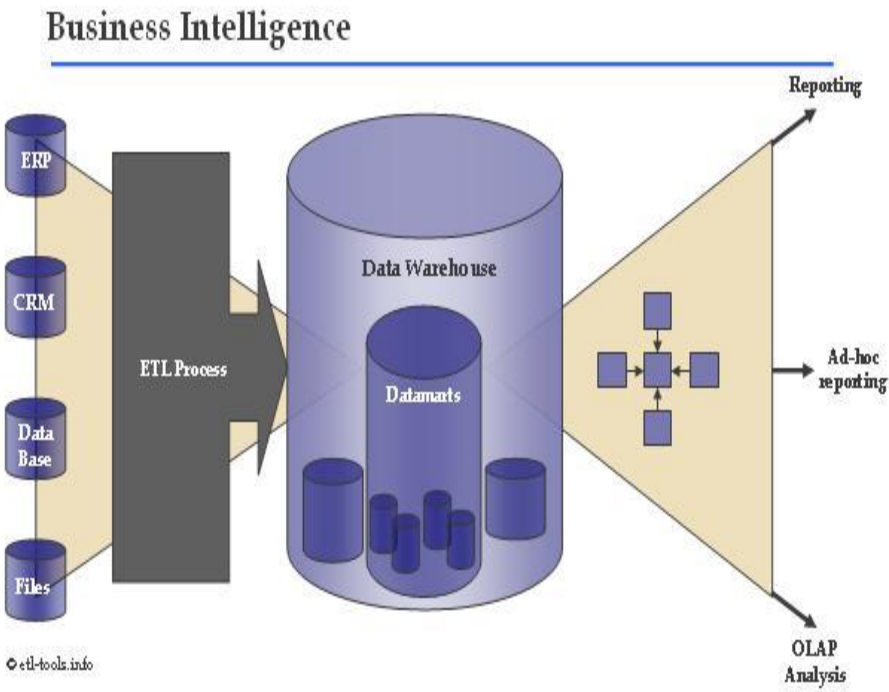
Data Warehousing, Business Intelligence, & Analytics

Business Intelligence

- As companies become larger and more multinational, it becomes harder and harder for them to understand who their customers are, how to best serve them, and how to *maximize* their own profits
- As a result, over the years, executives have relied on their "gut feeling“ - thousands of crucial business decisions have been made based on nothing more than the hunches of CEOs
- That's no way to run a business - unfortunately, in the past there was no alternative

Business Intelligence (con't)

- Until a new way of thinking emerged...
- **When it comes to business decisions, *nothing* should be left to chance**
- **Data on every single aspect of the business should be meticulously collected and then rigorously analyzed to make sure each action is optimal**



Business Intelligence (con’t)

- This approach relies on large “data warehouses” and complex software that uses sophisticated algorithms to pore through endless amounts of data
- Business technologists have many names for this revolutionary technology; "business intelligence” (BI), "data analytics," and "data mining" are among the most common
- **But no matter what you call it, there's no doubt this technology is *the* future of business**

Business Intelligence (con’t)

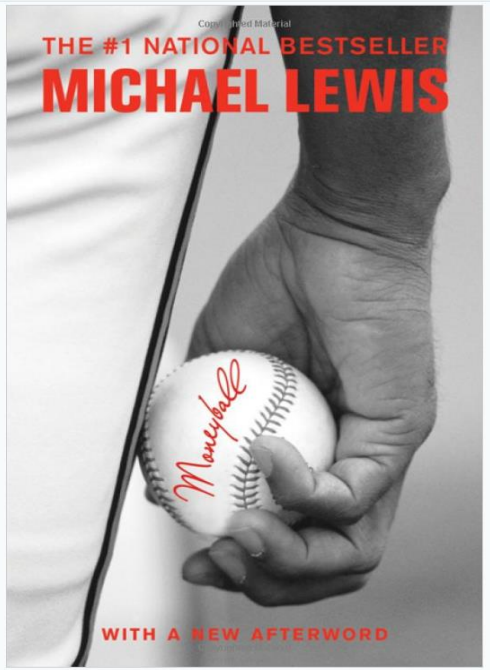
- *The Economist* says it's "a golden vein“, and business experts now call it "the new science of winning"
- FedEx, Capital One, and Amazon.com can't function without it
- It's been adopted by nearly every Fortune 500
- Even professional sports franchises like the Boston Red Sox, Oakland A's, and New England Patriots are being forced to use this technology

