contenuti in differenti sistemi OLTP non sono integrati fra loro. Questo significa che le loro chiavi hanno delle strutture differenti, non esiste una convenzione unica dei nomi, possono risiedere in file di differenti formati e in differenti piattaforme hardware;
- il data warehouse rappresenta invece un contenitore di dati perfettamente integrati. Per ottenere questo, bisogna standardizzare le strutture delle chiavi, i nomi e il formato dei file;
- dati storici: Un'altra sostanziale differenza riguarda la quantità di dati storici mantenuti nel database. Abbiamo visto che i sistemi OLTP mantengono in linea i soli dati correnti. Questo vuol dire che la storia mantenuta è di circa 60-90 giorni. In un data warehouse invece, vengono contenuti tipicamente dai 2 ai 5 anni di dati storici, al fine di fornire supporto a particolari tipi di analisi temporali, come la visualizzazione dei trends di determinate misure lungo il tempo, o la comparazione fra periodi di tempo (ad esempio, anno corrente vs. anno scorso);
- l'accesso ai dati: abbiamo visto che i sistemi OLTP sono pensati per processare efficientemente piccole transazioni. Questo implica che il tipo di accesso ai dati è rappresentato da operazioni di interrogazione, aggiornamento, inserimento e cancellazione che coinvolgono una piccola quantità di dati;

- il controllo della consistenza in questi sistemi è effettuato a livello di singola transazione. In un data warehouse il tipo di accesso ai dati è rappresentato da operazioni piuttosto pesanti, tipicamente in sola lettura e coinvolgenti una grande, a volte enorme, quantità di dati. I dati contenuti nel data warehouse non sono costantemente aggiornati, ma vengono caricati ad intervalli periodici di tempo attraverso dei processi batch. Il controllo della consistenza dei dati, viene effettuato solo durante questa operazione di caricamento;
- l'utilizzo delle risorse di sistema: L'ultima grande differenza fra un database OLTP e un data warehouse è il tipo di utilizzazione delle risorse hardware che si riscontra nei due ambienti. Per un sistema OLTP ci sono sia massimi e dei minimi, ma tendenzialmente il carico del sistema si mantiene costante;

Mentre per un sistema contenente il data warehouse, il segnale che descrive le performance assume approssimativamente la forma di un segnale binario, cioè o l'intero sistema viene utilizzato al 100% oppure non viene utilizzato per niente. Questo profilo di utilizzazione nasce dal fatto che tipicamente l'utilizzo dei sistemi DSS consiste nell'esecuzione di grandi interrogazioni, molto impegnative per il sistema, e di lunghe pause per analizzare i risultati ottenuti.

Questa fondamentale differenza, rappresenta una delle varie ragioni per cui si sconsiglia vivamente di mappare i due ambienti operativi su uno stesso sistema hardware. Infatti è possibile ottimizzare una macchina per gestire efficientemente un database transazionale, oppure per gestire efficacemente un data warehouse, ma tentare di ottimizzarla sia per l'uno che per l'altro ambiente si rivelerà sicuramente un insuccesso.

## **4.6 - Gli strumenti di accesso ai dati**

Per poter utilizzare efficacemente le informazioni contenute all'interno di un data warehouse, occorrono particolari strumenti e tecnologie che, nella figura 4.2, sono rappresentate semplicemente dal termine Data Access. Questi strumenti, come si vede nella figura, possono essere raggruppati in diverse categorie, vediamole:

- querytools. Forniscono l'accesso ai dati attraverso una semplice interfaccia grafica. L'utente interagisce direttamente con le strutture delle tabelle memorizzate nel

warehouse, oppure con un livello di astrazione, rappresentato dai metadati , che permette di assegnare dei nomi più significativi per l'utente, agli oggetti utilizzati;

- report tools. Rendono la creazione e la formattazione di un report facile ed intuitiva;
- Multidimensional Database (MDDB). Rendono l'analisi multidimensionale dei dati molto efficiente, memorizzando le informazioni direttamente in formato multidimensionale. Tipicamente, nel contesto di un progetto di data warehouse, i database multidimensionali vengono utilizzati come supporto per la realizzazione di data mart relativi a specifiche aree del business;
- questa classe di strumenti viene identificata anche dall'acronimo MOLAP (Multidimensional OLAP);
- advanceddecisionsupporttools. Permettono l'analisi multidimensionale dei dati direttamente su un database relazionale. Per ottenere questa funzionalità, questi strumenti si appoggiano su uno strato di metadati, che fornisce il supporto a operazioni di analisi avanzate come la creazione di metriche complesse, la navigazione dei dati, etc. Appartengono a questa categoria le architetture ROLAP (Relational OLAP), DOLAP (Desktop OLAP) e HOLAP (Hybrid OLAP);
- Executive Information System (EIS). Attraverso un'interfaccia grafica strutturata, essenziale ed altamente intuitiva, forniscono l'accesso a grafici o reports predefiniti, contenenti informazioni che permettono di controllare i vari processi di un business.

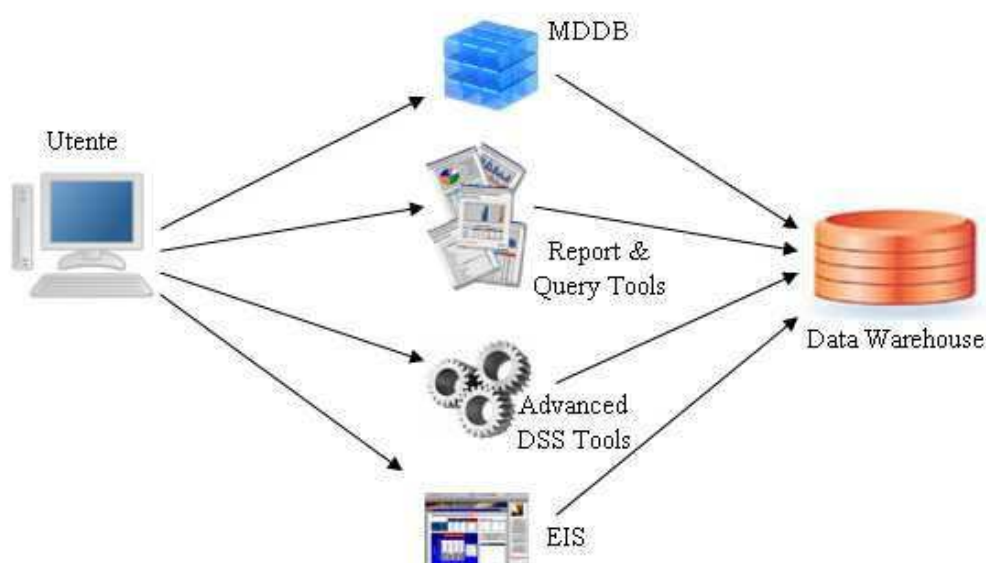


Figura 4.2. Gli strumenti di accesso ai dati di un data warehouse

## Capitolo 5

# QLIKVIEW

---

### 5.1 - QlikView introduzione

#### BUSINESS INTELLIGENCE E QLIKVIEW

Il mondo aziendale si evolve rapidamente e diventa sempre più complesso. La concorrenza si fa sempre più aggressiva e le esigenze dei clienti cambiano di continuo. Le linee di prodotto si sono moltiplicate, i cicli di vita sono più brevi e le attività vengono distribuite geograficamente o gestite in outsourcing. Anche i tempi necessari per le fusioni e le acquisizioni si sono ridotti notevolmente. Infine, le normative sono diventate più severe e la generazione di report sulla conformità è un processo sempre più impegnativo.

Le informazioni, che devono stare al passo con l'evoluzione del business in base ad esigenze specifiche, spesso non sono integrate o sono nascoste (bloccate all'interno di sorgenti di dati diverse, strutture di reporting incoerenti e fogli di calcolo dipartimentali), impedendo una corretta visione delle situazioni operative e finanziarie.

Le applicazioni di Business Intelligence (BI) sono state espressamente concepite per risolvere queste difficoltà e fornire ai responsabili aziendali informazioni che possano essere elaborate senza problemi, cosa che più di dieci anni fa ha portato all'introduzione di una serie di soluzioni sia sviluppate internamente che pre-configurate.

### 5.2 - Caratteristiche di QlikView

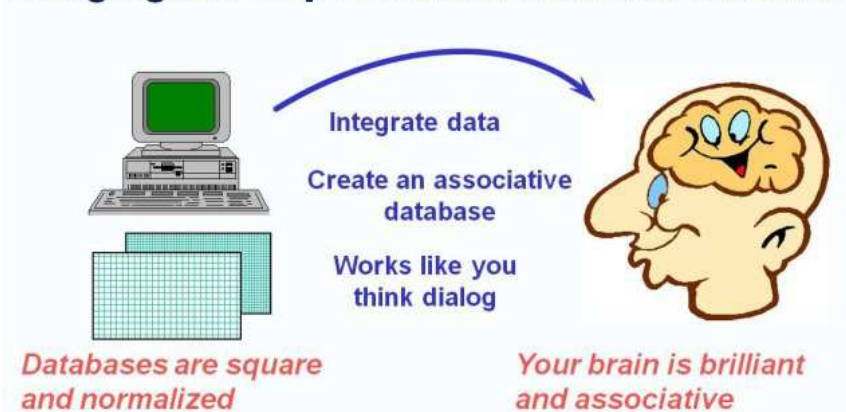
QlikView è un'azienda leader mondiale nelle tecniche di analisi in-memory ed è riconosciuta come la piattaforma di Business Intelligence con il più rapido tasso di diffusione al mondo. Essa propone una nuova classe di soluzioni analitiche facili da



utilizzare, veloci e flessibili che supporta gli individui nel miglioramento delle prestazioni delle organizzazioni nelle quali operano e nel governo dell'innovazione e del cambiamento. QlikView utilizza un'innovativa tecnologia associativa brevettata, che elabora dinamicamente i dati in memoria, per rendere più semplice lo sviluppo, l'utilizzo e la manutenzione di applicazioni di analisi e reporting. Questo prodotto è stato sviluppato partendo da una semplice premessa: l'analisi dovrebbe funzionare come la mente umana ossia per associazione. L'associazione è il concetto che rispecchia più da vicino il percorso della mente umana, un percorso che non segue un criterio lineare, ma si muove per associazioni. Per raggiungere l'obiettivo di analisi associativa, QlikView carica tutti i dati nella memoria e definisce delle metriche a mano a mano che l'utente esegue le selezioni.

Questo processo è chiamato "analisi in-memory".

### **Bridging the Gap Between Man and Machine**



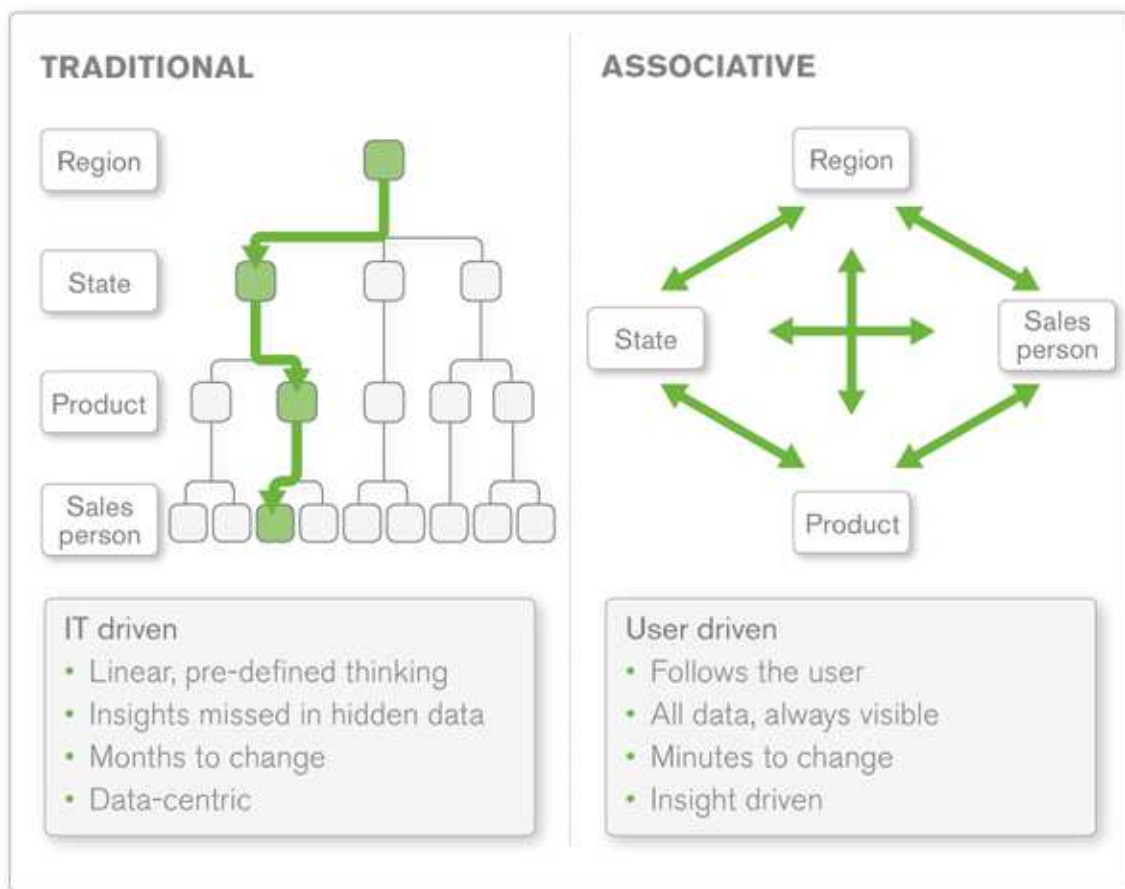
Gran parte dei database standard di QlikView sono basati su un modello relazionale. I record sono suddivisi in modo da ridurre la ridondanza, mentre i campi chiave vengono utilizzati per riunire nuovamente i record nel momento in cui vengono utilizzati. I programmatori di database devono trovare un difficile compromesso tra l'incremento della velocità e l'aumento dello spazio e del tempo necessari per aggiungere o modificare i record. QlikView è una soluzione di architettura semplice, dove tutti i dati sono salvati in memoria e i calcoli vengono eseguiti solo all'occorrenza. L'obiettivo di QlikTech consiste nel fornire potenti soluzioni di analisi e reporting in un quarto del tempo, a costi dimezzati e con un valore doppio rispetto ai prodotti basati sui cubi OLAP. QlikView è progettato in modo che l'intera applicazione (incluso il modello dati) sia memorizzata nella RAM, caratteristica che rende questa soluzione più efficiente

rispetto alle applicazioni OLAP tradizionali. Nel momento in cui avviene il caricamento dei dati da un'origine dati, QlikView crea un modello dati in-memory che consente l'accesso a milioni di celle di dati e risponde alle query in una frazione di secondo. È inoltre possibile eseguire associazioni ad alta velocità con un semplice clic sugli oggetti dei vari fogli e aggiornare immediatamente la visualizzazione. QlikView lavora a velocità superiori e richiede meno spazio rispetto a un database relazionale equivalente in quanto ottimizza le informazioni nel momento in cui le carica, rimuovendo i valori ridondanti dei diversi campi e correlando automaticamente le tabelle. Poiché non sono necessari indici, ogni campo è disponibile come campo di ricerca senza compromettere in alcun modo le prestazioni. Grazie a questa architettura, QlikView richiede in genere un decimo, o anche meno, dello spazio necessario per gli stessi dati rappresentati in un modello relazionale (ad es. in 10 GB di memoria sono contenuti 100GB di dati). Non esiste alcun limite al numero di tabelle consentite in un'applicazione o al numero di campi, righe o celle in una singola tabella. La capacità della RAM è l'unico fattore che limita le dimensioni dell'applicazione.

QlikView consente di analizzare grandi volumi di dati a velocità elevata grazie alla particolare struttura del modello dati in-memory. QlikView esegue calcoli "al volo", fornendo a tutti gli utenti - e non solo ai pochi esperti - funzionalità di analisi multidimensionale. Di conseguenza gli utenti possono ottenere risposte alle query ed eseguire calcoli in una frazione di secondo, nonché elaborare grandi dataset; dal canto loro le aziende possono soddisfare in modo rapido e conveniente le esigenze di un maggior numero di utenti. Sfruttando appieno i vantaggi della capacità di memoria della tecnologia a 64 bit, QlikView è in grado di fornire metriche a livello di riepilogo e dettagli a livello di record sulla stessa architettura. Le aziende possono quindi trarre vantaggio da soluzioni di analisi scalabili che forniscono indicatori KPI a livello di riepilogo, oltre ad analisi dettagliate ed estremamente granulari.

QlikView integra in una singola piattaforma unificata tutte le funzionalità che in genere caratterizzano suite di prodotti più complesse e costose. Oltre a fornire potenti applicazioni analitiche, QlikView consente di eseguire analisi flessibili ad hoc e di creare semplici report stampabili. In questo modo le aziende possono estendere QlikView a tutti gli utenti - dagli analisti qualificati che creano report dettagliati ai dirigenti che hanno bisogno di una panoramica sulle informazioni aziendali critiche,

fino ai responsabili della supervisione degli impianti che analizzano le prestazioni di output. Con QlikView è inoltre possibile sostituire i report cartacei non utilizzati con report generati on demand.



Il modello associativo dei dati gestito in memoria di QlikView consente la visione integrata delle informazioni mediante cruscotti, analisi ad hoc e report, con un unico strumento. Questo modello permette di analizzare i dati sia a livello aggregato sia di massimo dettaglio senza lo spreco di tempo dovuto alla costosa costruzione di cubi OLAP multidimensionali. In aggiunta le associazioni fra i dati sono mappate automaticamente in QlikView e rispondono istantaneamente alle selezioni dell'utente.

Siccome i dati risiedono in memoria, i tempi di risposta di qualsiasi calcolo si mantengono eccezionalmente rapidi anche su set di dati molto estesi e se analizzati contemporaneamente da molti utenti. QlikView fornisce tutte le funzionalità dei tradizionali strumenti di business intelligence in un'unica architettura. Include funzioni di ETL per estrarre, trasformare e caricare dati da una o più sorgenti di dati combinate

(dati ERP, di testo, Excel, XML, ecc.). Lo sviluppo in QlikView è facilitato da funzioni guidate (autocomposizioni) ampiamente esplicative.

L'interfaccia utente è basata sull'uso del mouse e fornisce funzionalità di rappresentazione visiva all'avanguardia, includendo cruscotti, diversi tipi di grafici e tabelle navigabili. Gli utenti possono trovare velocemente le informazioni di cui necessitano e possono condividere i risultati con i loro colleghi utilizzando apposite funzioni di condivisione, report, stampe e tramite la completa integrazione con Office.

