ROMA TRE UNIVERSITÀ DEGLI STUDI
Corso di Laurea in Matematica
Dipartimento di Matematica e Fisica

Sistemi per l'elaborazione delle informazioni

6. Data warehouse

Dispense del corso IN530 a.a. 2019/2020

prof. Marco Liverani

Sistemi operazionali e informazionali

- Nell'ambito di un sistema informativo si distingue tra sottosistemi o componenti del sistema informativo di due tipologie macroscopiche distinte:
  1. sistemi operazionali
  2. sistemi informazionali
- Sistemi **operazionali**:
  - sono i sistemi di **supporto operativo** allo svolgimento delle attività istituzionali dell'azienda o dell'ente
- Sistemi **informazionali**:
  - sono sistemi di **supporto alle decisioni** che supportano l'organizzazione nelle proprie scelte strategiche e nello studio e nella comprensione dell'andamento di specifici indicatori che sono ritenuti significativi in uno specifico ambito di business (es.: le vendite per un'azienda commerciale, il numero di iscritti e di laureati per un'Università, ecc.)
- Entrambe le tipologie di sistema sono supportate da archivi informatici (database) progettati ed utilizzati in modo differente, al fine di supportare al meglio gli obiettivi del sistema

Database operazionali

- Un database **operazionale** è un archivio che garantisce la persistenza ad un'applicazione che deve operare sui dati dell'archivio per realizzare funzioni di business
  - Esempio 1: l'archivio dei libri di una biblioteca viene utilizzato per consentire la ricerca efficiente dei volumi da parte degli utenti, viene frequentemente aggiornato tracciando i nuovi libri acquistati dalla biblioteca, i volumi in prestito, gli utenti registrati, i testi persi o dismessi;
  - Esempio 2: l'archivio del magazzino dei prodotti di un negozio viene utilizzato per garantire gli approvvigionamenti di prodotti venduti, gestire le scadenze dei prodotti alimentari deperibili, aggiornare i prezzi dei prodotti e automatizzare il calcolo del conto per ogni cliente in fase di pagamento alle casse, ecc.
- In un database operazionale le quattro operazioni di inserimento, selezione, aggiornamento e cancellazione **sono ugualmente frequenti** e riguardano pochi dati per volta: i dati in archivio sono tutti e soli quelli necessari all'operatività del programma
- In un database operazionale **i dati hanno una profondità temporale limitata**: vengono conservati solo i dati effettivamente utili a supportare le operazioni svolte dal programma che utilizza l'archivio
- I DBMS sono progettati per svolgere elaborazioni di tipo **OLTP**: *On Line Transaction Processing*
  - il DBMS supporta strutture normalizzate per la memorizzazione efficiente delle informazioni, indici per la ricerca efficiente dei dati, transazioni per l'elaborazione di sequenze di operazioni in modalità "reversibile"

Database informazionali

- Un **database informazionale** (in inglese: ***data warehouse***, magazzino dei dati) è un archivio la cui funzione d'uso è la costruzione di una base informativa che **cresce nel tempo**, utile per fornire informazioni utili alla comprensione di un fenomeno, allo studio di un andamento
  - Esempio 1: l'**archivio delle vendite** dei prodotti di una catena di supermercati, storicizzate nel tempo, consente di studiare la stagionalità nella vendita di determinati prodotti, prevedere i consumi, gli acquisti dei clienti e provvedere quindi al corretto approvvigionamento del magazzino
  - Esempio 2: l'archivio dei **dati storici degli studenti iscritti** ai diversi corsi di laurea di un Ateneo, tenendo conto anche del titolo di studio posseduto al momento dell'iscrizione all'Università, al fine di studiare le tendenze in ambito formativo e dimensionare correttamente la futura offerta formativa
- In un database informazionale l'operazione più frequente è la **selezione**; le operazioni di aggiornamento e di cancellazione sono rarissime (vengono effettuate solo per migliorare la qualità dei dati presenti nel data warehouse); le operazioni di inserimento per il caricamento dei dati in archivio avviene periodicamente e riguarda grosse quantità di dati
- La **profondità storica** dei dati presenti nel data warehouse è fondamentale per poter utilizzare i dati come oggetto di analisi e studiare l'evoluzione storica delle informazioni
- I **Data Warehouse** (DWH) sono progettati per eseguire operazioni di tipo **OLAP**: *On Line Analytical Processing*
  - L'archivio è organizzato su strutture multidimensionali per favorire l'analisi su dimensioni differenti (es.: la professione dei clienti di un supermercato, l'età, la stagionalità dei prodotti, ecc.)