Business Intelligence (con't)

- "It's not just the Ubers of the world that need access to a designated team of 'quants'
  - An analytics department is the hottest new function of the 21st century for companies in almost every industry
- To grow successfully, you need precise data analysis driving all key decisions – inventory, routing, pricing, and staffing; otherwise, you're just winging it – and you'll never make the most of your marketplace"
  - FORTUNE magazine, July 1 2015

Moneyball

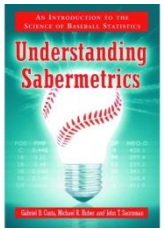
- Using IT, data warehousing, and statistics for competitive advantage
- “Information is baseball’s currency” – Epstein, Boston
- On-base % is a far better tool to evaluate a hitter than batting average
- Typical Apps:
  - Red Sox – “Carmine”
  - Indians – “DiamondView”



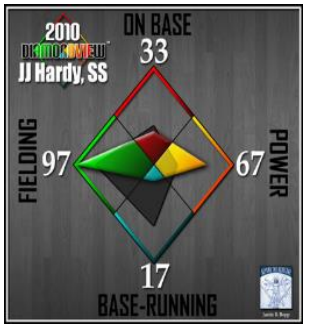
2003

Moneyball (con't)

- *Moneyball*, which opened 9/2011, is based on a 2003 book of the same name by Michael Lewis
- It describes how the Oakland A's general manager Billy Beane eschewed traditional metrics such as RBIs and home runs when evaluating and selecting players
- Instead, he focused on lesser-known and rarely used metrics such as walks plus hits per inning pitched (WHIP), on-base average, and value over replacement player (VORP) when deciding how valuable a player would be to the team
- His approach resulted in the creation of an Oakland baseball team that made it to several playoff rounds in the early to mid-2000s even though it had the lowest payroll in Major League Baseball



Sabermetrics



Sabermetrics is the specialized analysis of baseball through objective, empirical evidence, specifically baseball statistics that measure in-game activity
The term is derived from the acronym SABR, which stands for the Society for American Baseball Research; **Examples of sabermetric measurements:**

- Base runs (BsR)
- Batting average on balls in play (BABIP)
- Defense independent pitching statistics (DIPS)
  - Defense-Independent ERA
  - Defense-Independent Component ERA
  - Fielding independent pitching (FIP)
  - Expected FIP (xFIP)
- Equivalent average (EQA)
- Fantasy batter value (FBV)
- Late-inning pressure situations (LIPS)
- On-base plus slugging (OPS)
- PECOTA (Player empirical comparison and optimization test algorithm)
- Peripheral ERA (PERA)
- Pythagorean expectation
- NERD
- Range factor
- Runs created
- Secondary average
- Similarity score
- Speed Score
- Super linear weights
- Total player rating, or Batter-Fielder Wins (TPR, BFW); Total Pitcher Index, or Pitcher Wins (TPI, PW)
- Ultimate zone rating (UZR)
- Value over replacement player (VORP)
- Win shares
- wOBA
- Wins above replacement (WAR)

Analytics & Football

- Football teams are also using big data technology to guide their decision-making on and off the field
- For instance, by tracking personnel formations, run-pass distributions by field segment and repeated and successful play tendencies, teams can determine which areas of the field are leading to greater success
- They can then call plays that target those areas of the field

Analytics & Basketball

- Real shifts in strategic philosophy have been rare in basketball—the metrics employed a half-century ago by John Wooden and Red Auerbach to evaluate talent remain prevalent today
- But for nearly a decade now, many N.B.A. teams have taken clear steps to integrate advanced statistical analysis into their scouting processes and in-game strategy, a cultural shift that happened in other sports, like baseball and football, years ago
  - Twenty-two of the thirty N.B.A. teams have some kind of analytics department in their front offices now

Analytics & Golf

- A number of current PGA Tour pros have added a data analytics expert to their team (swing coach, short game guru, trainer, massage therapist, sport psychologist, and caddie)
- The PGA's Shotlink System (introduced in 2001) collects and circulates scoring and statistical data and scoring on every shot by every player

It started with Capital One...