Tutte queste funzionalità sono state sviluppate all'interno di un'unica architettura, in questo modo gli strumenti della suite QlikView si adattano perfettamente alle differenti necessità dei diversi ruoli aziendali. Questo approccio inoltre rende QlikView molto veloce da implementare. L'uso di un'architettura integrata inoltre rende QlikView estremamente flessibile ma anche facilmente personalizzabile ed aggiornabile, riducendo il costo di possesso delle applicazioni analitiche proprietarie. QlikView supporta nativamente moltissime dimensioni e qualsiasi valore di qualsiasi dimensione o misura può diventare il punto di partenza di un'analisi. Le modifiche o le integrazioni alle dimensioni e alle misure delle applicazioni possono essere fatte velocemente e questo rende facile rispondere alle mutabili esigenze analitiche aziendali. Gli utenti non sono vincolati da strutture gerarchiche rigidamente predefinite poiché tutti i dati sono portati in memoria e i calcoli sono effettuati al momento della richiesta e non a priori dunque possono trarre notevoli benefici. Dal momento che i dati non devono essere pre-aggregati, gli utenti possono dinamicamente analizzare l'intero volume di dati, fino al livello della singola transazione.

La piattaforma QlikView si integra agevolmente con l'infrastruttura esistente di un'organizzazione rendendo quindi estremamente facile e veloce raggiungere l'operatività. Le analisi effettuate sono rese accessibili attraverso tecnologie web, fornendo interattività, flessibilità e velocità all'interfaccia utente di QlikView.

La piattaforma BI di QlikView offre un'ampia gamma di opzioni di distribuzione, dai client Windows fino a Java e alle applicazioni "zero-footprint" AJAX: con un singolo click del mouse QlikView può essere installato su tutte queste piattaforme. Dipendenti, fornitori e clienti possono connettersi al server QlikView utilizzando un browser Web standard senza dover rinunciare all'uso delle svariate funzionalità interattive che caratterizza l'interfaccia grafica di analisi di QlikView.



Sfruttando le tecniche Web più avanzate, il client AJAX QlikView fornisce un ambiente analitico molto dinamico, interattivo e rapidamente installabile senza occupare memoria. Il client JAVA QlikView offre un'alternativa graficamente valida per l'integrazione di report e analisi all'interno di portali o ambienti intranet basati su Java. QlikView offre inoltre un client plug-in ActiveX per l'integrazione con Office. Il calcolo in memoria introduce ovviamente un collo di bottiglia per le prestazioni delle soluzioni realizzate con questo prodotto, infatti, per ottenere delle buone prestazioni, occorre avere sistemi dotati di molta memoria RAM: tra i requisiti di sistema si può leggere che al minimo sono necessari 2 Gb di RAM ma la capacità realmente necessaria per un buon funzionamento è legata alla mole di dati che ha in carico il sistema. Tuttavia l'introduzione delle piattaforme hardware a 64bit con processori multi-core a basso consumo in combinazione con la precipitosa caduta nel prezzo delle memorie ha permesso di incrementare la scalabilità e le prestazioni di QlikView venendo incontro anche alle esigenze delle organizzazioni più complesse. Oggi QlikView gestisce centinaia di migliaia di utenti e miliardi di record di dati, con il livello di dettaglio sulla singola transazione, con prestazioni quasi istantanee di query lavorando su hardware standard. Con queste performance QlikView costituisce una robusta piattaforma per la business analysis rivolta non solo all'impresa, ma anche alle reti dei clienti, fornitori e distributori fornendo soluzioni in una vasta gamma di aree funzionali quali vendite, marketing, supplychain e produzione, finanza e risorse umane.

### **QlikView – Terminologia:**

Documento QlikView:

- Documento che integra
  - La composizione grafica per implementare le analisi di business
  - La nuvola dei metadati

- Il file fisico al suo interno contiene i dati salvati durante il reload

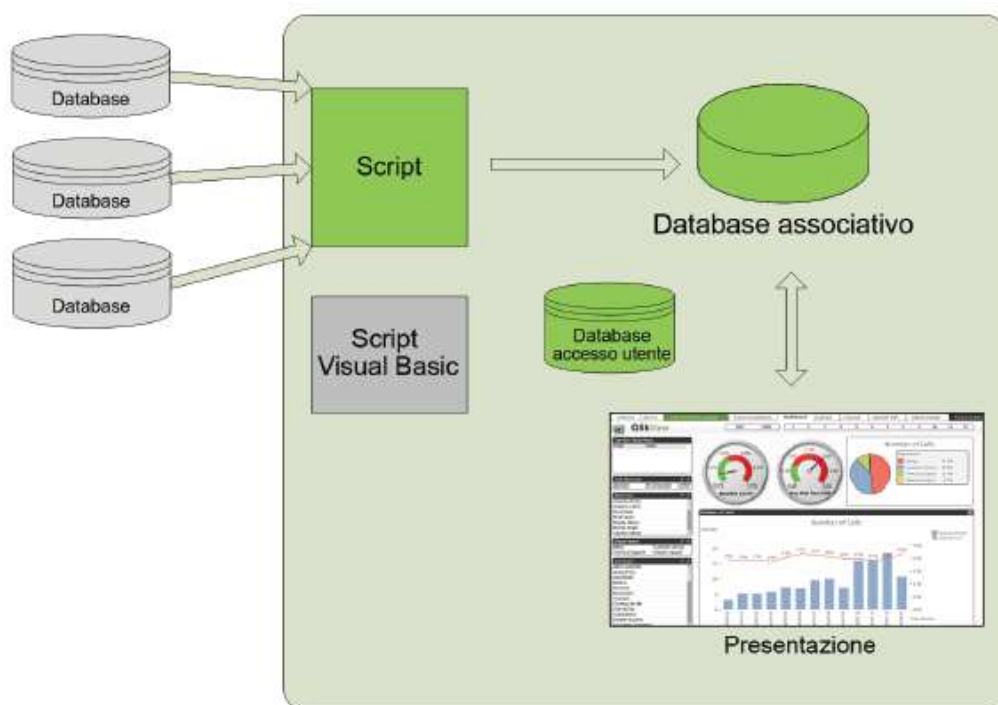
Nuvoladeimetadati:

- Modello che esprime il collegamento (relazione) tra le diverse tabelle
- Modellazione degli oggetti presenti nel dominio di business con una terminologia familiare al keyuser
- Contiene le logiche di trasformazione dei dati

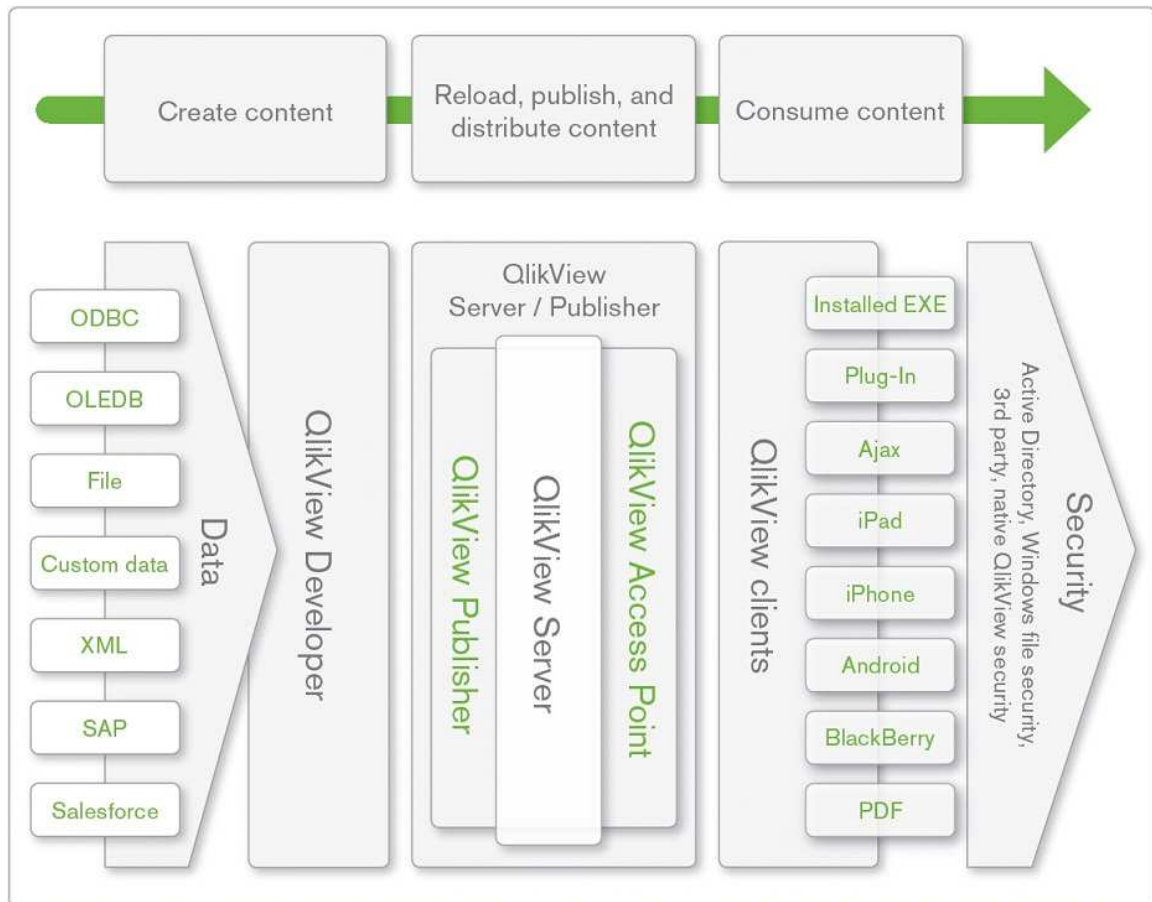
### QlikView – Formati ed estensioni:

All'interno della suite vi sono diversi formati proprietari:

- Qvw(QlikViewDocument)
  - Documento QlikView che contiene l'implementazione grafica dell'analisi
- Qvs (QlikView Script)
  - Documento testuale che contiene il codice AQL di definizione della nuvola dei metadati
- Qvd (QlikView Data)
  - File in un formato compresso e proprietario che contiene i dati
  - Alte performance di caricamento
- Qvx (QlikView Exchange)
  - File in formato xml che consente l'interscambio dati con altre applicazioni o prodotti
- QWT (Qlik Wholesale Trade) Project Plan
  - Documento di progetto di un documento QlikView



## 5.3 - Architettura



### • QlikView Server:

QlikView Server è una piattaforma per l'archiviazione e la condivisione delle informazioni QlikView attraverso Internet o una Intranet. Questa piattaforma è integrata in QlikView per fornire gli utenti finali una suite di tecnologie di analisi dei dati totalmente compatibili. Il componente server di QlikView Server rappresenta il nucleo di questa tecnologia, in quanto fornisce un insieme di documenti QlikView solido, gestito a livello centrale. Inoltre, è in grado di connettere più utenti, tipi di client, documenti e oggetti nell'ambito di un ambiente protetto e sicuro.

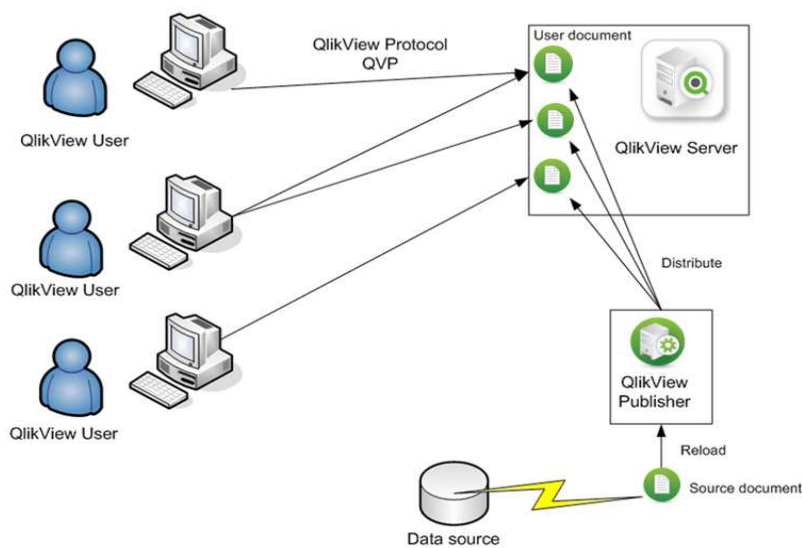
- Gestione utenti
- Repository per documenti
- Reload dei dati (demandato al publisher)
- Autenticazione degli utenti
- Autorizzazione per l'accesso ai dati

- **QlikView Publisher:**

QlikView Publisher è un componente della gamma di prodotti QlikView destinati alla gestione dei contenuti e degli accessi. Le funzioni di presentazione agli utenti finali di informazioni aggiornate e di gestione avanzata dei documenti di QlikView rendono QlikView Publisher un valido componente aggiuntivo di questa suite di prodotti.

QlikView Publisher consente di distribuire i dati salvati nei documenti QlikView agli utenti che si trovano all'interno e all'esterno dell'organizzazione. Grazie alla riduzione dei dati, ogni utente potrà visualizzare soltanto le informazioni di cui ha bisogno. Il servizio QlikView Publisher e l'interfaccia utente sono ora completamente integrati in QlikView Server e QlikView Management Console (QMC). Se non si dispone dell'accesso a QlikView Server o a QlikView Publisher, è possibile ricaricare i documenti QlikView su un'applicazione autonoma.

- Reload dati
- Riduzione dati
- Distribuzione reportistica + PDF
  - Per PDF serve licenza a parte





- **QlikView Access point:**

Portale di pubblicazione e consultazione dei report

- Portale di accesso a reportistica
- Download file

- **QlikView Desktop:**

- Creazione e modifica dei report

- **Mobile:**

- HTML 5 di adatta al device che viene utilizzato
- Visualizzazione mobile
  - Ogni foglio visto come una riga che contiene tutti i vari oggetti all'interno del tab

- **Tipi di architettura:**

- QlikViewstandalone
  - Per bloccare il reload del documento d'aparte dell'utente si può usare la sectionaccess con l'Access DB (che non è Access di microsoft)
  - Permette di disabilitare il pulsante di reload del documento per gli utenti che non devono farlo
- QlikView server

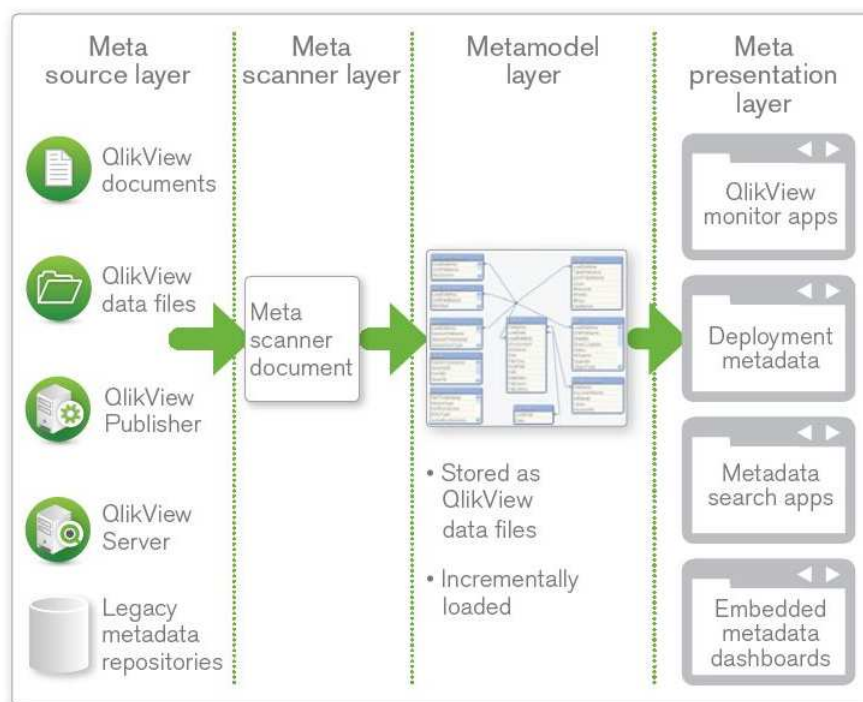
- **Modalità di distribuzione:**

- standalone
  - condivisione su share di rete
  - per un ambiente in cui c'è solo uno sviluppatore
  - come sviluppatore devo fare in modo che l'utente non possa modificare il report e non vada a visualizzare le proprietà dei grafici.
    - Si fa con le sectionaccess e con il tab sicurezza delle proprietà del documento (questo tab non è visibile dallo user consumatore)
- Pubblicazione su server
- Publisher
  - Divide il documento sorgente da quello che verrà pubblicato per l'utente

- **Modellazione dati:**

- Se una tabella ha 2 righe duplicate (identiche) quando questa tabella viene incrociata con altre nella nuvola, una viene persa
- ODBC
  - è più lento. Viene gestito come un driver hardware
  - è possibile forzare l'utilizzo del 32 bit quando il sistema è a 64bit. Carica un driver a 32 bit anche se l'ambiente è a 64
  - per Oracle, usare database driver della giusta versione, altrimenti legge male

## Architectural view of QlikView metadata model



© 2011 QlikTech

## 5.4 - Security in QlikView

QlikView è nato per proteggere i dati critici e il patrimonio di analisi aziendale. QlikView si integra con le infrastrutture di sicurezza esistenti, specificamente progettate per proteggere miliardi di record o migliaia di utenti. La console di sicurezza QlikView, facile da gestire, consente all'IT di:

- Controllare quali utenti vedono cosa: analisi, dati, metriche e risultati
- Definire la sicurezza dal livello applicativo fino al livello granulare di riga
- Supportare sistemi di servizi di directory standard quali Microsoft Active Directory e altre soluzioni LDAP
- Applicare regole basate su gruppi, ruoli e singoli utenti

I moduli Publisher e Access Point di QlikView forniscono funzioni di implementazione centralizzata che vanno dall'amministrazione alla pianificazione degli aggiornamenti dei dati e consentono di determinare quali tecnologie client degli utenti abilitare. Ciò consente all'IT di:

- Controllare quando devono essere aggiornati i dati di QlikView
- Determinare a quali utenti rendere accessibili i dati
- Stabilire le modalità di accesso ai dati da parte degli utenti

Tutte le opzioni di implementazione dei client: client Ajax scaricabile gratuitamente e indipendente dal browser, client installato, plug-in del browser, client mobile o PDF, sono gestiti da un'unica postazione centrale.