Integrazione di dati provenienti da fonti eterogenee

- Un data warehouse viene costruito aggregando in un unico database dati provenienti da **fonti differenti**, che trattano dati eterogenei e con rappresentazioni spesso incompatibili
- Nel data warehouse tutti questi dati devono trovare invece una **codifica omogenea**: in fase di caricamento i dati vengono ricodificati per sposare la codifica e la semantica del data warehouse

  Esempio:
  - in un database di un supermercato la tabella dei clienti che hanno sottoscritto una "tessera fedeltà" può contenere il codice fiscale del cliente
  - nel data warehouse l'analisi può essere condotta sulla base dell'età dei clienti, del genere e del luogo di nascita: dati che possono essere desunti dal CF in fase di caricamento

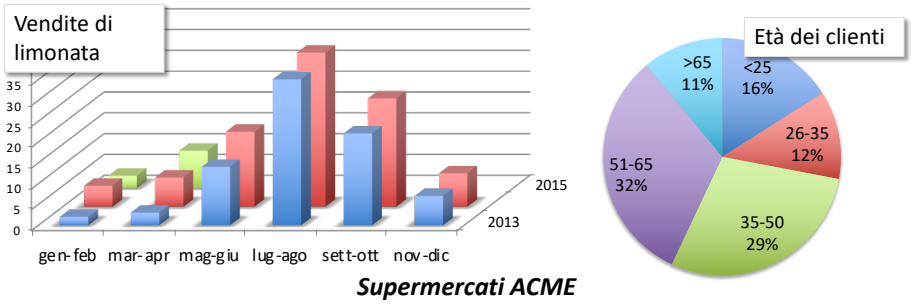
- Nel data warehouse i dati sono rappresentati in modo da supportare specifici **temi di analisi** focalizzati di volta in volta su **soggetti** diversi

  Esempio:
  - attitudini di acquisto dei clienti → il soggetto è il cliente
  - efficienza dei diversi punti vendita → il soggetto è il punto vendita (il negozio)
  - stagionalità del prodotto → il soggetto è il prodotto

  le diverse linee di analisi sono costruite intorno all'archivio degli eventi "**vendita di un prodotto**"

Processo di aggregazione e di analisi

- La costruzione di un data warehouse è centrata sul processo di aggregazione e di integrazione di informazioni provenienti da tante basi dati differenti
- Lo scopo del data warehouse è quello di supportare attività di analisi rese possibili proprio dall'aggregazione in un unico archivio di dati provenienti da fonti differenti, che offrano una vista unificata su tutti gli aspetti rilevanti dell'attività di business
  - Esempio: non l'archivio dei clienti, dei prodotti presenti in magazzino e degli incassi registrati nei singoli punti vendita, ma il data warehouse delle vendite i cui elementi sono "record" che aggregano informazioni relative al prodotto venduto, al cliente che lo ha acquistato, al prezzo che è stato pagato per approvvigionarsi del prodotto, al ricavo che è stato ottenuto dalla vendita, al negozio che ha effettuato la vendita e la data in cui la vendita è stata effettuata