- Back in the 1980s, consultants Richard Fairbank and Nigel Morris realized that by analyzing data, credit card companies -- like a tiny Virginia bank called Signet -- could systematically target the most lucrative customers, while leaving their competition to fight over the rest
-
- Their approach was so successful, Signet ended up spinning off its credit card division as a separate company, which became Capital One
-
- Today, Capital One runs approximately 300 data tests *per day* on everything from CD interest rates to rollover incentives to minimum balances -- and it credits data analysis with increasing the retention rate of its savings business by a whopping 87% while simultaneously slashing the cost of acquiring new customers by 83%
-
- BI has allowed Capital One to increase the value of its stock 1,000% over its first ten years as a public company -- outpacing the S&P 500 by a power of 10

Business Intelligence (con't)

- A recent Gartner survey of 1,400 chief information officers suggests that "business intelligence is the number one technology priority for IT organizations"
- Not to mention the fact that dozens of professional sports franchises are now using this technology both on and off the field to boost their winning percentages -- and their bottom lines
- Analytics are also being used in virtually every branch of the United States government and military to improve efficiency and drive profitability

Business Intelligence (con't)

- Companies are not short on data
- The average large business stores more than 200 terabytes (10^{12}) from their daily transactions
- This tells them who has bought what, where and when at what price
- But today business also need to know why, or why not
- How do they do that ?

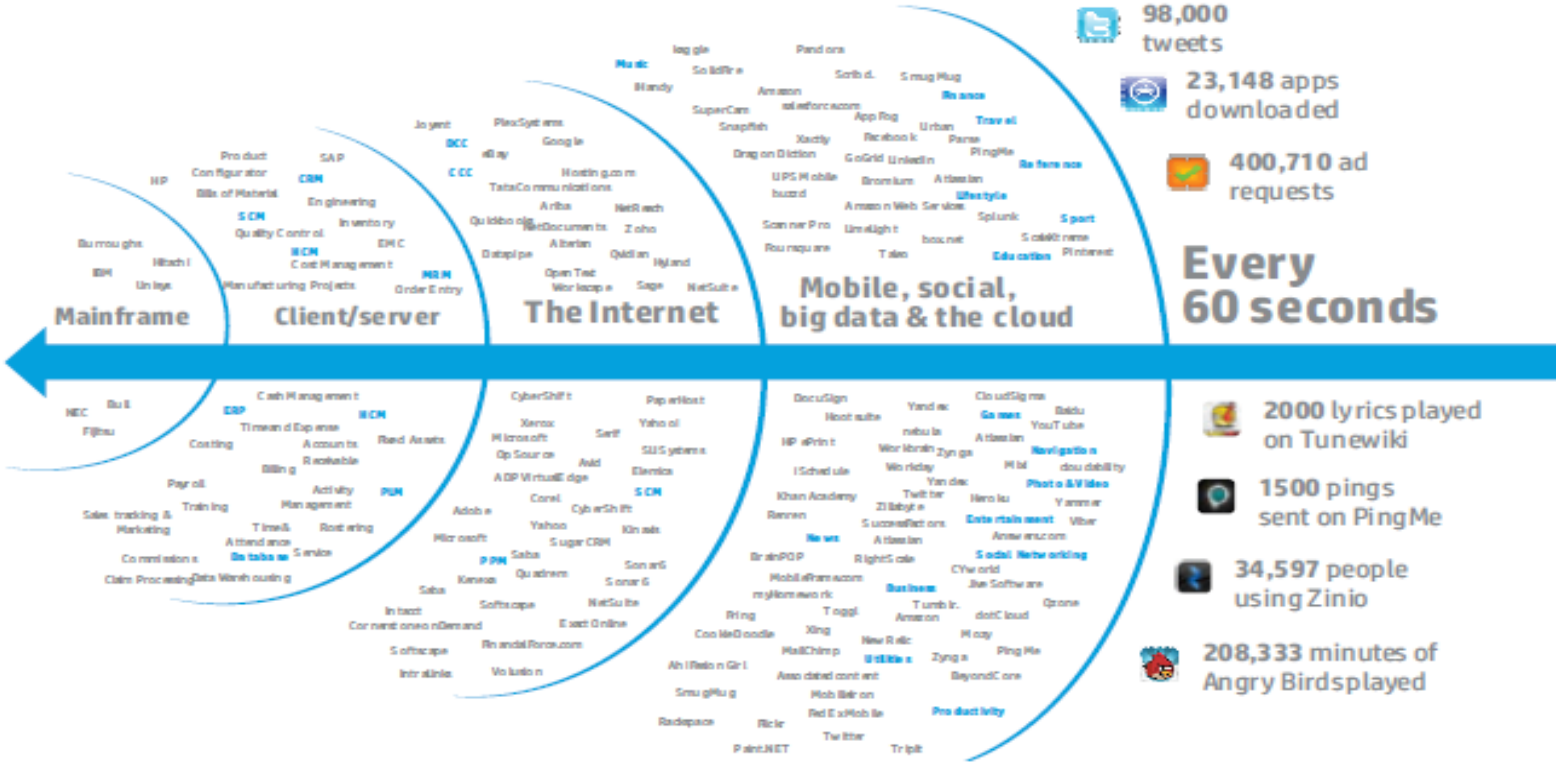
Business Intelligence (con’t)