### **Publisher Security:**

- Consente di suddividere un file QlikView di grandi dimensioni in file più piccoli in base alla protezione a livello di riga
- Efficace per la gestione della memoria se i profili di sicurezza NON contengono molti dati sovrapposti

### **Section Access:**

- Riduce dinamicamente il file QlikView all'accesso in base all'autorizzazione dell'utente

All'interno di un documento QlikView è possibile definire:

- gli utenti che possono accedere ai contenuti
- un accesso partizionato ai dati in base alle policy definite
- Livelli di accesso
  - ADMIN → può modificare qualsiasi elemento di un documento
  - USER → dispone di un accesso limitato
  - NONE → nessun accesso

Se non viene assegnato nessun livello di accesso, l'utente non potrà accedere al documento.

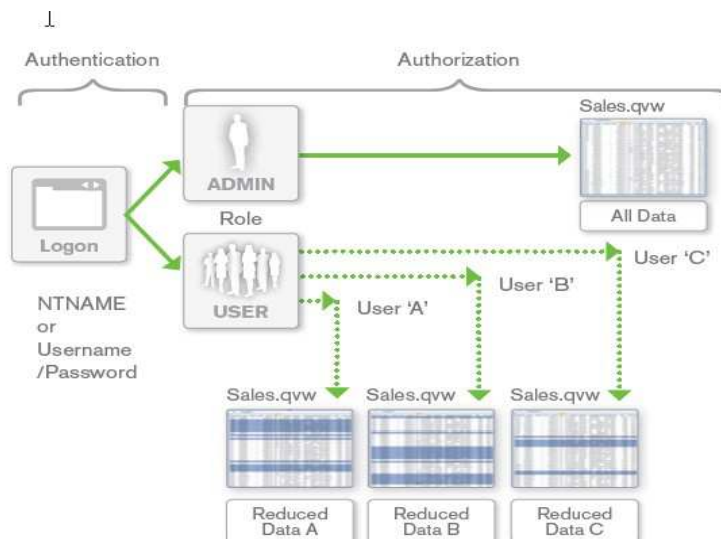
### Controllo degli Accessi:

Gestito mediante

- File di testo
- Database
- Dati inseriti INLINE

### Definizione di section access:

- Creazione di un hidden script
  - Richiede una password di admin
  - Se la si perde non ce modo di recuperarla, quindi ATTENZIONE!!!!
- Dichiarazione della SECTION ACCESS per la lettura degli utenti
- Dichiarazione della SECTION APPLICATION per la lettura delle altre tabelle che compongono la nuvola
- Al suo interno tutti i dati vengono trasformati in maiuscolo (in automatico)



## 5.5 - La piattaforma QlikView Business Discovery

La piattaforma QlikView Business Discovery offre la BI self-service che mette a disposizione degli utenti strumenti potenti che consentono un processo decisionale innovativo.

### **QlikView offre una comprensione tutta nuova del business attraverso:**

- Consolidamento dei dati importanti da più origini in un'unica applicazione
- Esplorazione delle associazioni dei dati
- Processo decisionale "sociale" attraverso la collaborazione sicura in tempo reale
- Visualizzazione dei dati con grafici interessanti allo stato dell'arte
- Ricerche su tutti i dati, direttamente e indirettamente
- Interazione con applicazioni dinamiche, dashboard e analisi
- Accesso, analisi e acquisizione dei dati da dispositivi mobili

### **Un percorso naturale dai dati alle decisioni**

Nessuno prende decisioni basate solo su dati numerici, ma tiene in considerazione anche input del mondo circostante, incluse le conversazioni con gli altri. Ecco perché QlikView mette a disposizione più fonti di conoscenza attraverso:

- **Persone**

"La Social Business Discovery consente di generare in modo collaborativo analisi condivise in tempo reale per prendere decisioni innovative più rapidamente".

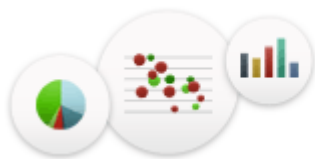


Processo decisionale collaborativo con colleghi tramite Social Business Discovery

- **Dati**

"QlikView fornisce risposte alla stessa velocità con cui vengono concepite le domande. Esplora i dati, fa scoperte e trova le informazioni che consentono agli utenti di risolvere i problemi aziendali secondo modalità nuove e innovative. Esegue ricerche dirette e

indirette e interagisce con dashboard e analisi dinamici su qualsiasi dispositivo. QlikView offre analisi inaspettate perché funziona come la mente umana".



Analisi basate su ERP, CRM, data warehouse, database SQL, dati di Excel e molto altro ancora.

- **Luoghi**

"Mobile Business Discovery consente agli utenti di sfruttare la libertà di essere in un determinato posto in un momento preciso. Inaspettatamente arriva un ordine per 10.000 pezzi di un articolo fuori produzione? Un produttore chiama perché deve piazzare tutto l'inventuto di un componente difettoso? In un attimo è possibile collegarsi all'applicazione di analisi di vendite, magazzino o supplychain dal dispositivo mobile e annotare le proprie osservazioni. Altri che hanno accesso all'applicazione vedono le annotazioni e inizia un thread di conversazione mentre i membri del team prendono decisioni che portano a una soluzione."



Informazioni concrete in tempo reale dal campo tramite la Business Discovery mobile.

## 5.6 - QlikView migliora il BI

### **Più strumenti a disposizione degli utenti**

L'esperienza associativa di QlikView fornisce risposte alla stessa velocità con cui vengono concepite le domande. Permette di interagire con i dati senza limitazioni per generare analisi secondo modalità impensabili.

### **QlikView consente di:**

- Individuare trend nascosti e fare scoperte che favoriscono decisioni innovative;

- Porre domande e cercare informazioni autonomamente o in collaborazione con altri, in tempo reale, per strada;
- Eseguire ricerche su tutti i dati per una visibilità completa. Basta digitare in qualsiasi ordine una parola o una frase nella casella di ricerca di QlikView per ottenere risultati associativi immediati che consentono di vedere nuove connessioni e relazioni tra i dati;
- Sfruttare i vantaggi della BI self-service senza la necessità di rivolgersi agli analisti IT o aziendali per nuove query o report.

## 5.7 - Business Discovery

- **Analisi ovunque**

La Business Discovery è un modo tutto nuovo di fare le cose che mette l'utente al posto di comando. A differenza della BI tradizionale, che coinvolge solo poche persone, la Business Discovery consente a tutti di creare analisi approfondite. Gruppi di lavoro, reparti e intere business unit hanno accesso ai dati di cui necessitano per adottare decisioni migliori. Con QlikView le aziende possono analizzare ogni aspetto dell'organizzazione, consentendo a ogni singolo utente di svolgere il proprio lavoro in modo più intelligente e veloce che mai. Con QlikView tutti gli utenti possono creare analisi approfondite personalizzate in grado di soddisfare esigenze aziendali e tempistiche esclusive.

- **Ambiente sociale e collaborativo**

La Business Discovery consente agli utenti di condividere dati importanti con altri utenti singoli, gruppi di lavoro e reparti. Consente il discovery collettivo nell'ambito di spazi di lavoro in-context, l'acquisizione di conversazioni, la condivisione sicura delle informazioni e la valutazione dei risultati delle decisioni. È basato su una piattaforma aperta che ne consente l'integrazione con i social network e le reti aziendali esistenti.

- **Modello di applicazioni**

La Business Discovery consente agli sviluppatori di implementare in pochi giorni o settimane applicazioni analitiche in grado di risolvere problemi specifici utilizzando i

dati forniti dall'IT. Le applicazioni QlikView riguardano attività specifiche e sono facilmente accessibili e utilizzabili senza formazione.

- **Trasformazione e ricomposizione**

Nessuno può prevedere quali domande avranno gli utenti nel momento in cui iniziano ad analizzare i dati, neanche gli utenti stessi. La Business Discovery semplifica la trasformazione e la ricomposizione dei dati in nuove viste e la creazione di nuove visualizzazioni immediate per una maggiore comprensione.

- **Mobilità**

La Business Discovery supporta un nuovo mondo di applicazioni e dati mobili. I dati e le analisi live consentono agli utenti di rispondere a qualsiasi domanda mentre sono fuori ufficio, con qualsiasi strumento: tablet Apple iPad, Android o smartphone come Apple iPhone e Android.

## **5.8 - Più professionisti IT sono in grado di fornire la BI self-service**

QlikView consente ai professionisti IT di offrire un'esperienza di BI self-service completa e servire l'azienda come mai in passato, il tutto assicurando l'assoluta sicurezza, qualità e governance dei dati sul back end.

### **È possibile:**

- Consolidare i dati da più sorgenti per offrire agli utenti una visione completa
- Rimanere concentrati sulla sicurezza dei dati, sul provisioning di dati e applicazioni, sulla governance dei dati e sulla manutenzione dei sistemi
- Eliminare la necessità di scrivere e riscrivere i report, modificare query e creare cubi



La console di gestione di QlikView basata su web offre:

- Un modo semplice per garantire a tutti l'accesso ai dati e alle analisi più recenti;
- Enterprise Manageability con amministrazione basata sui ruoli per garantire che l'accesso ai dati sia consentito solo agli utenti provvisti di autorizzazione.

### **Offre potenti strumenti all'IT**

Grandi organizzazioni di tutto il mondo, appartenenti ai più diversi settori, si affidano alla solida piattaforma di classe enterprise QlikView Business Discovery per prendere decisioni migliori. QlikView consente ai professionisti dell'IT di fornire la BI self-service, offrendo all'azienda un vantaggio competitivo. Grazie a QlikView, l'IT è in grado di offrire agli utenti la possibilità di creare analisi personalizzate per poter giungere a decisioni innovative. QlikView consente all'IT di servire il business come mai in passato. Il tutto garantendo l'assoluta sicurezza, la qualità e la governance dei dati.

L'architettura associativa, in-memory di QlikView offre la scalabilità per gestire miliardi di set di dati o migliaia di utenti. In combinazione con le funzionalità di accesso diretto ai dati della Direct Discovery di QlikView, è possibile gestire una quantità di dati virtualmente illimitata. Consente all'IT di fornire una user experience coerente e altamente proattiva, e di avere più tempo per concentrarsi su competenze chiave.

### **Semplificare l'attività di professionisti IT e sviluppatori**

QlikView semplifica il lavoro di analisti aziendali e sviluppatori impegnati nella creazione di applicazioni di Business Intelligence all'avanguardia che gli utenti possono migliorare e ampliare man mano che cambiano le esigenze aziendali.

### **Con QlikView, professionisti e sviluppatori QlikView possono:**

- Estrazione dei dati da sistemi di origine, creazione di modelli di dati, trasformazione dei dati, descrizione di metadati, creazione di livelli di archiviazione dei dati e definizione dell'interfaccia utente;
- Creazione e manipolazione di un'ampia gamma di visualizzazioni e relativa modifica in tempo reale;
- Creazione di estensioni utilizzando QlikView Workbench (un plug-in di Microsoft Visual Studio);
- Integrazione con i portali aziendali e i siti di collaborazione utilizzando le parti web di QlikView per Microsoft SharePoint;
- Collegamento diretto con le applicazioni aziendali e le origini dati più diffuse, tra cui Salesforce.com, SAP NetWeaver e Informatica.

## **Creazione e personalizzazione semplice di AppQlikView**

La piattaforma QlikView Business Discovery offre un approccio completamente nuovo alla BI, consentendo agli sviluppatori e agli analisti aziendali di creare in modo rapido e semplice applicazioni per gli utenti, i quali possono partecipare al processo di creazione. Le soluzioni di BI tradizionali sovraccaricano le organizzazioni IT con cicli di sviluppo lunghi e costosi che rimandano un valore dimostrabile di settimane, mesi o addirittura anni. QlikView, invece, offre la possibilità di creare e modificare le applicazioni in pochi minuti, eliminando l'attesa in linea per il supporto dell'help desk o per la risoluzione dei ticket.

## **5.9 - Limiti diQlikView**

Logica associativa:

- Le tabelle vanno in associazione automaticamente se hanno campi con nomi uguali. Se non si volesse l'associazione bisogna rinominare i campi.

In-memory:

- Per navigare sui dati si caricano prima tutti i dati in memoria. Questa operazione richiede tempo e memoria fisica disponibile.

Back-end:

- Scripting AQL, è necessario avere conoscenze del linguaggio proprietario di QlikView molto simile al SQL ma con specifiche e funzionalità caratteristiche del prodotto.

## 5.10 - QlikView Best practices

Le regole di best practices:

### Struttura delle cartelle del progetto

- DataSource
- Documentazione
- DW Staging
- DW Associativo
- Impostazioni
- Qvw ETL
- Qvw Presentation

#### DataSource

In questa cartella andiamo ad inserire tutti i file che ci permettono di gestire le info esterne: vedi file excel, sql, csv, o altre fonti.

#### Documentazione

In questa cartella inseriamo tutta la documentazione utile ai fini del progetto.

#### DW Staging

Questa cartella conterrà tutti i file QVD (QlikView Data) del primo livello.

In altre parole nella fase di ETL01 importeremo semplicemente tutte le tabelle che ci serviranno all'interno di questa cartella, trasformate in formato qvd.

Questo permetterà una lettura delle tabelle per la fase di Staging, in tempi notevolmente ridotti, rispetto alle letture sulle sorgenti.

#### DW Associativo

Questa cartella conterrà tutti i file QVD (QlikView Data) del secondo livello.

In altre parole nella fase di ETL02 andremo a costruire *la Nuvola*.

La logica della costruzione della nuvola (che corrisponde al Datamart) dipende dal progetto.

## **Impostazioni**

Questa cartella conterrà dei file di configurazione, che serviranno a controllare le fasi di Extraction/Load dell'ETL01 e dell'ETL2 contenute nel file Impostazioni.xls, inoltre servirà per la fase di rinomina delle Dimensioni/FactTable e dei loro Metadati, in maniera controllata e centralizzata, per l'ultima fase di costruzione della nuvola:

- Parametri di connessione,
- Tabelle da estrarre +query utilizzate,
- KPI, KPI language, KPI ordering
- *Renaming\_Tables, Renaming\_Metadata*
- Variabili utilizzate

## **Qvw\_ETL**

Questa cartella conterrà tutti i file di ETL, con le varie logiche di primo livello e secondo livello.

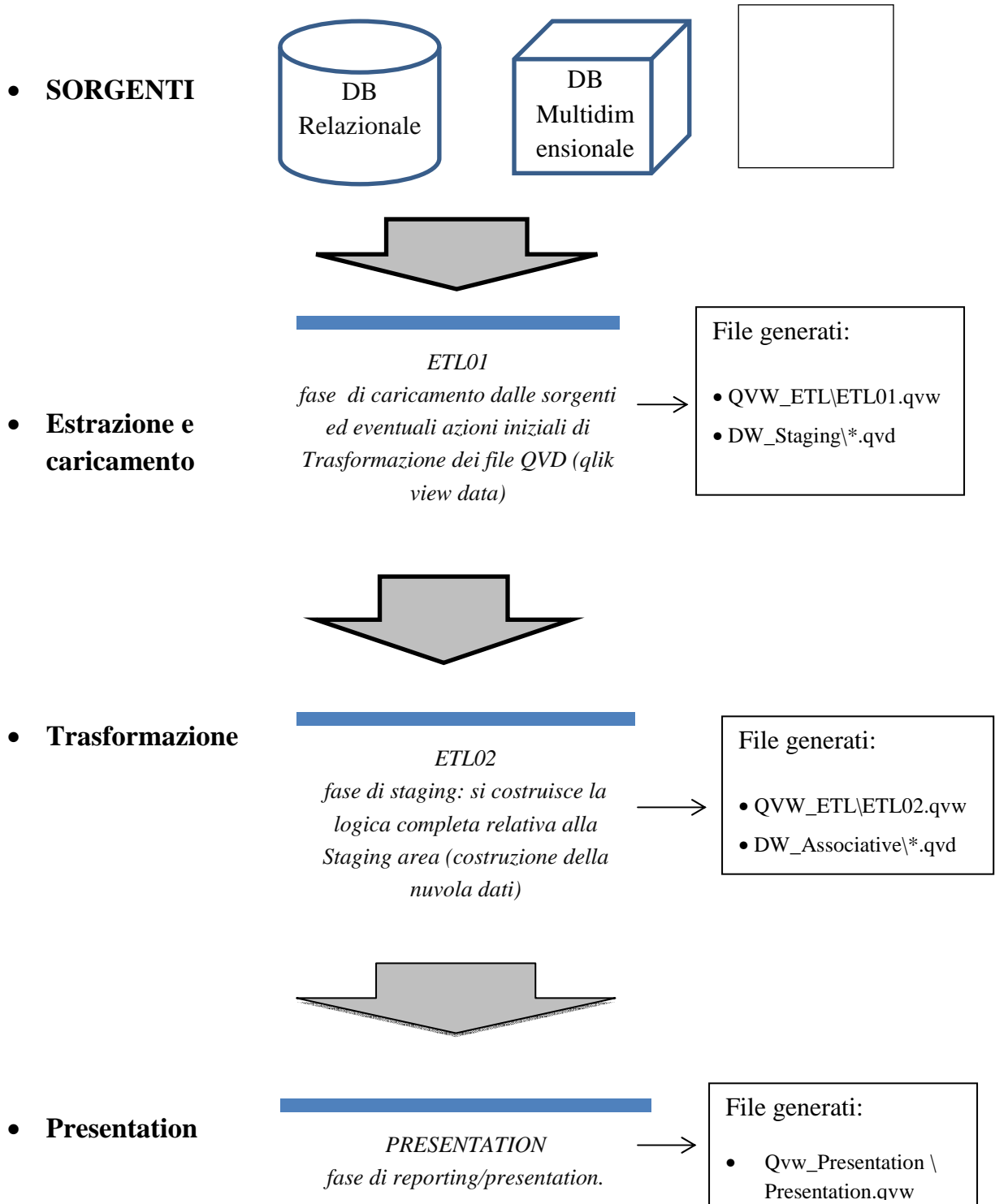
I file potrebbero essere anche suddivisi se i progetti si rilevano lunghi e complessi.

Inoltre si consiglia la suddivisione degli stessi, nel caso di progetti complessi, per permettere una maggior rapidità nei task di caricamento e costruzione della logica, in quanto si può parallelizzare tutto il processo.

## **Qvw\_Presentation**

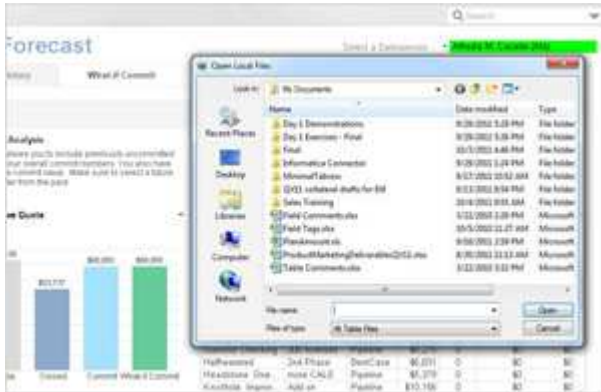
Presentation.qvw sarà, invece, file con la logica di business creato in ETL2, dove i progettisti/designer si potranno dedicare alla sola fase di costruzione delle Dashboard e dei Report per le analisi.