Supermercati ACME

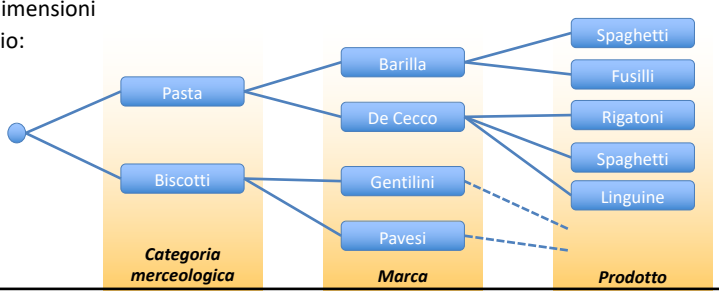
Struttura logica di un data warehouse

- Nella progettazione di un *data warehouse* è essenziale identificare i seguenti elementi nel contesto informativo che si vuole rappresentare ed analizzare:
  - fatto
  - misure
  - dimensioni
- I **fatti** sono l'insieme dei dati da analizzare; tipicamente sono eventi che si collocano nel tempo e sono caratterizzati da diverse informazioni associate al singolo fatto
  - Esempio: le vendite della ACME; ogni evento di vendita è un fatto caratterizzato da numerose informazioni, quali il prodotto venduto, la quantità venduta, il valore della merce, il cliente, la collocazione geografica del negozio, la data e l'ora della vendita, la categoria merceologica del prodotto, ecc.
- Le **misure** sono dati quantitativi numerici che rappresentano gli aspetti da misurare in relazione all'analisi dei fatti
  - Esempio: la quantità di un determinato prodotto per ciascuna vendita, il ricavo per ciascuna vendita, ecc.
- le **dimensioni** sono informazioni che caratterizzano il fatto, a valori discreti, rappresentano le linee di analisi dei fatti (o meglio: delle misure associate ai fatti)
  - Esempio: i prodotti di un supermercato, il tempo, i clienti, ecc.

Struttura logica di un data warehouse

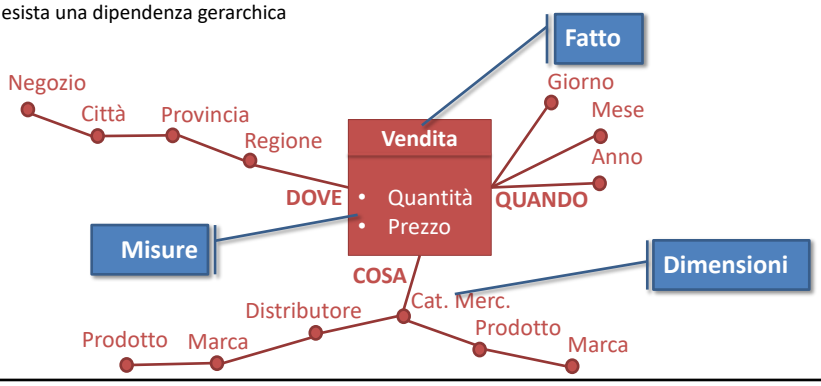
- Le dimensioni di analisi dei fatti di un data warehouse possono essere numerose
- È possibile stabilire una **dipendenza gerarchica** tra alcune delle dimensioni di analisi, in modo da aggregare o espandere le informazioni relative all'analisi di una determinata dimensione

  Esempio: nel data warehouse delle vendite della ACME alcune dimensioni di analisi possono essere aggregate formando delle gerarchie:
  - Gerarchia della dimensione del prodotto venduto: prodotto → marca → categoria merceologica
  - Gerarchia del tempo in cui sono avvenute le vendite: giorno → mese → anno
  - Gerarchia dei clienti: nome → età → genere → professione,
    o anche: nome → titolo di studio → età → genere
- Definendo una gerarchia nella dimensione si crea una struttura ad albero nei valori discreti delle dimensioni
- Esempio:



Dimensional Fact Model

- Il **Dimensional Fact Model** (*Golfarelli e Rizzi, Università di Bologna, 1998*) è un formalismo grafico per la modellazione logica di un data warehouse
- Nel DFM si focalizza l'attenzione proprio sui rapporti tra fatti, misure e dimensioni, mettendo in evidenza eventuali gerarchie nelle dimensioni
  - **Fatto**: rappresentato con un rettangolo identificato dal nome del fatto stesso;
  - **Misure**: riportate all'interno del rettangolo che identifica il fatto a cui si riferiscono;
  - **Dimensioni**: sono collegate al fatto, come i vertici di un grafo, o ad altre dimensioni con le quali esista una dipendenza gerarchica

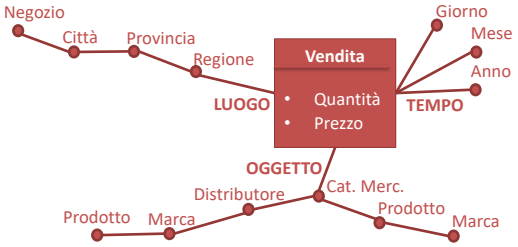


Operazioni di analisi su un DWH

- Su un data warehouse organizzato sulla base di fatti, misure e dimensioni l'operazione tipica è quella di estrazione dei dati (misure) riportati nei fatti, aggregandoli sulla base di una o più dimensioni
- L'obiettivo è il **calcolo di una funzione di gruppo** sui sotto-insiemi ottenuti eseguendo un partizionamento dei dati presenti nel data warehouse in base all'aggregazione per una o più dimensioni (es.: partizionamento su base geografica (regioni, province, comuni) e/o su base temporale (aggregazione per anno, mese, giorno)
- Il calcolo che può essere fatto sulle misure (che per questo sono di tipo numerico) sono le tipiche funzioni di gruppo già esaminate nel linguaggio SQL:
  - conteggio dei record (fatti), per ricavare la cardinalità dei sottoinsiemi della partizione
  - somma o media aritmetica sui valori di una specifica misura nei diversi sottoinsiemi della partizione
  - minimo o massimo valore sui valori di una misura nei sottoinsiemi della partizione
- Esempio: le vendite dei supermercati ACME aggregate per regione e provincia (somma totale e somme parziali) e per anno
- Le funzioni di calcolo non si riducono solo alle funzioni di gruppo offerte da SQL, ma possono essere anche **funzioni statistiche** più sofisticate implementate dal software utilizzato per effettuare l'analisi dei dati presenti nel data warehouse (*business intelligence*)

Operazioni di analisi su un DWH

- Prendendo ad esempio il *Dimensional Fact Model* delle vendite dei supermercati ACME, possiamo calcolare facilmente un'aggregazione per regioni e province e la distribuzione nel tempo delle vendite



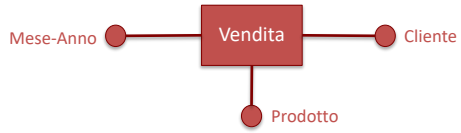
Regione	Provincia	2012	2013	2014	2015
Lazio	*Roma*	6.400	6.850	7.630	9.125
	Viterbo	1.850	1.930	2.360	2.270
	Latina	2.450	2.820	3.515	3.920
Campania	*Napoli*	5.850	6.230	6.540	6.520
	Salerno	3.810	4.200	4.075	4.350