- Do not look ahead !

Business Intelligence (con't)

- Traditionally this was done with classical business research such as surveys, focus groups, etc.
- But today it comes from tweets, videos, likes, clickstream data, and other social media sources
  - This is called "Big Data"
  - Most of this data is unstructured

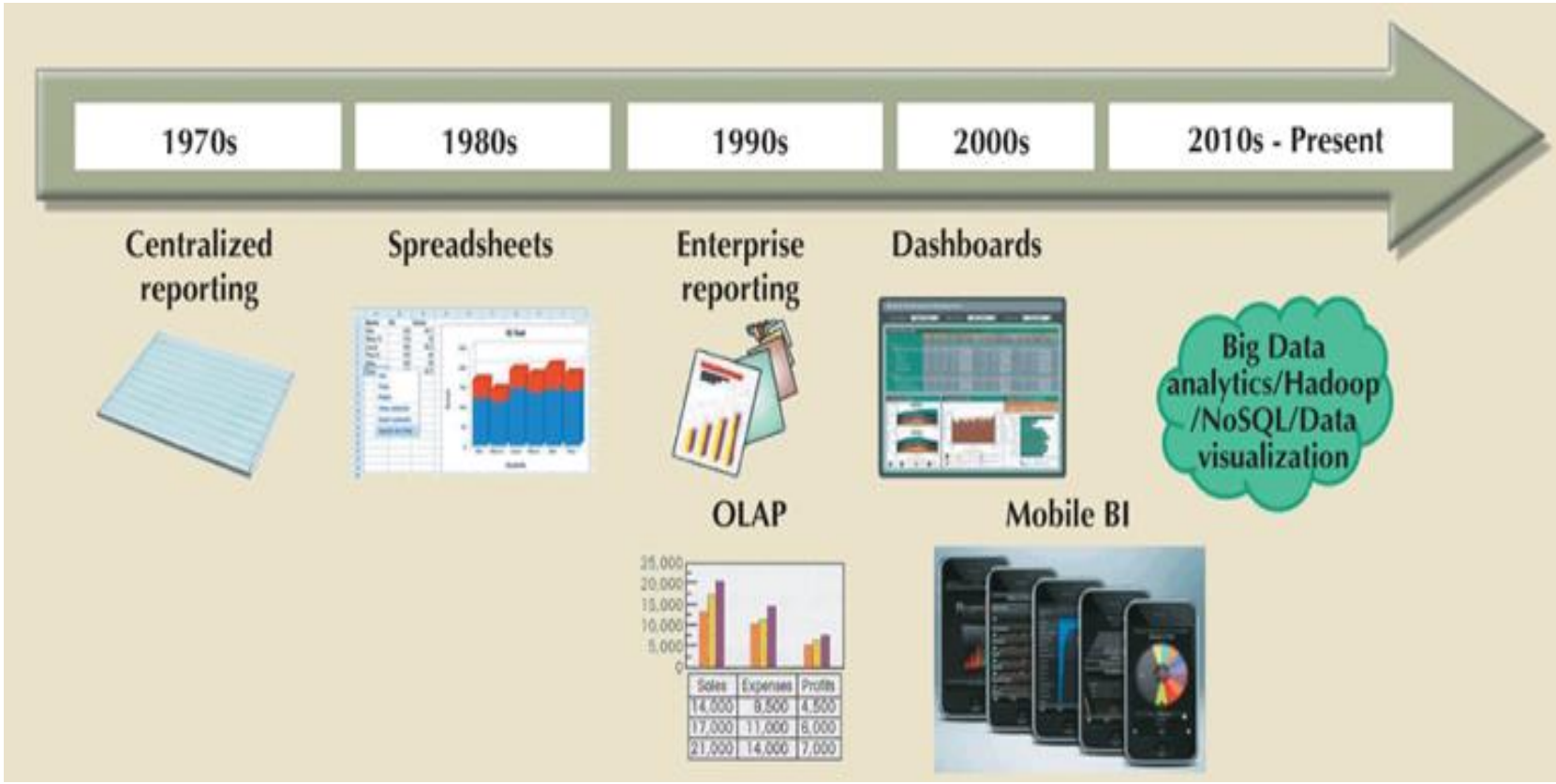
Business Intelligence (con't)