## Flusso base di un progetto



## 5.11 - Sviluppo veloce e rapido time-to-value

In una frazione del tempo richiesto dalle soluzioni BI tradizionali, QlikView consente di creare applicazioni che offrono funzionalità di analisi, reporting e dashboard fondamentali per il processo decisionale e le operazioni. **Funzionacosì:**

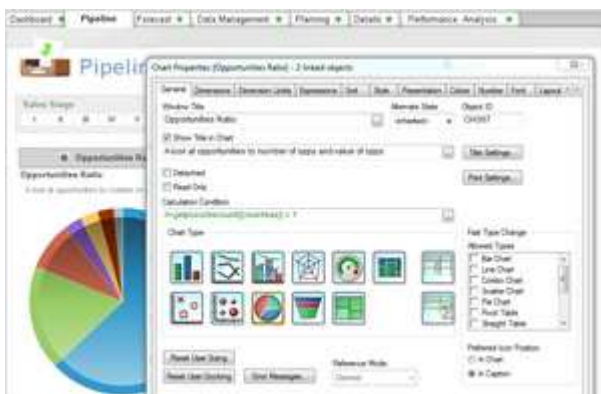


### 5.11.1 - Caricamento dei dati

QlikView rende semplice e veloce il processo di acquisizione dei dati grazie a:

- Consolidamento automatico dei dati in un'unica visualizzazione
- Associazioni di dati gestite automaticamente
- Procedure guidate per i collegamenti e connettori pre-configurati per SAP, Informatica e Salesforce.com

Questo processo richiede una frazione del tempo necessario a una soluzione BI tradizionale e tutto viene eseguito all'interno di un unico ambiente QlikView. Non è necessario acquistare e integrare più prodotti, ciascuno con un'interfaccia utente diversa.



## 5.11.2 - Creazione di applicazioni con la tecnica point and click

Con QlikView basta puntare e cliccare in tempo reale sugli elementi integrati dell'interfaccia utente. La tastiera non serve. Basta utilizzare il mouse per aggiungere:

- Caselle di elenco con dati caricati automaticamente per eseguire analisi associative immediate
- Grafici, diagrammi e tabelle
- Menu a discesa
- Caselle di ricerca
- Pulsanti tipo browser web (ad es. Avanti, Indietro e Annulla)
- Immagini integrate
- Mappe

QlikView consente di velocizzare lo sviluppo, presentare un primo approccio ad altri aspetti del business e apportare le necessarie modifiche in pochi minuti ed ore—non in giorni o settimane.



## 5.11.3 - Estensione e miglioramento

Dopo aver creato un'applicazione QlikView, è possibile ampliarla e migliorarla in modo semplice e veloce per rispondere a esigenze aziendali sempre diverse. È possibile integrare nuove origini dati, visualizzazioni e immagini di terze parti o personalizzate e QlikView esegue un aggiornamento dinamico per rispecchiare le modifiche.



### 5.11.4 - Visualizzazione

L'applicazione QlikView consente agli utenti di analizzare i dati scegliendo tra un'ampia varietà di visualizzazioni interattive. **QlikView offre:**

- Una serie completa e associata di visualizzazioni dei dati a disposizione di tutti gli utenti
- Diagrammi e grafici interessanti e allo stato dell'arte che riuniscono più punti di vista in una singola visualizzazione
- La possibilità di fare clic in un punto qualsiasi all'interno di una visualizzazione per accedere a dati più granulari



## 5.12 - Stima dimensionamento

### Requisit:

La piattaforma applicativa **QlikView** implementa la tecnologia associativa *in memory*. Per ottenere i maggiori benefici in termini di *performances* che questa tecnologia consente, è necessario che l'infrastruttura tecnologica che la ospita disponga delle corrette risorse hardware: memoria RAM e processori / *core* in particolare.

Una ipotetica applicazione QlikView, che si sostanzia in un file .qvw, utilizza la RAM per:

- contenere la nuvola associativa (quindi le tabelle dei dati, le strutture binarie interne destinate a mappare le associazioni e ad ottimizzare l'occupazione eliminando la ridondanza in caso di dati ripetuti);
- contenere gli oggetti di *presentation* (es. tabelle, tabelle pivot, grafici ecc.);
- calcolare le espressioni contenute negli oggetti (maggiore è la complessità delle espressioni da calcolare *real time*, maggiore sarà la richiesta di memoria e risorse di calcolo ...);
- elaborare gli script per il caricamento della nuvola associativa a partire dalle fonti alimentanti (c.d. processi ETL). In questo caso l'ammontare della memoria richiesta sarà funzione delle necessità o meno di effettuare elaborazioni in memoria quali join, mapping ecc. ;

La medesima applicazione, utilizza invece le risorse di calcolo (processori / *core*):

- durante l'elaborazione degli script, in particolare durante la fase di calcolo delle associazioni che avviene una volta completato il caricamento in memoria della nuvola associativa;
- per calcolare le espressioni contenute negli oggetti di presentazione (es. funzioni, aggregazioni, *set analysis* ecc.)

QlikView è in grado trarre beneficio dalla disponibilità di sistemi multiprocessore / *multicore* sia nella fase di processo ETL che nella fase di *renderizzazione* dell'informazione.

Nel calcolo della configurazione del sistema, occorre inoltre considerare il numero di utenti concorrenti e la complessità delle logiche di calcolo implementate. Per ciascun utente occorre infatti prevedere un ammontare addizionale di memoria RAM valutabile mediamente nel *range* del **10-13%** dell'ammontare originario di memoria usata dall'applicazione. La concorrenza di più utenti e la complessità delle logiche di business implementate impatta poi sulla necessità di processori / *core*.

Ciò premesso, per agevolare il calcolo dei requisiti minimi hardware, **QlikTeck** ha sviluppato un foglio di calcolo che partendo dall'occupazione di RAM prevista per ciascuna applicazione, dal numero di utenti concorrenti che insistono sulla medesima applicazione e da una stima del peso delle *query* / elaborazioni di *presentation*, consente

di stimare l'ammontare di memoria necessaria ed il numero di processori / core.

**Occorre precisare che il processo di calcolo è comunque empirico ed è basato sull'esperienza accumulata, sull'accuratezza dei dati disponibili e, più in generale, sulla disponibilità o meno di un modello dati associativo già disegnato.**

Il punto di partenza sono dunque i requisiti da soddisfare.

Le informazioni disponibili su cui basare la stima di occupazione sono di seguito riepilogate:

- Necessità di sviluppare un applicazione / cruscotto QlikView basato sull'analisi di 4 facttable che presentano il medesimo tipo di tracciato
- Cardinalità delle tabella dei fatti:
  - F\_MOMA\_SETTIMANALE: circa 120M di righe
  - F\_MOMA\_MENSILE: circa 30M di righe
  - F\_MOMA\_CLIENTI\_SETTIMANALE: circa 8M di righe
  - F\_MOMA\_CLIENTI\_MENSILE: circa 2M di righe
- Lunghezza record massima della tabella dei fatti: circa 90 Byte.
- Non si è a conoscenza di logiche di calcolo particolari che non siano delle semplici aggregazioni, né di processi di aggiornamento *nearreal time*.

Non essendo ancora disponibile il disegno del modello associativo previsto ed una previsione della cardinalità di ciascun campo (ricordiamo che QlikView adotta un algoritmo di compressione della memoria utilizzata che funziona mediante l'eliminazione delle ridondanze), si ipotizza la stima di occupazione in RAM di una nuvola avente le seguenti caratteristiche:

- 1 tabella dei fatti risultante dalla concatenazione delle 4 facttable sopra citate con quindi circa 160 milioni di record, 10 campi di analisi e 4 campi misure con un'occupazione fisica di 15 Gb;
- Bassa complessità di calcolo della presentation.

Sulla base dell'esperienza accumulata in altri progetti di dimensione e complessità simile, la stima è di un'occupazione in RAM di circa **10 Gb**.

Si è quindi proceduto ad effettuare le seguenti assunzioni:

- 50 utentonominali
- Un indice di concorrenza del 15% , ovvero 8 utenti mediamente concorrenti

Applicando i parametri di cui sopra al modello di calcolo otteniamo il seguente **fabbisogno minimo** da dedicare all'applicazione:

**26 Gb di memoria Ram + Due core di elaborazione.**

## Capitolo 6

# IL CASO DI STUDIO: Commercial Tool

---

Nella fase seguente procederemo con la descrizione del caso di studio che è stato preso in considerazione per lo sviluppo delle tecniche e delle logiche progettuali precedentemente descritte.

È molto importante tenere presente che i dati rappresentati nel corso della tesi (negli screenshot di esempio) rappresentano dei valori assolutamente diversi e non veritieri o saranno offuscati, questo perché l'azienda cliente non autorizza alla divulgazione in quanto dati sensibili per la stessa.

Il caso di studio di seguito trattato rappresenta un progetto pilota. È il primo esempio di collaborazione fra l'azienda di consulenza ICONSULTING e il cliente. Nello specifico gli attori coinvolti sono:

- **Azienda Cliente:** una società multinazionale di macchine agricole e macchine di movimento terra (utilizzatore finale e committente del progetto).
- **Iconsulting** (sviluppatore del progetto)

Il fine ultimo del progetto è stato quello di realizzare una piattaforma di Business Intelligence in grado di fornire un supporto alle operazioni di forza vendita e di analisi. La richiesta, fatta dal cliente, ha permesso di sviluppare un efficace sistema di supporto alle operazioni di vendita e di analisi, che andrà migliorando nel tempo.

### 6.1 - Descrizione Funzionale

Obiettivo di questa fase è quello di descrivere il modello di implementazione delle relazioni richieste nell'ambito del progetto.

La soluzione proposta ha l'obiettivo di fornire ai gestori di vendita (Sales Manager) uno strumento di vendita di Business Intelligence. Lo strumento è finalizzato a consentire ai responsabili di zona (Zone Manager) di analizzare le principali informazioni del portafoglio ordini (OrdersPortfolio), in particolare l'analisi del workflow (Ordini di

vendita → Ordine di produzione → Fatturazione) dei propri clienti, sia tramite dispositivi mobile che desktop, consentendo una interazione più efficace e più efficiente con i venditori (Dealer) e i clienti (Customers).

Descriveremo nel dettaglio le esigenze degli utenti emerse dopo diverse riunioni e scambio di email con il reparto IT e i zone manager.

## **6.2 - Scopo del progetto**

L'intento del progetto è stato quello di costruire un nuovo sistema di reportistica basato su tecnologia QlikView.

Nel processo attuale, i zone manager effettuano analisi basate su report costruiti ad hoc per ciascun mercato dal reparto IT.

I strumenti di reportistica usati dal cliente sono MS Excel, MS ACCESS e COGNOS BI.

Il nuovo strumento ha lo scopo di integrare tutti questi report ad hoc in unico punto di accesso per l'analisi e reporting del portafoglio vendite, che consente all'organizzazione di fornire informazioni e analisi omogenee, integrate e certificate.

Basandosi sui requisiti da noi raccolti e le esigenze degli utenti che abbiamo identificato, abbiamo progettato i componenti di una soluzione software utilizzando lo strumento QlikView.

Questo strumento permetterà all'azienda interessata di ottenere i seguenti importanti vantaggi:

- Migliorare l'analisi e la reportistica.
- Gestire tutte le informazioni del Portfolio Ordini in un strumento unico per l'intera Società.
- Aumento dell'efficienza dei gestori di vendita, fornendo uno strumento per controllare facilmente lo stato dell'ordine.

Requisiti principali:

- Schede di dati del cliente gestite tramite Portfoglio Ordini
- L'informazione fornita dal Business Intelligence Tools sarà consistente e certificata con il loro Enterprise DWH.
- Analisi Offline
- Gli utenti avranno un accesso read-only
- Supporto a mobile
- Etichette in diverse lingue
- Data history di 5 anni per l'analisi on line
- Modulo di ricerca
- Manuale utente
- Strumento unico per tutti i mercati

## 6.3 - Analisi dei requisiti

Basandosi sui requisiti da noi raccolti e le esigenze degli utenti che abbiamo identificato, si è arrivati alla condivisione dei mockup che rappresentano un esempio a grandi linee dei report finali.

**Home page:**

- L'ultima foto degli KPI principali
- Prompt filtri di dimensioni
- Prompt di ricerca
- Tabella doppia entrata KPI e Dimensioni
- 2 Grafici

**Summary Report Monthly:**

- Prompt di ricerca
- Prompt filtri di dimensioni
- Tabella Order Summary: KPI month to date
- Tabella Status Report: KPI del ciclo di vita di un veicolo dalla richiesta dell'ordine di produzione fino alla consegna del veicolo.

### **Summary Report Yearly:**

- Promptdi ricerca
- Prompt filtri di dimensioni
- Dimensioni :Year e Point of View
- 2 view diverse: Year view, Season view
- Tabella doppia entrata visualizza I valori dei diversi KPI lungo la dimensione temporale con la possibilità di vedere il cumulate(year to date) e il periodico (month to date)
- Grafici dei KPI Year CY vs. PY

### **Year and BDG Compare:**

- Prompt filtri di dimensioni e KPI
- Promptdi ricerca
- Tabella KPI confrontati negli anni con grafico
- Tabella KPI confronto fra budget, forecast e actual con grafico

### **Orders Details:**

- Promptdi ricerca
- Prompt filtri di dimensioni
- Dettagli dei ordini

### **Unit Details:**

- Tutti i dettagli di un singolo ordine

Sono stati usati dei mockup per dare l'idea all'utente di quale sarà la struttura dei report finali



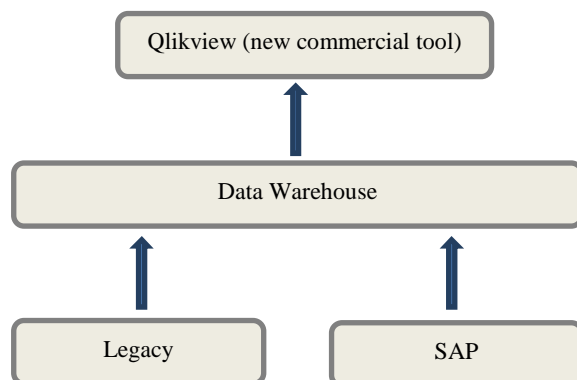
Report	Report Category	Notes
Home Page	-	
Summary Report	Event Report	Allows jump to Order Detail
Status Analyzer	Event Report	Free selecting functionalities
Dealer & Company Stock	Status Report	Allows jump to Order Detail
Order Status	Status Report	Allows jump to Order Detail
Orders Not Built	Status Report	Allows jump to Order Detail
Unit Detail	Master Data Report	
Configuration	Master Data Report	
Order Detail	Master Data Report	

## 6.4 - Originedeidati

Il sistema informativo del cliente è fatto da un Data Warehouse, composto da diversi Data Mart.

**Il Data Warehouse si alimenta da:**

- sistemi gestionale Legacy;
- sistemi gestionale SAP.



### 6.4.1 - Modello logico dei dati e di analisi:

**Modello logico dei dati:**

- analisi del portafoglio ordini
- analisi del Budget e Forecast
- composizione:
  - logica dei calcoli
  - KPI (di base e derivati)
  - dimensioni (gerarchie, attributi e date)

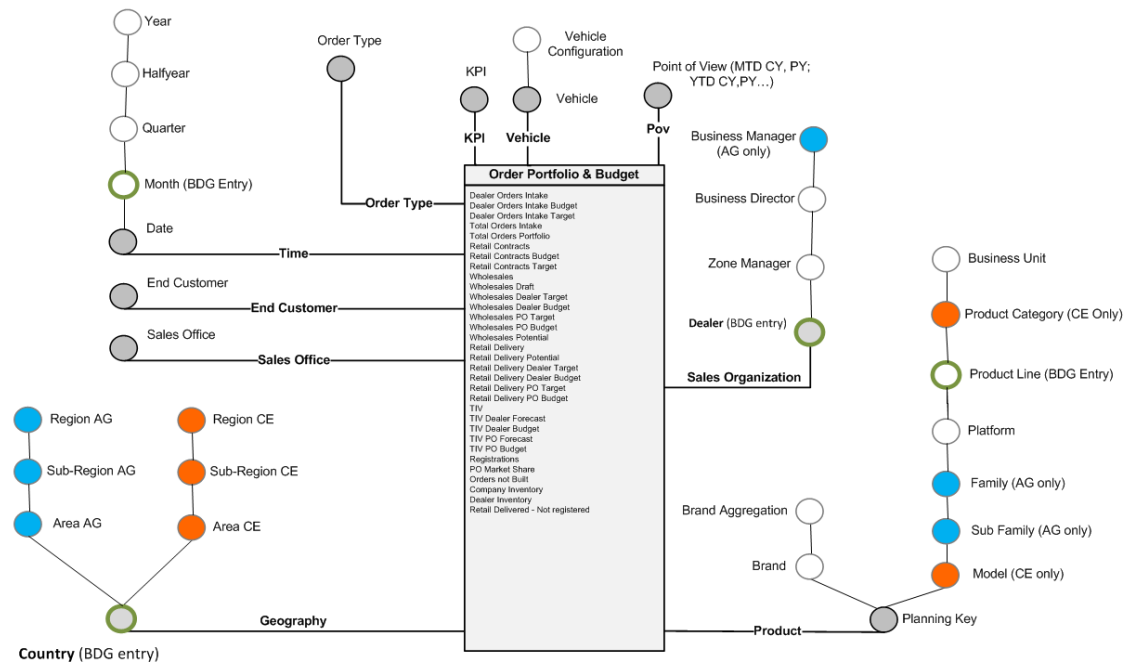
**Il DFM (data fact model)** è un modello concettuale grafico per data mart, pensato per:

- supportare efficacemente il progetto concettuale;
- creare un ambiente su cui formulare in modo intuitivo le interrogazioni dell'utente;
- permettere il dialogo tra progettista e utente finale per raffinare le specifiche dei requisiti;



- creare una piattaforma stabile da cui partire per il progetto logico (indipendentemente dal modello logico target);
- restituire una documentazione a posteriori espressiva e non ambigua.

La rappresentazione concettuale generata dal DFM consiste in un insieme di schemi di fatto (uno solo nel nostro caso). Gli elementi di base modellati dagli schemi di fatto sono i fatti, le misure, le dimensioni e le gerarchie.