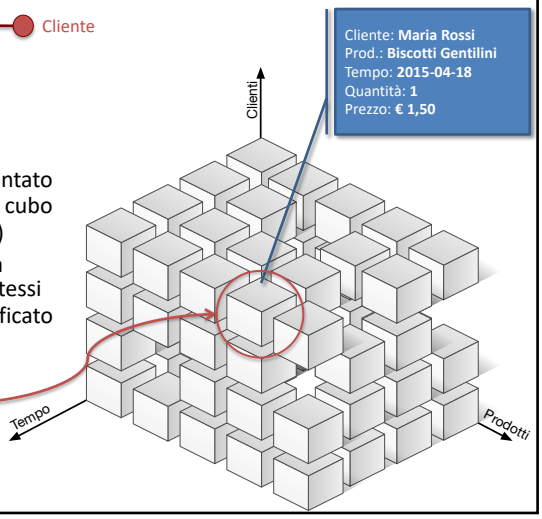
Modello multidimensionale e ipercubi

- Il *Dimensional Fact Model* con cui viene definita la struttura logica di un data warehouse, ci suggerisce la possibilità di rappresentare il data warehouse come uno spazio multidimensionale, in cui i fatti siano dei punti posizionati nello spazio sulla base di coordinate definite dal valore del fatto per ciascuna delle dimensioni del modello
- Ricordiamo che le dimensioni sono **insiemi discreti di cardinalità finita** (es.: gli anni della profondità storica del data warehouse, i prodotti venduti dai Supermercati ACME, i clienti, i fornitori, ecc.)
- Il data warehouse può quindi essere rappresentato come un **cubo** (quando le dimensioni indipendenti di analisi sono tre) o un **ipercubo** (quando le dimensioni indipendenti sono in numero maggiore di tre) i cui elementi sono i fatti del data warehouse
- Su ciascuna delle dimensioni dell'ipercubo sono distribuiti (con un ordine arbitrario) i valori discreti della dimensione stessa

Modello multidimensionale e ipercubi

- Consideriamo un modello a tre dimensioni: un database delle vendite i cui fatti sono le operazioni di vendita e le cui dimensioni siano il tempo, il prodotto venduto, il cliente





- Il data warehouse può essere rappresentato come un **ipercubo *n*-dimensionale** (un cubo a tre dimensioni nell'esempio in figura)
- Ogni punto del cubo è un "fatto" o una aggregazione di fatti identificati dagli stessi valori delle dimensioni; il fatto è identificato da un valore del dominio su ciascuna dimensione
- Nell'esempio ogni punto del cubo rappresenta l'acquisto di un prodotto effettuato da un cliente in un certo giorno

Modello MOLAP per la rappresentazione del DWH

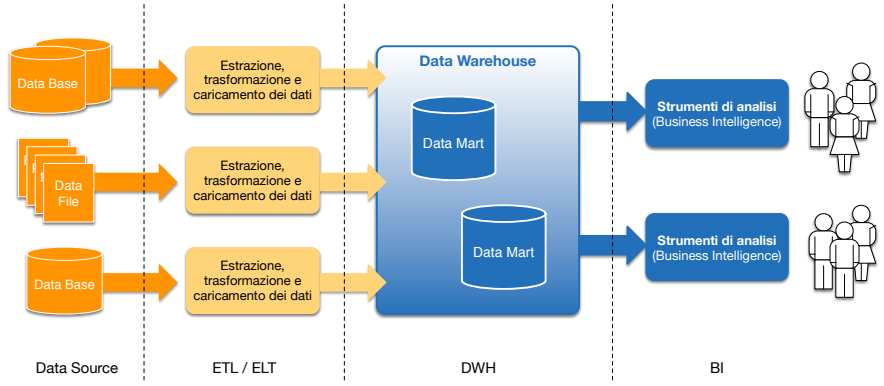
- Alcuni sistemi rappresentano il data warehouse proprio con un cubo multidimensionale, utilizzando apposite strutture dati: tali sistemi si chiamano **MOLAP** (*multidimensional on-line analytical processing*)
- La rappresentazione di un ipercubo OLAP è assai onerosa in termini di risorse di memoria
- La struttura di ipercubo spesso è piena di "buchi", se i fatti sono disposti nel cubo in modo poco denso: questo rende poco efficiente la rappresentazione diretta mediante un cubo
- D'altra parte la rappresentazione con un ipercubo OLAP rende immediate e molto efficienti in termini di tempo le operazioni di analisi:
  - **slice**: si estrae una "fetta" dell'ipercubo, analizzando l'archivio fissando il valore di una o più dimensioni
  - **dice**: si estrae un sotto-cubo, analizzando l'archivio fissando un sottoinsieme di valori per ogni dimensione
  - **roll-up**: si aggregano i dati delle *misure* (somma, media, conteggio, altre funzioni statistiche) sulla base di un valore di una o più dimensioni
  - **drill-down**: si spacchettano dati aggregati sulla base di una o più dimensioni (o limitatamente ad alcuni livelli della gerarchia delle dimensioni) per ottenere maggior dettaglio