### Modello di analisi:

- analisi di stato: monitorare il numero dei veicoli a un certo istante:
  - analisi attualizzate;
  - analisi storiche,
- analisi di evento: monitorare il numero di veicoli ordinati, venduti e consegnati in un periodo;
- anno corrente vs. anno precedente (time trend);
- analisi di Actual vs. Budget : solo per KPI di evento.

### KPI:

- KPI di stato : numero di veicoli in certo istante in un certo stato (es. numeri di veicoli prodotti ma non ancora spediti in questo istante );
- KPI di evento: la somma di un singolo evento (es. vendite) in un periodo.

## 6.5 - Estrazione dei dati

Il nostro strumento si aggancia al Data Mart del portafoglio ordini.

**Le tabelle di interesse sono:**

- FactTable;
- dimensione prodotto;
- dimensione mercato;
- dimensione veicolo;
- dimensione tipo di ordine.

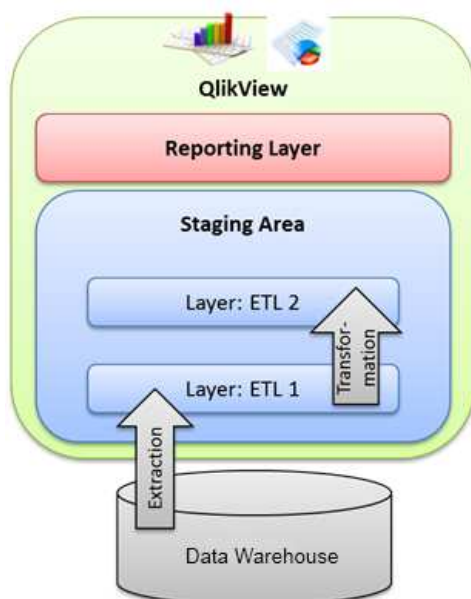
**La prima fase del ETL:**

- estrazione dei dati dal Data Warehouse;
- memorizzazione di ogni singola tabella in un file “.qvd” (QlikView);

**Le informazioni che servono per estrarre i dati:**

- stringa di connessione al Data Base;
- le queries di ciascuna tabella;
- i nomi nuovi delle tabelle;

Tutte le informazioni sono salvate in un file excel. (Best practise)



## 6.6 - Nuvola dati

Per costruire la nuvola dati o meglio ETL2 dobbiamo:

- capire i punti di forza di QlikView;
- scegliere lo schema adatto;
- trovare dei nomi parlanti per le tabelle e i campi;
- rinominare i campi in caso non vogliamo che le tabelle si associno in automatico.

### QlikView:

- è più efficiente su tabelle con pochi campi e tanti record (estensione verticali);
- lo schema è indifferente dipendendo dai tipi di dati.

### Schema:

- Star schema: più efficace quando l'informazione serve non aggregata;
- Snowflake schema: più efficace quando l'informazione serve aggregata.

### Nomi:

- devono essere parlanti;
- devono descrivere la tabella o il campo in maniera chiara;
- nomi diversi per campi simili se non vogliamo che le tabelle vengano in associazione in quel campo.

Sorgente i file .qvd creati in ETL1

## 6.7 - Analisi dello sviluppo progettuale

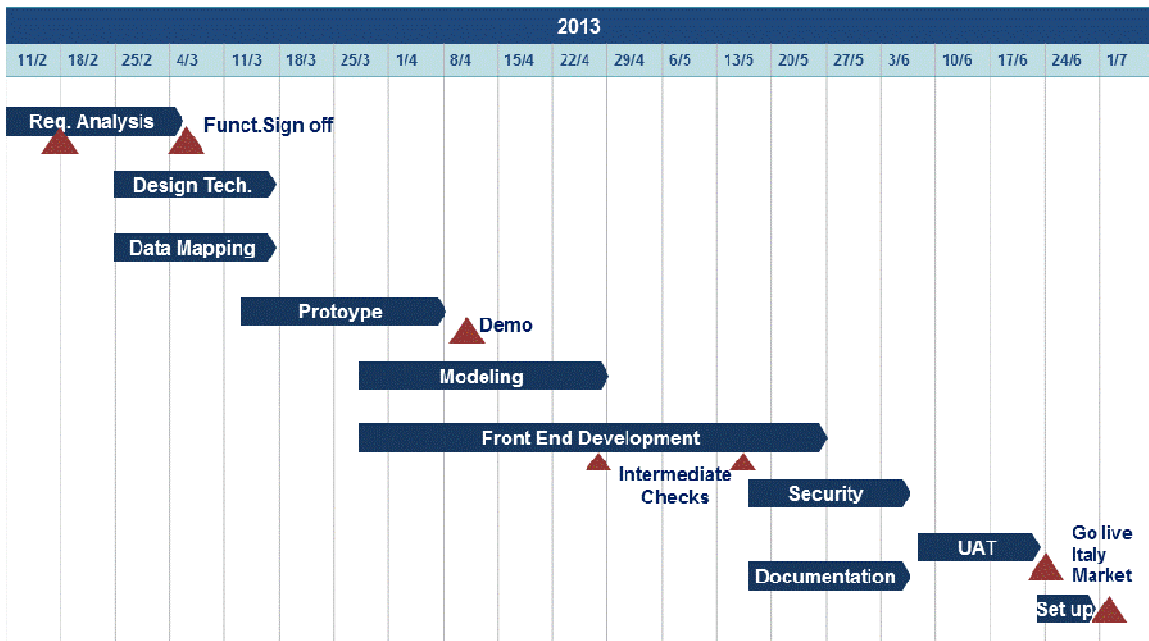
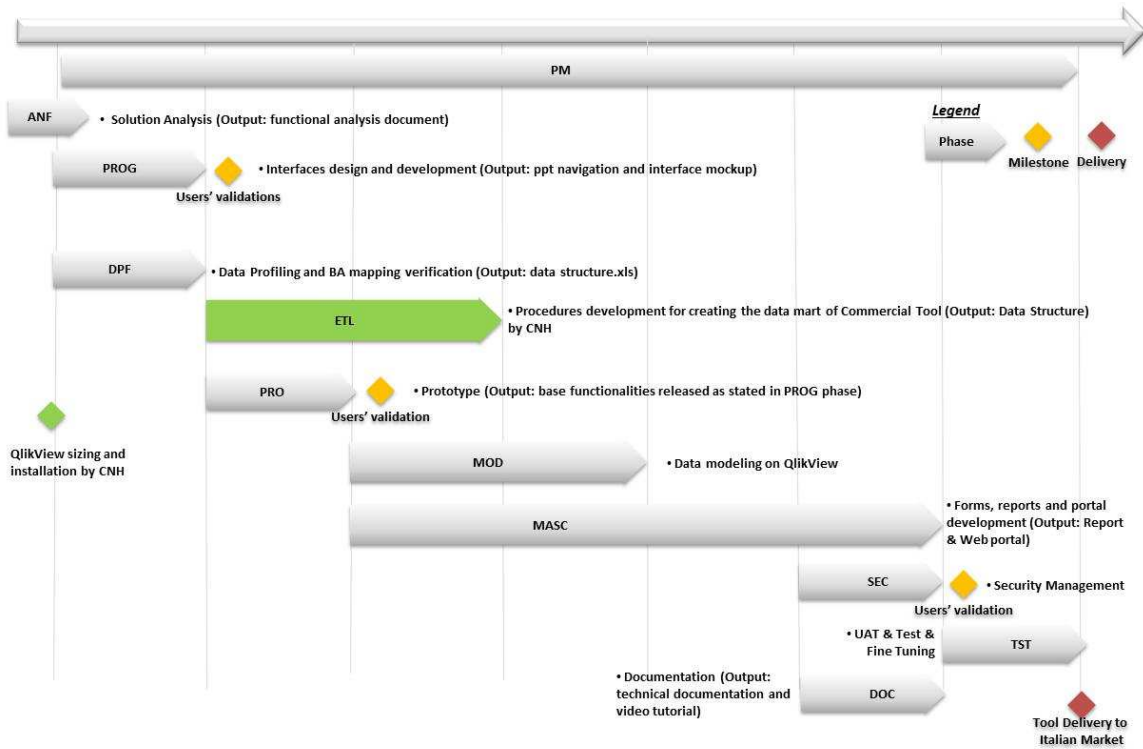
### 6.7.1 - Stima delle attività

Alla fine il progetto è stato chiuso nei tempi previsti con molta soddisfazione da parte del cliente.

La stima delle attività:

- **Analisi Soluzione.** Analisi della soluzione proposta concordando le funzionalità che si volevano sviluppare ed eventuali difficoltà progettuali. Durante questa fase sono state prodotte una serie di POC(Proof of concept), utili a concordare aspetti grafici e stilistici del prodotto da sviluppare.
- **ModelingDataMart Analitico.** Sarà creato il modello analitico (DFM) con focus sui KPI e Dimensioni di Analisi.
- **Staging Area.** Saranno importate le tabelle di interesse e sarà eseguita la Staging area direttamente all'interno della nuvola di QlikView.
- **Definizione Metadati:** Sarà definita la nuvola dei metadati del modello. In questa fase saranno anche effettuati dei test di coerenza del modello e delle regole di aggregazione.
- **Sviluppo Layout Reporting.** Sarà sviluppato il layout dei report.
- **Realizzazione Reports e UAT:** Saranno sviluppati esattamente i report richiesti nel documento di analisi funzionale. Iconulting eseguirà dei test di coerenza, ma sarà demandato agli utenti finali il test finale (UAT) e l'accettazione del modello. La gestione degli UAT sarà a carico dell'IT Corporate.
- **Documentazione Tecnica e Manuale Utente (Ingelse).** Sarà prodotta la documentazione tecnica relativa alla soluzione secondo le specifiche precedentemente concordate.
- **Formazione.** Saranno introdotti agli utenti gli strumenti di analisi (QlikView) e saranno illustrati KPI e Reports in un corso.

# Diagramma di Gantt:



## 6.7.2 - Project management

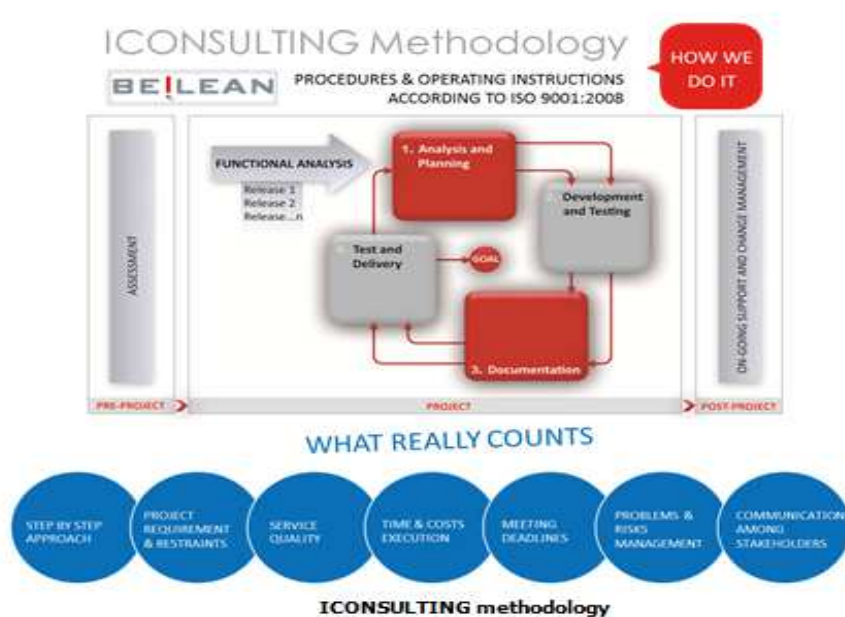
Iconsultingha sviluppatouna metodologia internaper la gestione deiprogetti sulla basedelle esperienzesvoltein oltre300progetti di BIE diEPM.

Dalprimo utilizzodella nostra esperienzaper identificarele esigenze del clientefino alla consegnadell'ultimarelease,la metodologiasupportaun ambiente operativoche consenteal Cliente diottenere ivantaggiattesi nelmodo migliore, piùveloce e piùconveniente.

Sullabase delleinformazioni acquisite durantele riunionipreliminari eleesperienze passatedacontesti simili, Iconsultingpropone unapproccio diprogetto dedicatoalla realizzazionedel sistema di Business Intelligence.

I vantaggi offerti dallametodologiadiIconsultingsono:

- focus suobiettiviimmediatamentericonoscibiliper mezzo diun processo di analisicondivisa conl'utente finale;
- ottimizzazione del tempodi riconoscimento dei requisitigrazie alla disponibilitàdi un modello didati configurato, aperto anchead eventualirichieste di modificavoltia migliorareil processo di caricamentodei dati.
- la rapida di attuazionebasato su un approccioiterativo direalizzazione/configurazione.
- Il framework metodologico di tutti i passi della progettazione e descritto in figura:

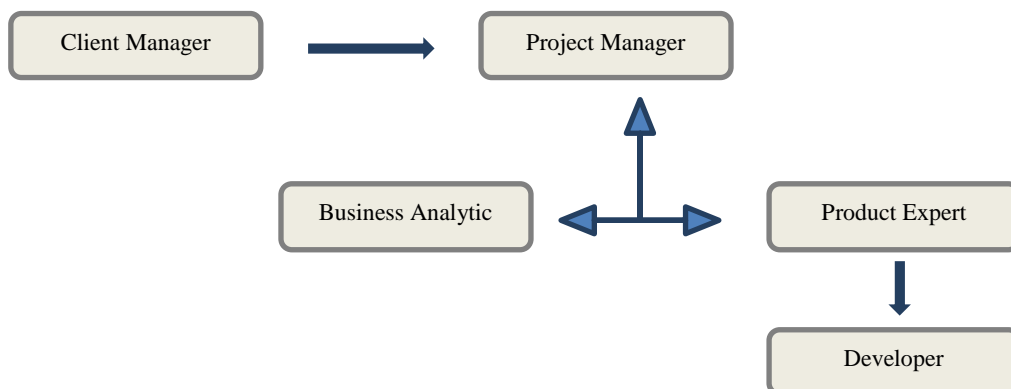


Struttura del team di progetto

## Ruoli:

- Sponsor : Cliente
- Client manager , ICONSULTING. Gestisce il rapporto con il cliente
- Project Manager , ICONSULTING che Cliente. Gestisce il progetto, responsabile dei tempi, delle risorse e dei raggiungimenti degli obiettivi.
- Program Manager, Cliente. Gestisce le risorse l'andamento del progetto.
- Staff
- Business Analytic, ICONSULTING. Responsabile della funzionalità e del contenuto del progetto.
- Product Expert, ICONSULTING. Responsabile del implementazione delle soluzioni e del andamento di sviluppo.
- Developer, ICONSULTING. Sviluppo tecnico del progetto.

## Struttura del team di Iconsulting:



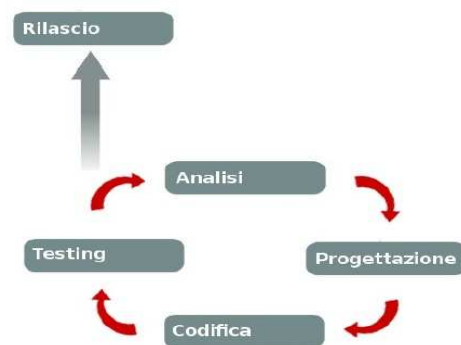
### 6.7.3 - La Metodologia Agile

Nell'ingegneria del software, per metodologia agile (o leggera) o metodo agile si intende un particolare metodo per lo sviluppo del software che coinvolge quanto più possibile il committente, ottenendo in tal modo un'elevata reattività alle sue richieste.

Ad ogni ciclo corrisponde il rilascio di una o un insieme di funzionalità. Il ciclosicompone di 4 fasi principali:

- **Analisi** : fase di analisi dei requisiti e delle funzionalità del sistema o della componente da sviluppare;
- **Progettazione** : viene progettata la funzionalità precedentemente definita in fase di analisi, valutandone la complessità e gli impatti sull'intero sistema.
- **Codifica** : viene generato il codice che implementa le funzionalità, concordate in fase di analisi e progettate in fase progettuale;
- **Testing** : si effettuano i test per valutare l'efficacia del sistema e per assicurarsi che l'informazione fornita sia consistente a quella desiderata.
- **Rilascio** : viene rilasciata la funzionalità all'utente, coinvolgendolo nel processo di controllo ed utilizzo della stessa per cercare di individuare eventuali bug o inconsistenze, e risolvere prima possibile.

#### Iterazioni e incremento di valore



L'intento di una programmazione di tipo Agile è la piena soddisfazione del cliente e non solo l'adempimento di un contratto. L'uso di queste metodologie, inoltre, serve ad abbattere i costi di sviluppo del software.



*I principi* su cui si basa una metodologia leggera che segua i punti indicati dall'Agile Manifesto, sono:

- La nostra massima priorità è soddisfare il cliente rilasciando software di **valore**, fin da subito e in maniera continua.
- Accogliamo i **cambiamenti** nei requisiti, anche a stadi avanzati dello sviluppo. I processi agili sfruttano il cambiamento a favore del vantaggio competitivo del cliente.
- **Consegniamo frequentemente** software funzionante, con cadenza variabile da un paio di settimane a un paio di mesi, preferendo i periodi brevi.
- Committenti e sviluppatori devono **lavorare insieme** quotidianamente per tutta la durata del progetto.
- Fondiamo i progetti su **individui motivati**. Diamo loro l'ambiente e il supporto di cui hanno bisogno e confidiamo nella loro capacità di portare il lavoro a termine.
- Una conversazione **faccia a faccia** è il modo più efficiente e più efficace per comunicare con il team ed all'interno del team.
- Il software funzionante è il principale metro di **misura di progresso**.
- I processi agili promuovono uno sviluppo sostenibile. Gli sponsor, gli sviluppatori e gli utenti dovrebbero essere in grado di mantenere indefinitamente un **ritmo costante**.
- La continua **attenzione all'eccellenza** tecnica e alla buona progettazione esaltano l'agilità.
- La **semplicità** - l'arte di massimizzare la quantità di lavoro non svolto - è essenziale.
- Le architetture, i requisiti e la progettazione migliori emergono da team che si **auto-organizzano**.
- A intervalli regolari il **team riflette** su come diventare più efficace, dopodiché regola e adatta il proprio comportamento di conseguenza.

## 6.8 - Sviluppo del Commercial Tool

In questa fase descriviamo nel dettaglio le procedure adottate per lo sviluppo e l'implementazione del commercial tool precedentemente descritto.

Lo sviluppo del commercial tool è stato fatto sfruttando una macchina remota messa a disposizione dal cliente, con i tutti software necessari per lo sviluppo.