Modello ROLAP per la rappresentazione del DWH

- Lo stesso *data warehouse* può essere rappresentato utilizzando un DBMS relazionale (RDBMS) costruendo uno **schema a stella** (***star schema***), sulla falsa riga del *dimensional fact model*:
  - Al centro dello star schema c'è la tabella dei fatti
  - La tabella dei fatti è collegata con le tabelle delle dimensioni attraverso opportune chiavi esterne (*foreign key*)
- In questo modo si utilizza uno strumento robusto e ben consolidato, potendo contare anche sul linguaggio SQL per l'esecuzione delle operazioni di aggregazione (*slice* e *roll-up*) e di scomposizione (*dice* e *drill-down*) dei dati del data warehouse
- La rappresentazione **ROLAP** (*relational on-line analytical processing*) è più efficiente in termini di risorse di memoria, ma può risultare meno efficiente nell'esecuzione delle operazioni di analisi, a meno di non utilizzare accuratamente degli indici per velocizzare le operazioni di selezione e aggregazione

Architettura di un sistema data warehouse

- L'architettura del sistema di data warehouse è definita sulla base delle componenti funzionali che si vuole integrare per costituire il sistema informazionale dell'azienda:
  - sorgenti informative
  - archivio centralizzato (il data warehouse)
  - strumenti di analisi dei dati facilmente accessibili dagli utenti
- Possiamo quindi schematizzare come in figura l'architettura di un sistema DWH:



Architettura di un sistema data warehouse

- I **sistemi sorgente per l'alimentazione del data warehouse** sono spesso i database dei sistemi operazionali dell'azienda (anagrafica dei fornitori, anagrafica dei clienti, database degli acquisti e delle vendite, ecc.)
- I sistemi sorgente sono per loro natura **eterogenei**:
  - possono essere costituiti da **sistemi di tipo diverso** (diversi tipi di DBMS, file in formati proprietari o XML, ecc.)
  - **differente rappresentazione delle informazioni** (diversa codifica, diversi tipi di dato per attributi analoghi su sorgenti diverse, ecc.)
  - sono ospitati da **sistemi operativi e da server differenti**, collegati in rete con il sistema Data Warehouse aziendale
- Le procedure **ETL** (*extract transform and load*) consentono di eseguire periodicamente delle query di estrazione dei dati dai sistemi sorgente e li trasformano codificandoli in una modalità omogenea (come formato e come contenuto) per poi caricarli sul data warehouse
- Bisogna definire:
  - la modalità operativa della procedura ETL (*agent-based*, *agentless*, ecc.)
  - la periodicità con cui avviene l'estrazione da ciascuna sorgente (tutti i giorni, tutti i mesi, ecc.)
  - la logica di selezione dei dati da estrarre dalla sorgente (è necessario implementare una logica per non caricare ogni volta tutti i dati, ma solo quelli nuovi o modificati sulla sorgente)
  - la codifica e la modalità di rappresentazione di dati eterogenei in un modello unico (ipercubo OLAP o *star schema*)

Architettura di un sistema data warehouse

- Il data warehouse vero e proprio è costituito da uno strumento RDBMS eventualmente dotato di opportuni moduli di gestione di ipercubi OLAP
- Il DWH è costituito da uno o più ***data mart***, ossia porzioni autonome del data warehouse, inerenti una determinata tematica di analisi
  - La suddivisione del data warehouse in più *data mart* è eseguita sulla base di considerazioni di efficienza nelle operazioni di analisi e di riservatezza delle informazioni che vengono rese disponibili agli utenti
- Il DWH ha bisogno di un'attività di **manutenzione continua** finalizzata a:
  - verificare la corretta esecuzione delle operazioni di caricamento dei dati
  - verifica dei dati caricati sul data warehouse, per correggere eventuali incongruenze dovute alla errata attuazione delle regole di trasformazione dei dati in fase di caricamento (*data quality*)
  - *tuning* per migliorare l'efficienza del sistema, creando o ricreando indici sulle tabelle, gestendo l'enorme mole di dati del data warehouse, attraverso la partizione fisica dei data file su cui si appoggia il DBMS
  - progettazione e realizzazione di nuove procedure di analisi e presentazione dei dati (report, cruscotti, *dashboard*)