Software utilizzati:

- *QlikView 11.20 R2*. Questo è il software utilizzato per lo sviluppo e la distribuzione dell'applicazione.
- *Oracle SQL Developer*. Esso è usato per connettersi al Data Warehouse ed effettuare le query sul DB. Inoltre ci ha permesso di testare il codice sql sviluppato per l'applicazione (vedi fase ETL1).
- *Microsoft Excel 2010*. In quanto alcuni vecchi report erano sviluppati in Excel, è usato come strumento di supporto per fare le quadrature dei dati. Inoltre ci è servito per costruire i file di configurazione (vedi cartella Settings).
- *Microsoft Access 2010*. In quanto il loro vecchio commercial tool era sviluppato in Access, è stato usato come strumento di supporto per fare le quadrature dei dati. Inoltre ci è servito come punto di riferimento per diversi report.

### 6.8.1 - Solution Analysis

Durante questa prima fase l'attività svolta ha riguardato due aspetti principali:

- l'analisi funzionale;
- i test sui dati.

#### Analisi Funzionale

Questo punto è stato portato avanti in stretta concomitanza col cliente, il quale rappresenta il maggior interessato agli aspetti funzionali del sistema. Per concordare le funzionalità sono stati sviluppati dei prototipi di modelli che sono serviti a dare l'idea di cosa e come il cliente voleva che l'applicazione visualizzasse le informazioni e, allo stesso tempo, ci hanno permesso di capire dove erano maggiormente concentrate le criticità del commercial tool, in modo da focalizzare maggiormente l'attenzione.

Per ogni report sono stati precedentemente costruiti dei prototipi utili a chiarire in questa fase gli aspetti precedentemente descritti.

### **Test sui dati**

In questa fase abbiamo effettuato dei test diretti sulla sorgente dati.

Tramite connessione remota al server di sviluppo, abbiamo potuto effettuare analisi dirette sui dati contenuti nel Data Warehouse tramite query in SQL.

È stata effettuata una prima fase di analisi di conformità dei dati e di ricerca delle tabelle da coinvolgere nel commercial tool.

Attenzione, dall'analisi effettuata e da quello che ci è stato comunicato dai sistemisti, abbiamo preso atto che i dati presenti nel commercial tool non sono tutti pienamente consistenti.

## **6.8.2 - Modellazione dell'ETL1**

Questa fase di sviluppo, è stata impostata partendo dalle riflessioni e dai risultati dell'analisi emerse nella prima fase. Infatti in questa fase è stata impostata e sviluppata la parte di Connection and Extraction delle tabelle all'interno della fase di ETL01.

Le tabelle estratte sono le seguenti:

- FactTable;
- dimensione prodotto;
- dimensione mercato;
- dimensione veicolo;
- dimensione tipo di ordine.

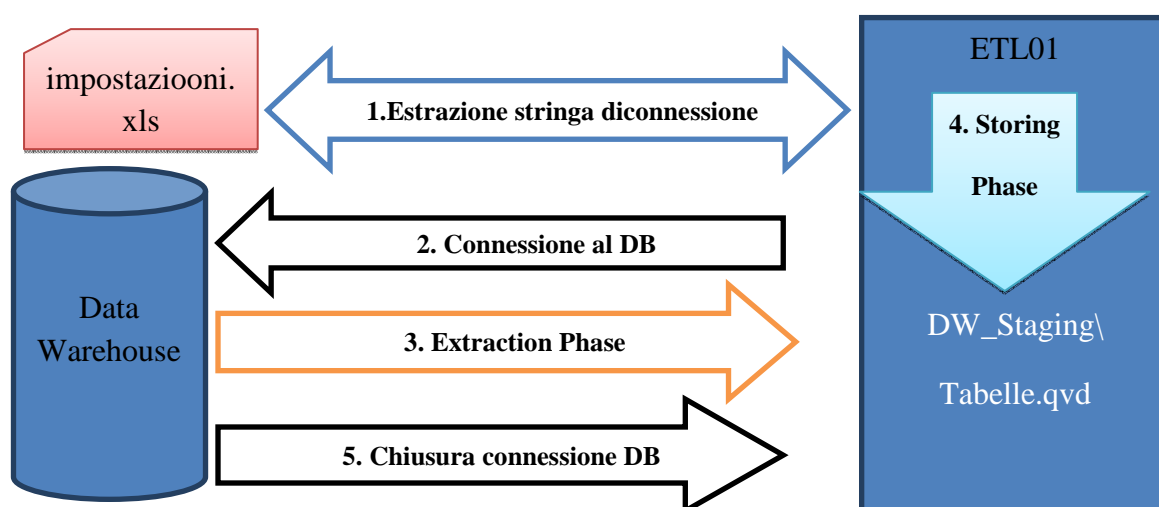
Il flusso delle operazioni, compute in fase di ETL01, è il seguente:

1. lettura stringa di connessione dal file setting;
2. connessione al Data Warehouse;
3. fase di Estrazione dei dati (ExtractionPhase) ;
4. fase di salvataggio delle tabelle nella cartella DW\_Staging (Storingphase);

5. chiusura connessione.

Come si nota chiaramente l'estrazione avviene una sola volta nella fase di ETL01, questo significa che in questo modo le operazioni di estrazione vengono centralizzate in unico file secondo una specifica sequenza, in modo tale da facilitare eventuali gestioni e manutenzioni del sistema. Il tecnico manutentore saprà esattamente che in caso di problemi di estrazione dovrà controllare esclusivamente quel file senza preoccuparsi degli altri (QlikView best practices).

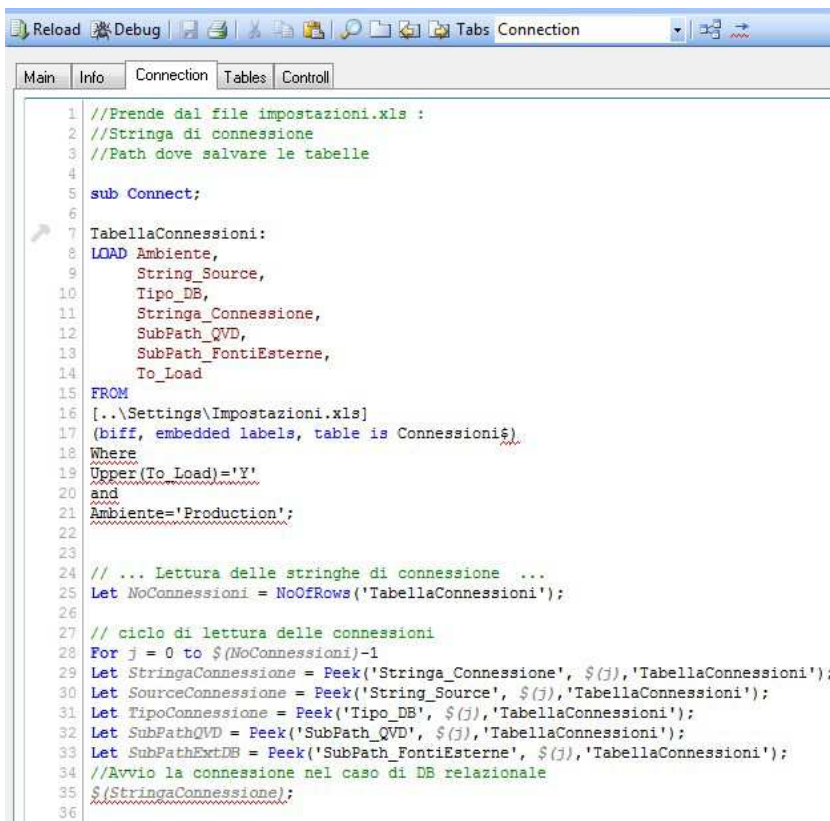
Il risultato è la creazione dei file “.qvd”. Esso conterranno le tabelle originali importate dal Data Warehouse. I file serviranno come sorgente per l' ETL2.



Il file Impostazioni.xls foglio Connessioni:

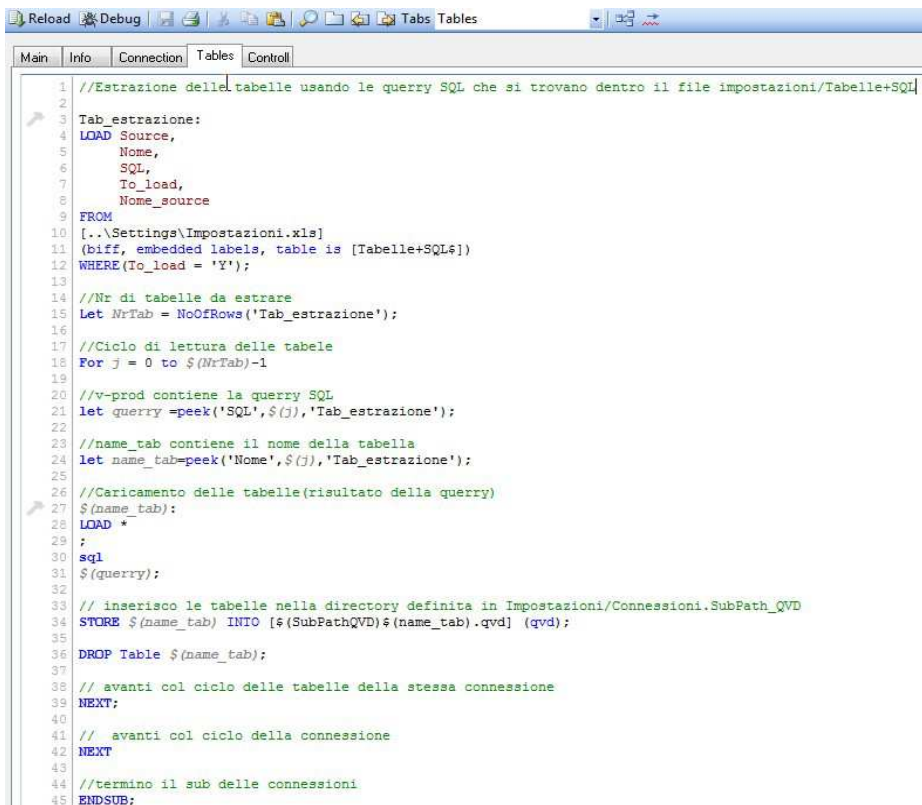
Ambiente	String_Source	Tipo_DB	Stringa_Connessione	SubPath_QVD	SubPath_FontiEsterne
Nome ambiente	Stringa della sorgente DB	Tipo di Database utilizzato	Stringa di connessione per il DB	Path dove salvare i file QV DATA:	Path fontiesterne:
Quality	SQL-Oracle	Relazionale	OleDb Connect To	..\DW_Staging\	NULL
Production	SQL-Oracle	Relazionale	OleDb Connect To	..\DW_Staging\	NULL
Staging	QVD	Relazionale		..\DW_Associative\	..\DW_Staging\
Production	SQL-Oracle	Relazionale	OleDb Connect To	..\DW_Staging\	NULL

## AQL script : Importa i dati della Connessione dal file Impostazioni.xls



```
1 //Prende dal file impostazioni.xls :
2 //Stringa di connessione
3 //Path dove salvare le tabelle
4
5 sub Connect;
6
7 TabellaConnessioni:
8 LOAD Ambiente,
9     String_Source,
10    Tipo_DB,
11    Stringa_Connessione,
12    SubPath_QVD,
13    SubPath_FontiEsterne,
14    To_Load
15 FROM
16 [..\Settings\Impostazioni.xls]
17 (biff, embedded labels, table is Connessioni$)
18 Where
19 Upper(To_Load)='Y'
20 and
21 Ambiente='Production';
22
23
24 // ... Lettura delle stringhe di connessione ...
25 Let NoConnessioni = NoOfRows('TabellaConnessioni');
26
27 // ciclo di lettura delle connessioni
28 For j = 0 to $(NoConnessioni)-1
29 Let StringaConnessione = Peek('Stringa_Connessione', $(j),'TabellaConnessioni');
30 Let SourceConnessione = Peek('String_Source', $(j),'TabellaConnessioni');
31 Let TipoConnessione = Peek('Tipo_DB', $(j),'TabellaConnessioni');
32 Let SubPathQVD = Peek('SubPath_QVD', $(j),'TabellaConnessioni');
33 Let SubPathExtDB = Peek('SubPath_FontiEsterne', $(j),'TabellaConnessioni');
34 //Avvio la connessione nel caso di DB relazionale
35 $(StringaConnessione);
36
```

## AQL script : Importa i le query dal file Impostazioni.xls e salva le tabelle nel formato QlikView “.qvd”



```
1 //Estrazione delle tabelle usando le query SQL che si trovano dentro il file impostazioni/Tabella+SQL
2
3 Tab_estrazione:
4 LOAD Source,
5     Nome,
6     SQL,
7     To_load,
8     Nome_source
9 FROM
10 [..\Settings\Impostazioni.xls]
11 (biff, embedded labels, table is [Tabella+SQL$])
12 WHERE (To_load = 'Y');
13
14 //Nr di tabelle da estrarre
15 Let NrTab = NoOfRows('Tab_estrazione');
16
17 //Ciclo di lettura delle tabelle
18 For j = 0 to $(NrTab)-1
19
20 //v-prod contiene la query SQL
21 let query = peek('SQL', $(j), 'Tab_estrazione');
22
23 //name_tab contiene il nome della tabella
24 let name_tab = peek('Nome', $(j), 'Tab_estrazione');
25
26 //Caricamento delle tabelle (risultato della query)
27 $(name_tab):
28 LOAD *
29 ;
30 sql
31 $(query);
32
33 // inserisco le tabelle nella directory definita in Impostazioni/Connessioni.SubPath_QVD
34 STORE $(name_tab) INTO [$(SubPathQVD)$(name_tab).qvd] (qvd);
35
36 PROP Table $(name_tab);
37
38 // avanti col ciclo delle tabelle della stessa connessione
39 NEXT;
40
41 // avanti col ciclo della connessione
42 NEXT
43
44 //termino il sub delle connessioni
45 ENDSUB;
```

### 6.8.3 - Modellazione dell' ETL2

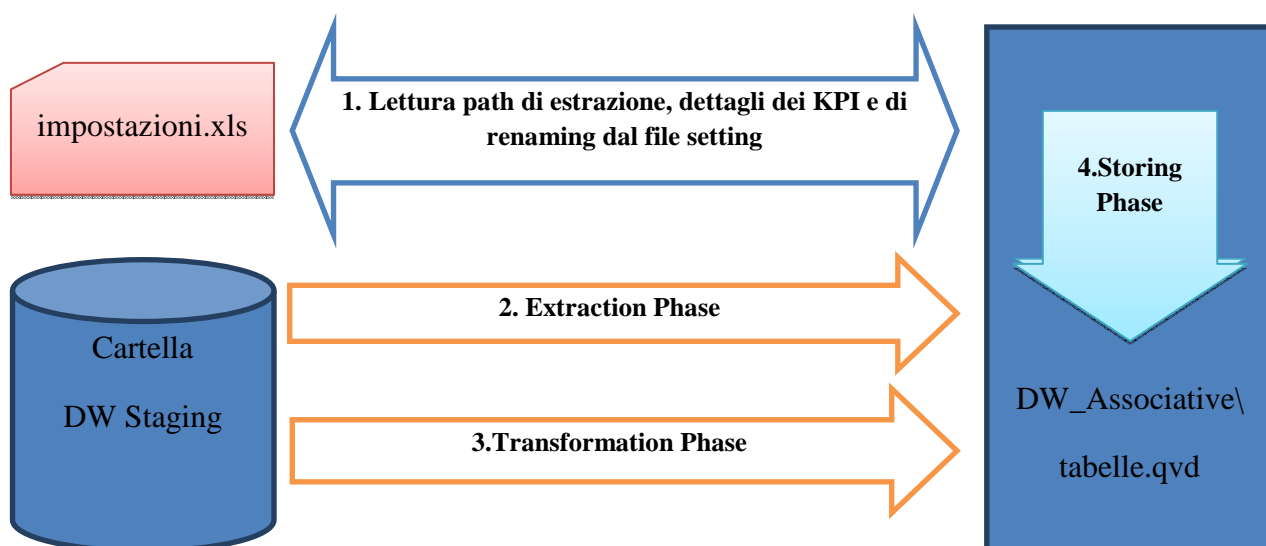
Durante la fase di **ETL2** verranno effettuate le operazioni di :

- costruzione del modello interessato, secondo le dimensioni e le gerarchie che abbiamo descritto in precedenza;
- rinomina delle tabelle e dei campi (impostazioni.xls)

Questa è la fase con maggior criticità per l'intero progetto, in quanto viene interessata non solo per le prime operazioni di costruzione del modello, ma per tutte le operazioni di modifica successive. Infatti, dopo il rilascio di ogni report, le modifiche e obiezioni del cliente hanno comportato variazioni su questo livello. Per questo motivo si è deciso di investire maggior tempo per organizzare al meglio e nel modo più chiaro possibile il codice di **ETL2**.

Il flusso delle operazioni, compute in fase di ETL02, è il seguente:

1. Lettura path di estrazione, dettagli dei KPI e di renaming dal file setting;
2. Estrazione delle tabelle da DW\_Staging (ExtractionPhase);
3. Fase di Trasformazione dei dati (TransformationPhase);
4. Fase di salvataggio delle tabelle nella cartella DW\_Associative (Storingphase);



Per permettere il funzionamento della logica di QlikView, i nomi delle chiavi delle tabelle sono stati rinominati allo stesso modo, così potranno andare in associazione fra di loro automaticamente.

AQL script : Importa i dettagli dei KPI dal file Impostazioni.xls e costruisce la fact\_table:

```

Main Info KPI Dimension Commercial Orders Time KPI Dimension Renaming Fields Controll
1 Directory;
2
3 SUB KPI;
4
5 KPI:
6 LOAD KPI,
7     KPI_Name,
8     KPI_Month
9 FROM
10 [..\Settings\Impostazioni.xlsx]
11 (coxml, embedded labels, table is KPI);
12
13 //Campi comuni a tutti i KPI
14 let fk='Planning_Key_Identifier, Order_Type_Identifier, Vehicle_Identifier,
15     | Sales_Org_Id_Internal, Technical_Type_Identifier, Dealer_Identifier';
16
17 let nr=NoOfRows('KPI');
18
19 //Loop per concatenare tutti gli KPI dentro "Commercial Orders"
20 for i=0 to $(nr)-1
21
22 Let KPI = peek('KPI', $(i), 'KPI');
23 Let KPI_Name = peek('KPI_Name', $(i), 'KPI');
24 Let KPI_Month = peek('KPI_Month', $(i), 'KPI');
25
26 "Commercial Orders":
27 LOAD $(fk),
28     $(KPI_Month) as YearMonth,
29     '$(KPI_Name)' as KPI,
30     $(KPI) as Value
31 FROM
32 [..\DW_Staging\CommercialOrders.qvd]
33 (qvd)
34 where $(KPI)<>0;
35
36 NEXT
37
38 DROP table KPI;
39
40 ENDSUB;
41

```

AQL script : Importa del renaming dal file Impostazioni.xls e salva le tabelle in “.qvd”:

```

Main Info KPI Dimension Commercial Orders Time KPI Dimension Renaming Fields Controll
1 SUB Renaming;
2
3 Renaming_Attribute:
4 Mapping LOAD Attribute_Name,
5     Attribute_Rename
6 FROM
7 [..\Settings\Impostazioni.xlsx] (coxml, embedded labels, table is Renaming_Fields)
8 WHERE (To_load = 'Y');
9
10 Rename Fields using Renaming_Attribute;
11
12 //Store di tutte le tabelle
13 STORE "KPI Home" INTO [..\DW_Associative\KPI Home.qvd] (qvd);
14 STORE "KPI OrdSummary" INTO [..\DW_Associative\KPI OrdSummary.qvd] (qvd);
15 STORE "KPI StatusReport" INTO [..\DW_Associative\KPI StatusReport.qvd] (qvd);
16 STORE "KPI SumYearly" INTO [..\DW_Associative\KPI SumYearly.qvd] (qvd);
17 STORE "KPI Versus" INTO [..\DW_Associative\KPI Versus.qvd] (qvd);
18
19 STORE "Commercial Orders" INTO [..\DW_Associative\Commercial Orders.qvd] (qvd);
20
21 STORE "Product" INTO [..\DW_Associative\Product.qvd] (qvd);
22 STORE "Sales Organization" INTO [..\DW_Associative\Sales Organization.qvd] (qvd);
23 STORE "Vehicle" INTO [..\DW_Associative\Vehicle.qvd] (qvd);
24 STORE "Order Type" INTO [..\DW_Associative\Order Type.qvd] (qvd);
25
26 //Drop di tutte le tabelle dopo essere state salvate nei file
27 drop tables "KPI Home",
28     "KPI OrdSummary", "KPI StatusReport", "KPI SumYearly", "KPI Versus",
29     "Commercial Orders", "Product", "Sales Organization", "Vehicle", "Order Type";
30
31 endsub;
32

```

## 6.8.4 - Nuvola Dati

Durante la fase di **ETL2** vera costruita la **Nuvola Dati**.

La nuvola di QlikView non è altro che il modello di metadati generato dal software e che gli permette di effettuare le elaborazioni sui dati. La nuvola viene interamente caricata in memoria principale così da permettere rapidità ed efficienza nell'elaborazione dei dati.

Seguendo le Best practices QlikView ha migliori performance su tabelle con poche dimensioni e tante righe (estensione verticale) piuttosto che tante dimensioni e poche righe (estensione orizzontale).

### **Facttable:**

Trasposizione della tabella lungo la dimensione KPI:

- creazione della dimensione KPI ;
- campo KPI\_ID : chiave esterna tabella KPI;
- campo Time Identifier : chiave esterna tabella Time.

### **KPI dimension:**

Contiene:

- KPI\_ID
- lingua
- livello del KPI
- nomi KPI
- report di appartenenza
- L'ordine in un report

### **Time dimension:**

Contiene:

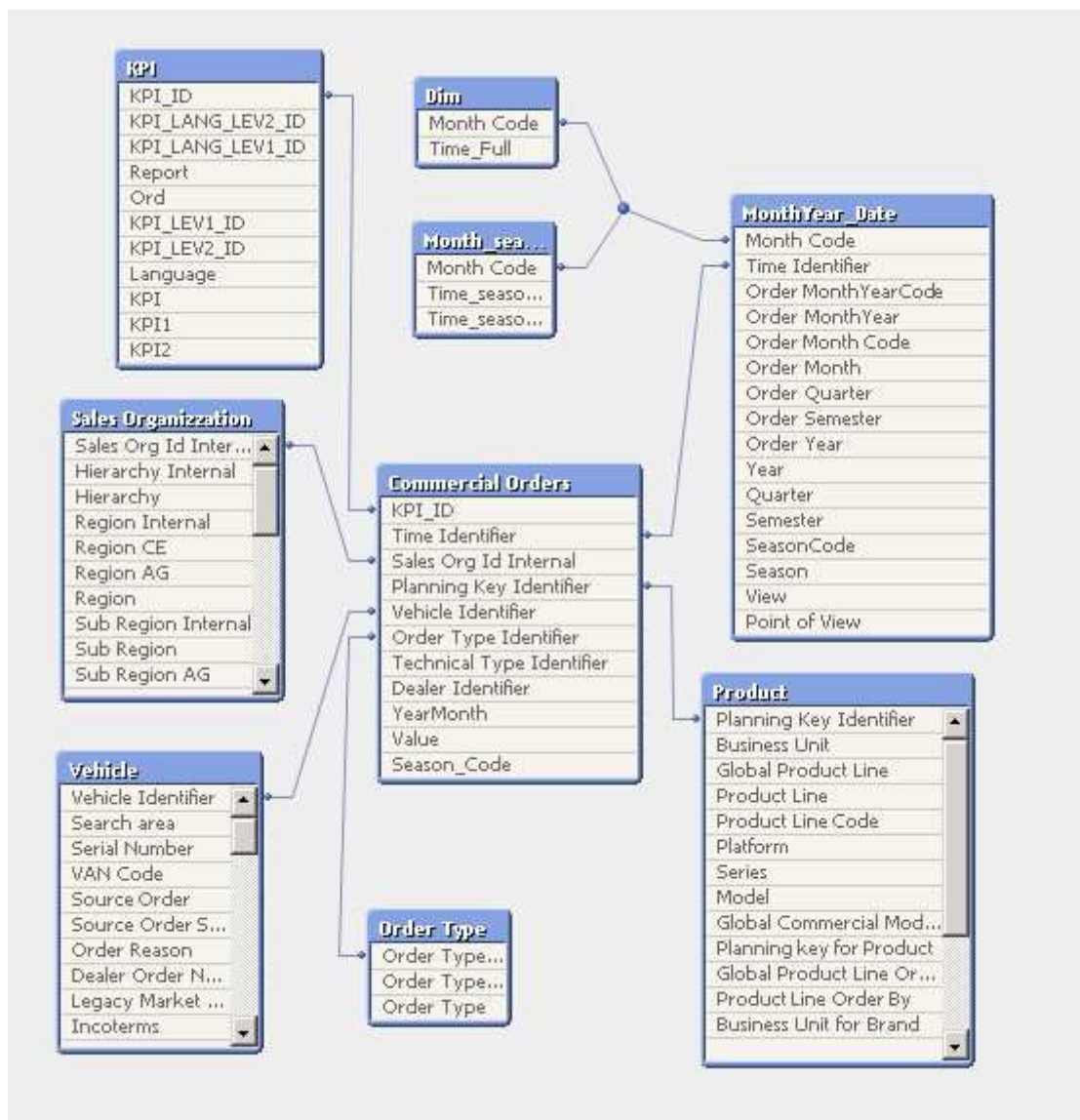
- Time Identifier
- ordine



- codice mese
- anno
- semestre
- trimestre
- stagione
- View : Periodo o Stagione
- Point of View : Periodo o Cumulato

La nuvola dati:

Star schema con una facttable centrale e 6 dimensioni.



## 6.9 - Presentation del commercial tool

La presentation è l'insieme di tutti i report messi a disposizione del cliente.

Lo sviluppo dei report è stato il momento in cui l'utente ha potuto tangere con mano i risultati del lavoro.

Questa fase ha generato il picco massimo livello di impegno sul progetto, in quanto, con rilasci periodici, si è provveduto a rilasciare report nuovi e allo stesso tempo a recuperare e risolvere bug o problemi relativi ai report precedenti.

### **Sicurezza:**

Il file "Presentation.qvw" verrà pubblicato sul QlikView Publisher e potrà essere consultato o tramite un internet browser o tramite l'applicazione QlikView per Ipad.

Ogni utente vedrà determinate informazioni in base al suo profilo.

### **Report:**

Il file "Presentation.qvw" contiene 7 report.

#### Struttura dei report

Tutti i report si compongono di una struttura di filtri fissa alla sinistra e altri filtri in base al report:

- Dimensioni
- Analisi libera
- Tempo : anno, mese
- Point of View
- Season View
- Search

Sono presenti tre elementi ben distinti:

- Filtri: i filtri riguardano principalmente tutte le dimensioni del modello, ed hanno la possibilità di essere applicati sul singolo elemento o su di una selezione multipla. Il filtro temporale può essere applicato o dal calendario, oppure selezionando direttamente anni e mesi. Il filtro point of view (period o cumulate);
- CurrentSelection: si trova in basso la currentselection ci dà l'informazione relativa ai filtri applicati;
- Search: si trova in alto a sinistra, l'utente può effettuare una ricerca globale su qualsiasi informazione;
- Tabrow (Report navigation) : si trova in alto, l'utente può navigare fra i report;
- Bottoni : si trovano in alto a destra, per informazioni esterne;
- Language bar: si trova in basso, per selezionare la lingua desiderata.

## 6.9.1 - Home Page

È il primo report che si visualizza all'utente, contiene:

- Un banner con l'ultima foto dei KPI principali
- Filtri delle dimensioni
- Filtri temporali : anno e mese
- Point of view : cumulate o period
- Analisi libera dimensioni (per aggiungere alla tabella principale)

La situazione predefinita:

- Lingua : English
- Anno : attuale
- Mese : attuale
- Country : country dell'utente loggato
- Point of view : Period
- Tabella con Zone Manager e i KPI della Home Page
- Due grafici (Bar chart) con Product Line e KPI

Tutti gli elementi della pagina sono sensibili a tutti i filtri, tranne il banner dei KPI che non è sensibile al filtro dell'anno.



## 6.9.2 - Summary Report Monthly

Il report contiene:

- Filtri delle dimensioni
- Analisi libera dimensioni (per aggiungere alla tabella principale)
- Due tabelle in mutua esclusione.

La situazione predefinita:

- Lingua : English
- Anno (non modificabile) : attuale
- Mese (non modificabile) : attuale
- Country : country dell'utente loggato
- Point of view (non modificabile) : Period
- Tabella con Zone Manager e i KPI
- Due grafici (Bar chart) con Product Line e KPI

Tutti gli elementi della pagina sono sensibili solo ai filtri applicati in questa pagina.

**Summary Report Monthly**

Report Description | Data Dictionary  
Mail to: | Getting Started  
Welcome B. Trajani | Sep 11, 2013

Source O... | Serial Nr. | Dealer Code | Order Summary | Status Report

Search

**Order Summary**

Product Line	KPI Lev 2			Total Order		Wholesale Qty		WS ACT vs. Last FCST		Wholesales Last FCST	
	KPI Lev 1	Total	Dealer	Company Stock	Total	Wholesale Qty	Total	WS ACT vs. Last FCST	Total	Wholesales Last FCST	
<b>Total</b>		<b>1.064</b>	<b>1.049</b>	<b>15</b>	<b>995</b>	<b>995</b>	<b>222</b>	<b>222</b>	<b>617</b>	<b>617</b>	
COMBINES		14	5	9	6	6	-3	3	3	3	
CORN HE...		-	-	-	9	9	-3	-3	1	1	
GRAIN H...		6	5	1	10	10	-1	-1	1	1	
LARGE S...		2	1	1	1	1	-1	-1	-	-	
SP FORA...		2	2	-	4	4	2	2	4	4	
SP GRAPE...		5	5	-	6	6	-3	-3	2	2	
SPFH HE...		2	2	-	3	3	-1	-1	-	-	
TELEHAN...		2	1	1	4	4	1	1	2	2	
TRACTORS		1.031	1.028	3	944	944	225	225	596	596	
UTILITY...		-	-	-	8	8	-	-	8	8	

English | Italiano | Français | Español | Português

Current Selection: Language: English | Month Code: 08 | View: Year View | Point of View: Period | Country: Italy

## 6.9.3 - Summary Report Yearly

Il report contiene:

- Filtri delle dimensioni
- Filtri temporali : anno e mese
- Filtri Season view
- Point of view : cumulate o period
- Analisi libera temporale : anni e messe (per aggiungere alla tabella principale)
- Analisi libera Point of view : period e cumulate (per aggiungere alla tabella principale)
- Analisi libera dimensioni (per aggiungere alla tabella principale)
- Una tabella principale
- Grafico (Line Chart): andamento e confronto temporale di 3 KPI

La situazione predefinita:

- Lingua : English
- Anno : dal 2008 al attuale
- Mese : tutti i mesi
- Country : country dell'utente loggato
- Point of view : tutti e due (period e cumolato)
- Tabella : con dimensione KPI e tempo
- Due grafici (Bar chart) con Product Line e KPI

Tutti gli elementi della pagina sono sensibili solo ai filtri applicati in questa pagina.

**Summary Report Yearly**

Report Description | Data Dictionary  
Mail to: | Getting Started  
Welcome B. Trajani | Sep 11, 2013

Year View | Season View NOV-DCT | Season View SEP-AUG

Summary Report Year View

KPI	Year	Point of View	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	FY	Q1	Q2	Q3	Q4	H1	H2	
Retail Delivery	2013	Period	183	325	457	340	419	621	123	-	-	-	-	-	-	2,468	965	1,380	123	-	2,345	12
		Cumulate	183	508	965	1,305	1,724	2,345	2,468	-	-	-	-	-	-	-	2,468	965	2,345	2,468	-	2,345
	2012	Period	265	341	522	385	483	597	401	254	346	367	403	683	5,047	1,128	1,465	1,001	1,453	2,593	2,45	
		Cumulate	265	606	1,128	1,513	1,996	2,593	2,994	3,248	3,594	3,961	4,364	5,047	5,047	1,128	2,593	3,594	5,047	2,593	5,04	
	2011	Period	263	476	749	422	617	710	421	265	479	428	433	642	5,905	1,488	1,749	1,165	1,503	3,237	2,66	
		Cumulate	263	739	1,488	1,910	2,527	3,237	3,658	3,923	4,402	4,830	5,263	5,905	5,905	1,488	3,237	4,402	5,905	3,237	5,90	
2010	Period	255	438	584	547	623	765	465	284	452	429	447	605	5,894	1,277	1,935	1,201	1,481	3,212	2,66		
	Cumulate	255	693	1,277	1,824	2,447	3,212	3,677	3,961	4,413	4,842	5,289	5,894	5,894	1,277	3,212	4,413	5,894	3,212	5,85		
2009	Period	291	391	609	634	794	810	467	312	435	432	471	542	6,188	1,291	2,238	1,214	1,445	3,529	2,65		
	Cumulate	291	682	1,291	1,925	2,719	3,529	3,996	4,308	4,743	5,175	5,646	6,188	6,188	1,291	3,529	4,743	6,188	3,529	6,18		
2008	Period	336	437	507	535	662	708	524	366	559	455	456	408	6,115	1,362	1,985	1,449	1,319	3,347	2,76		
	Cumulate	336	775	1,362	1,897	2,559	3,247	3,871	4,237	4,796	5,251	5,707	6,115	6,115	1,362	3,347	4,796	6,115	3,347	6,11		
Retail Delivery Last FCST	2013	Period	178	325	467	350	422	593	383	-	-	-	-	-	-	2,718	970	1,365	383	-	2,335	38
		Cumulate	178	503	970	1,320	1,742	2,335	2,718	-	-	-	-	-	-	-	2,718	970	2,335	2,718	-	2,335
2012	Period	265	346	511	400	483	578	404	255	341	369	387	664	5,003	1,122	1,461	1,000	1,420	2,583	2,42		
	Cumulate	265	611	1,122	1,522	2,005	2,583	2,987	3,242	3,583	3,952	4,339	5,003	5,003	1,122	2,583	3,583	5,003	2,583	5,00		
2009	Period	286	391	603	624	783	747	444	308	411	422	468	529	6,016	1,280	2,154	1,163	1,419	3,434	2,56		
	Cumulate	286	677	1,280	1,904	2,687	3,434	3,878	4,186	4,597	5,019	5,487	6,016	6,016	1,280	3,434	4,597	6,016	3,434	6,01		
2008	Period	346	454	601	551	654	755	491	351	547	453	456	412	6,071	1,401	1,960	1,389	1,321	3,361	2,71		
	Cumulate	346	800	1,401	1,952	2,606	3,361	3,852	4,203	4,750	5,203	5,659	6,071	6,071	1,401	3,361	4,750	6,071	3,361	6,07		
TIV Last FCST	2013	Period	78	114	128	159	170	62	57	-	-	-	-	-	-	768	320	391	57	-	711	5
		Cumulate	78	192	320	479	649	711	768	-	-	-	-	-	-	768	320	711	768	-	711	76
2012	Period	85	152	196	160	144	74	42	12	18	53	53	69	1,058	433	378	72	175	811	24		
	Cumulate	85	237	433	593	737	811	853	865	883	936	989	1,058	1,058	433	811	883	1,058	811	1,05		
2009	Period	1,976	2,254	2,568	2,477	2,745	3,302	3,058	2,224	2,351	1,913	1,736	1,532	28,136	6,798	8,524	7,633	5,181	15,322	12,81		
	Cumulate	1,976	4,230	6,798	9,275	12,020	15,322	18,380	20,604	22,955	24,868	26,604	28,136	28,136	6,798	15,322	22,955	28,136	15,322	28,13		

English | Italiano | Français | Español | Português

Current Selection: Language: English | View: Year View | Country: Italy

## 6.9.4 - Year & BDG compare

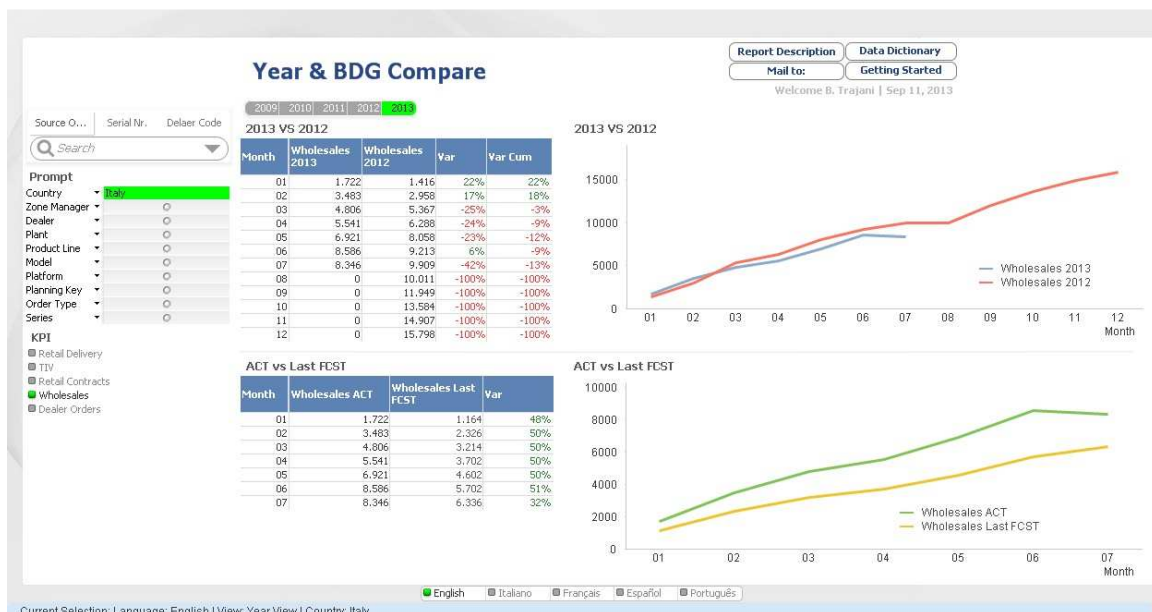
Il report contiene:

- Filtri delle dimensioni
- Filtro KPI
- Filtro Anni
- Due tabelle
- Due grafici (Line chat)

La situazione predefinita:

- Lingua : English
- Anno : attuale
- Country : country dell'utente loggato
- Point of view (non modificabile) : Period
- Tabella 1 : confronta l'anno attuale con quello precedente del KPI selezionato.
- Grafico 1 (Line chart) : visualizzazione grafica della tabella di sopra
- Tabella 2 : confronta l'Actual con il Budget del KPI selezionato
- Grafico 2 (Line chart) : visualizzazione grafica della tabella di sopra

Tutti gli elementi della pagina sono sensibili solo ai filtri applicati in questa pagina.