Architettura di un sistema data warehouse

- La componente di presentazione dei dati analitici e di sintesi, è spesso una componente software che implementa complesse funzionalità di analisi e di rappresentazione grafica delle misure presenti sul data warehouse
- Si parla in questi casi di sistema di ***Business Intelligence*** (**BI**), uno strumento software su cui è possibile definire:
  - report corrispondenti a specifiche viste sul data warehouse
  - cruscotti e *dashboard* di sintesi, anche in forma grafica (istogrammi, grafi, ...)
  - strumenti per eseguire dinamicamente le operazioni di navigazione dei dati presenti nel modello multidimensionale del data warehouse (*drill-down*, *roll-up*, *slice*, *dice*, *pivoting*, ecc.)
- I software di BI sono programmi che si agganciano ad un modello multidimensionale del data warehouse ed offrono funzioni di rielaborazione e presentazione dei dati stessi; frequentemente i software di BI sono applicazioni *web based*
- Con i moduli di "modeling/design" di un prodotto BI, l'utente può costruire i propri report e cruscotti, senza dover necessariamente conoscere la struttura fisica del database sottostante, né il linguaggio SQL

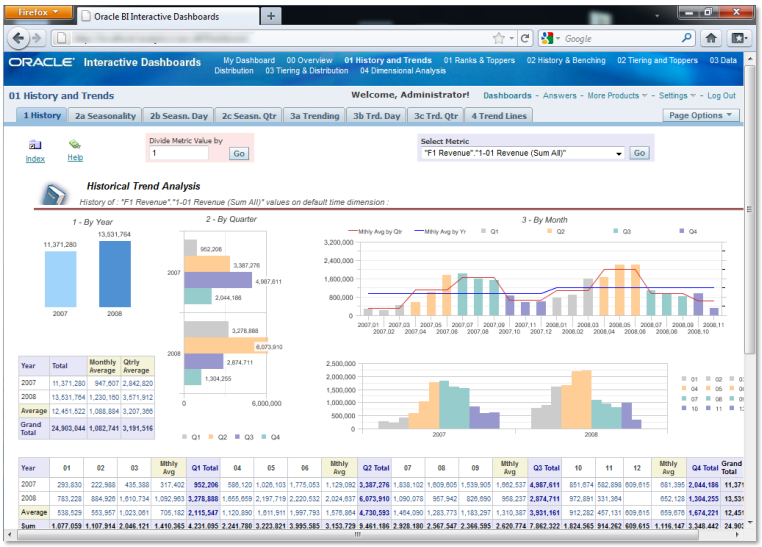
Bibliografia essenziale

① Golfarelli, Rizzi, *Data Warehouse – teoria e pratica della Progettazione*, McGraw-Hill, 2006

② Kimball, Ross, *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*, terza edizione, Wiley, 2013

③ Golfarelli, Maio, Rizzi, *The Dimensional Fact Model: a Conceptual Model for Data Warehouses*, International Journal of Cooperative Information Systems, vol. 7, n. 2&3, 1998

④ Golfarelli, Rizzi, *Progettazione concettuale di Data Warehouse da schemi logici relazionali*, Proceedings Sesto Convegno Nazionale su Sistemi Evoluti Per Basi Di Dati, Ancona, 1998



Oracle Business Intelligence

Una tipica schermata del prodotto Oracle Business Intelligence Enterprise Edition, che consente di costruire cruscotti di analisi dati, accessibili con un web browser, per eseguire attività di analisi su un data warehouse