## 6.9.5 - Order Details

Il report contiene:

- Filtri delle dimensioni
- Analisi libera dimensioni (per aggiungere alla tabella principale)
- Analisi libera date (per aggiungere alla tabella principale)
- Tabella : Tutti i dettagli degli ordini

La situazione predefinita:

- Lingua : English
- Country : country dell'utente loggato
- Tabella : Tutti i dettagli degli ordini

Tutti gli elementi della pagina sono sensibili a tutti i filtri applicati anche nei altri report.

**Orders Details**

Report Description | Data Dictionary  
Mail to: | Getting Started  
Welcome B. Trajani | Sep 11, 2013

KPI	VAN Code	Product Line	Platform	Series	Model	Request Schedule	Schedule Date	OK On Pad Date	Zone Manager	Dealer	Vehicle Creation Date
	0050834351	TRACTORS	Compact	T3000	T3010	01/06/2013	17/06/2013	17/06/2013	Loris VALENTINI	PEGASO S.P.A.	21/03/2013
	0050464430	TELEHANDLE...	Telehandler	LMS000_D...	LMS020A	01/03/2012	16/04/2012	17/04/2012	Beniamino CAN...	RANUCCI S.R.L.	27/12/2011
	0050773098	GRAIN HEADERS	Combine H...	GH BUR NH	7830FD	01/02/2013	-	15/05/2013	Andrea MASSO...	AGRIMEC LAMI ...	27/09/2012
	0050773082	GRAIN HEADERS	Combine H...	GH BUR NH	7825FS	01/03/2013	18/03/2013	17/05/2013	Andrea MASSO...	AGRIMEC LAMI ...	27/09/2012
	0050127290	TRACTORS	APL	TS000	TS060	13/06/2011	28/06/2011	24/06/2011	Federico SULSE...	MOCICARO FR...	28/06/2013
	0050534634	TRACTORS	Specialty	TD400F	TD4040F	01/10/2012	22/10/2012	31/01/2013	Mario DI GIOIA	VIGORITO ANG...	02/07/2012
	0050541047	TRACTORS	Specialty	TD400F	TD4040F	01/11/2012	05/11/2012	05/11/2012	Giancarlo BIASI...	CONSORZIO A...	31/07/2012
	0050541041	TRACTORS	Specialty	TD400F	TD4030F	01/11/2012	22/11/2012	19/11/2012	Mario DI GIOIA	SETTANNI S. R. L.	31/07/2012
	0050541166	TRACTORS	Specialty	TD400F	TD4040F	01/11/2012	23/11/2012	19/11/2012	Paolo CRISCUOLO	CENTRO MACC...	31/07/2012
	0050541039	TRACTORS	Specialty	TD400F	TD4030F	01/11/2012	22/11/2012	21/11/2012	Claudio RAVA	EMMETI S.R.L.	31/07/2012
	0050541040	TRACTORS	Specialty	TD400F	TD4030F	01/11/2012	22/11/2012	29/11/2012	Paolo CRISCUOLO	CENTRO MACC...	31/07/2012
	0050541150	TRACTORS	Utility	POWERST...	PWRSTR 75	01/11/2012	22/11/2012	19/11/2012	-	CONS.AGR.DIP...	31/07/2012
	0050541157	TRACTORS	Utility	POWERST...	PWRSTR 75	01/11/2012	26/12/2012	14/12/2012	Giancarlo BIASI...	CONSORZIO A...	31/07/2012
	0050541140	TRACTORS	Utility	POWERST...	PWRSTR 75	01/11/2012	29/12/2012	20/12/2012	Paolo CRISCUOLO	TIFERNO MACC...	31/07/2012
	0050541141	TRACTORS	Utility	POWERST...	PWRSTR 75	01/11/2012	29/12/2012	18/12/2012	Paolo CRISCUOLO	TIFERNO MACC...	31/07/2012
	0050541139	TRACTORS	Utility	POWERST...	PWRSTR 75	01/11/2012	29/12/2012	24/12/2012	Mario DI GIOIA	CECAS COMM...	31/07/2012
	0050517651	TRACTORS	APH	T6_T4A	T6_150	03/09/2012	05/09/2012	04/09/2012	Paolo CRISCUOLO	CENTRO MACC...	20/04/2012
	0050630191	TRACTORS	APH	T6_T4A	T6_160	26/11/2012	26/11/2012	23/11/2012	Giancarlo BIASI...	CONS.AGR.LOM...	05/09/2012
	0050630197	TRACTORS	APH	T6_T4A	T6_140	26/11/2012	07/01/2013	18/12/2012	Giancarlo BIASI...	CONS.AGR.LOM...	05/09/2012
	0050476155	TRACTORS	SWB	T7_SWB...	T7_210 AC	08/10/2012	23/10/2012	22/10/2012	Giancarlo BIASI...	CONS.AGR.LOM...	16/02/2012
	0050538109	TRACTORS	SWB	T7_SWB...	T7_210 R...	01/11/2012	14/11/2012	12/11/2012	Paolo CRISCUOLO	CENTRO MACC...	19/07/2012
	0050540826	TRACTORS	SWB	T7_SWB...	T7_200 AC	26/11/2012	03/12/2012	28/11/2012	Giancarlo BIASI...	CONS.AGR.LOM...	31/07/2012
	0050541515	TRACTORS	CCM LWB	T7_T4A	T7_220 PC...	12/11/2012	19/11/2012	16/11/2012	Claudio RAVA	MANILDO S.r.l.	01/08/2012
	0050778229	TRACTORS	Utility	T4000	T4040DS	03/12/2012	07/12/2012	30/11/2012	Claudio RAVA	MANILDO S.r.l.	11/10/2012
	0050779219	TRACTORS	Utility	T4000	T4040DS	10/12/2012	19/12/2012	05/12/2012	Paolo CRISCUOLO	M.C.S.R.L.	17/10/2012
	0050779229	TRACTORS	Utility	T4000	T4050DS	10/12/2012	14/12/2012	05/12/2012	Claudio RAVA	MANILDO S.r.l.	17/10/2012
	0050469902	TRACTORS	Specialty	T4000F	T4060F	28/05/2012	07/06/2012	10/07/2012	Mario DI GIOIA	SETTANNI S. R. L.	19/01/2012
	0050512757	TRACTORS	Specialty	T4000F	T4050F	04/06/2012	13/06/2012	13/06/2012	Giancarlo BIASI...	CONSORZIO A...	30/03/2012
	0050516436	TRACTORS	Specialty	T4000F	T4060F	11/06/2012	13/06/2012	16/06/2012	Giancarlo BIASI...	CONSORZIO A...	13/04/2012

English Italiano Français Español Português

Current Selection: Report: SumYearly | Language: English | KPI1: Wholesales | Month Code: 06 | Year: 2013 | View: Year View | Point of View: Period | Country: Italy

### Unit Details:

Report che contiene tutti gli dettagli e la configurazione dell'ordine selezionato.



## 6.10 - Schedulazione dei caricamenti

Il sistema nuovo creato con QlikView lavora con i dati in-memory per ciò c'è bisogno che i dati vengono aggiornati continuamente.

### 6.10.1 - Schedulazione dei task

Task del QlikView server:

- Task creati per ogni singolo documento
- Possono essere schedulati in tempo o in dipendenza di qualche altro task
- Reload di tutti il documento QlikView
- Reload parziale del documento QlikView
- Distribuzione sul Access Point
- Distribuzione bursting: Da un documento produrre n documenti, ciascun documento filtrato su un elemento prescelto.

Task del nostrosistema:

- ETL1 : caricamento totale. Prende tutti i dati che ci servono dal Data Warehouse. Schedulazione manuale (da batch file);
- ETL2 : caricamento totale. Prende tutti i dati dai file “qvd” prodotti nel ETL1 e crea la nuvola dati. Schedulazione automatica, parte quando il task del ETL1 e andato a buon fine.
- Presentation: caricamento totale. Prende tutti il file “qvd” prodotti nel ETL2 e rilancia tutti i report coi in dati nuovi. Schedulazione automatica, parte quando il task ETL2 e andato a buon fine.

## 6.10.2 - Caricamenti

Il caricamento dei dati nel DataWarehouse :

- Caricamenti Totali 1 al giorno a mezzanotte (con l'ora locale)
- Caricamenti parziali ogni 30min (solo qualche campo nella facttable)

Passi per il caricamento dei dati:

1. Quando il loro Data Warehouse ha finito di caricare lancia un task.
2. Noi catturiamo il task e tramite un batch file lanciamo il task di ETL1
3. Poi parte la catena dei task, ETL2 e Presentation

Batch file:

```
1 set AppProgram="D:\QlikView\Distribution Service\QVDistributionService.exe"
2 set QvwName=Presentation.qvw
3 set ProcessName=Presentantion
4 set QvwPath=D:\Qlik_View\QVW_Presentation
5
6 %AppProgram% -out="%QvwPath%\%ProcessName%_reload.log" -r="%QvwPath%\%QvwName%"
7
8 .\QMSEDX_ComandLine>QMSEDX.exe -task="task name" -password=1234
9
10 IF %ERRORLEVEL% NEQ 0 goto Error
11
12 @echo Finished-%date% %time%
13 PAUSE
14 EXIT
15
16 :Error
17 @echo Finished abend-%date% %time%
18 PAUSE
19 EXIT
```

## Capitolo 7

# Conclusioni

---

In questo ultimo capitolo tratteremo dei risultati ottenuti da:

- Cliente, utilizzando il nuovo commercial tool;
- Iconsulting, per l'esperienza e le competenze acquisite;
- Me, per l'esperienza e le competenze acquisite.

### 7.1 - Miglioramenti percepite dal cliente

Grazie alla nuova soluzione sviluppata, il reparto vendite ha centralizzato tutti i mercati su unico portale, ha migliorato notevolmente la capacità di fare ricerche mirate o generiche sull'efficacia dei loro venditori, o sulla tipologia di mercato.

Tutto questo è stato reso possibile grazie ad un sistema semplice ed efficiente, in grado di fornire risultati in tempi brevissimi, quasi istantanei, processando anche milioni di informazioni.

#### **I vecchi sistemi utilizzati erano:**

- più di uno
- consultabile solo attraverso lo strumenti che è stato sviluppato
- vecchi
- il tempo di risposta è alto
- la visualizzazione dei dati poco leggibile

#### **Nuovo sistema QlikView:**

- unico strumento per tutti i mercati
- consultabile anche da un Ipad
- consultabile anche offline
- la nuova visualizzazione è molto più immediata e leggibile
- la navigazione dei data molto facile
- possibilità di analisi libera

Il nuovo sistema è stato di grande valore per il cliente, che ha potuto apprezzare la capacità di raggiungere gli obiettivi prefissati, nei tempi e modi stabiliti.

In conclusione, da parte del cliente, c'è stato grande apprezzamento per l'impegno e la serietà con cui si è portato a termine il progetto.

**La reportistica automatizzata permette:**

- un'allocazione del tempo lavorativo più proficua
- una maggiore qualità e solidità delle decisioni di business sostenute
- la parziale eliminazione delle procedure cartacee
- la centralizzazione delle informazioni

## **7.2 - Esperienze e competenze acquisite**

L'esperienza di questa tesi ed in particolare il progetto su cui ho lavorato, con l'appoggio dei consulenti di Iconsulting, mi hanno portato ad effettuare diverse considerazioni sia dal punto di vista "tecnico" che da quello "organizzativo", in particolare:

- In primo luogo la logica informatica e la logica gestionale, devono necessariamente operare in una collaborazione completa, in quanto le specifiche sono dettate dai bisogni dell'azienda che rientrano in un'ottica del tutto diversa rispetto a quella tecnologica, deve essere il "consulente" a individuare i limiti e le opportunità, progettando ed implementando un'applicazione in grado di rispecchiare le reali necessità e di integrarle con soluzioni innovative.
- L'organizzazione nel suo complesso deve essere pronta al cambiamento in quanto lo strumento informatico porta con sé molti vantaggi, ma anche "switching costs" che per persone che operano da anni con una certa metodologia possono risultare elevati. Ma il trend moderno richiede un cambiamento per migliorare le performance interne, massimizzando l'allocazione delle risorse e la loro valorizzazione, venendo ad essere l'apporto del singolo o della singola unità, competenza dell'intera azienda (ovviamente se valida).

- I prodotti di B.I. sono utili per tutte le tipologie di imprese, ma i vantaggi riscontrati sono più evidenti nelle aziende medio-grandi caratterizzate da complessità organizzativa e ben strutturate dal punto di vista I.T.. Le piccole possono trarre beneficio solo se gli investimenti in I.T. hanno valenza strategica, visti quindi come un'opportunità e non come un costo.

### **Iconsulting:**

Dal punto di vista, si è portato avanti un discorso di crescita di competenze nell'ambito della programmazione di QlikView. Infatti dopo una serie di incontri e corsi fatti con esperti del software, si è riusciti a delineare e a costruire un documento ben formato, nel quale sono state raccolte tutte una serie di tecniche e best practices di progettazione.

Prima, le competenze interne condivise erano limitate alla sola logica di costruzione base di un progetto, senza applicare approcci particolari di sviluppo dello stesso, generando un codice poco mantenibile.

Oggi l'azienda può contare su di una documentazione ben precisa, che descrive passo dopo passo le procedure da seguire per impostare e sviluppare un progetto in maniera efficiente.

Le seguenti procedure, descritte in parte nel capitolo 5, se seguite correttamente permettono di raggiungere in modo rapido ed efficace la costruzione di un modello, e allo stesso tempo facilitano le azioni di manutenzione successive.

Concludendo possiamo affermare che il progetto ha arricchito da una parte l'azienda, in termini di competenze ed esperienza, dall'altra mi ha permesso di entrare a conoscenza di un mondo nuovo, dal quale spero di aver tratto i migliori benefici, impegnandomi ad apprendere tutto quello che mi è stato proposto.

## Bibliografia

Richard Adhikari. "Migrating legacy data". Software magazine, Vol. 16, No. 1, Gennaio 1996.

"The Role of the Multidimensional Database in a Data Warehousing Solution". Arbor Software, 1995.

Joyce Bischoff e Ted Alexander. "Data Warehouse : practical advice from the experts". Prentice-Hall, 1997.

Scott Brown. "Try Slice and dice". Computing Canada, Vol. 21, No. 22, Ottobre 1995.

Luca Cabibbo e Riccardo Torlone. "A Logical Approach to Multidimensional Databases". Università di Roma Tre, Rapporto Interno.

E.F. Codd. "Providing OLAP (On-Line Analytical Processing) to User-Analyst: An IT Mandate". Arbor Software, 1993.

"Oracle Express Server 6.2 : Administration Guide". Oracle Corporation, 1998.

Kevin Fogarty. "Data mining". Network world, Vol. 11, No. 23, Giugno 1994.

Jim Gray e Andreas Reuter. "Transaction processing : concepts and techniques". Morgan Kaufmann, 1993.

William H. Inmon. "Building the Data Warehouse". John Wiley & Sons, 1992.

William H. Inmon. "Building the data bridge: the ten critical success factors of building a data warehouse". Database programming & Design, 1992.

William H. Inmon. "EIS and the data warehouse : a simple approach to building an effective foundation for EIS." Database programming & Design, 1992.

John Kador. "One to one". Midrange Systems, Vol. 8, No. 20, Ottobre 1995.

"An introduction to Multidimensional Database Technology". Kenan Technologies, 1995.

Ralph Kimball. "The Data Warehouse Toolkit". John Wiley & Sons, 1996.

Ralph Kimball, Laura Reeves, Margy Ross e Warren Thornthwaite. "The Data Warehouse Lifecycle Toolkit : expert methods for designing, developing, and deploying data warehouse". John Wiley & Sons, 1998.

David Menninger. "Oracle OLAP Products: Adding Value to the Warehouse". Oracle Corporation.

“Relational OLAP: An Enterprise-Wide Data Delivery Architecture”. Microstrategy Incorporated, 1994.

“The Case for Relational OLAP”. Microstrategy Incorporated, 1995.

“Data Warehousing, Data Modelling & Design”. Microstrategy Education, 1996.

“Fundamentals of DSS Agent & Architect”. Microstrategy Education, 1998.

E. Mollona. *Analisi dinamica dei sistemi aziendali*. Egea, 2000.

M. Golfarelli. New trends in business intelligence. *In proceedings ACM 11th International Conference on Business Intelligence Systems, Opatija, Croatia.*, 2005.

R. Stadler J. Verhees A. Zanasi P. Cabena, P. Hadjinian. *Discovering Data Mining - From Concept to Implementation*. Prentice Hall PTR, Upper Saddle River, New Jersey 07458, 1997.

S. Brobst V. Poe, P. Klaner. *Building a Data Warehouse for Decision Support*. Prentice-Hall, 1998.

QlikView. Disponibile al sito: <http://www.qlikview.com/>

QlikViewCommunity. Disponibile al sito: <http://community.qlikview.com/welcome>

QlikViewSlides: disponibili all'interno della sezione Community di QlikView:

- QlikView Optimization Best Practices,
- QlikView Reference Manual,
- Developing on QlikView,
- QlikView Server Reference Manual

Wikipedia, 2013. Disponibile al sito: <http://it.wikipedia.org/>