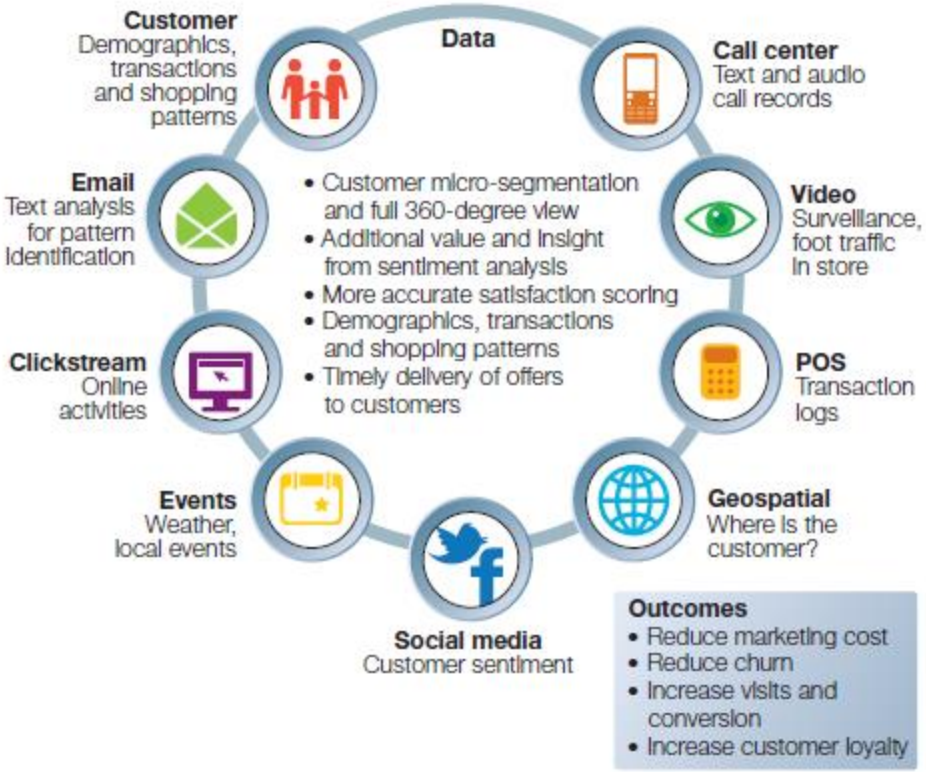
Business Intelligence (con't)

- For example, organizations can correlate sales records with "social mentions"
- So instead of learning which customers it has lost, an organization can learn which customers it may lose, and why, so they can present timely offers or corrections !

Evolution of BI



“360 View of Customers”



“360 View of Customers” (con’t)

- Predict optimal pricing and maintain a price leadership position by analyzing price and demand elasticity
- Select the right merchandise for each channel and fine-tune local assortment planning by drawing on insights from social media, market reports, internal sales data and customer buying patterns
- Optimize inventory across multiple channels by using leading indicators such as customer sentiment and promotional buzz to anticipate future demand
- Improve logistics by using real-time traffic, weather data and more to re-route shipments and avoid costly
- Fine-tune store product placement by analyzing customer buying patterns and purchasing trends (what is a planogram?)

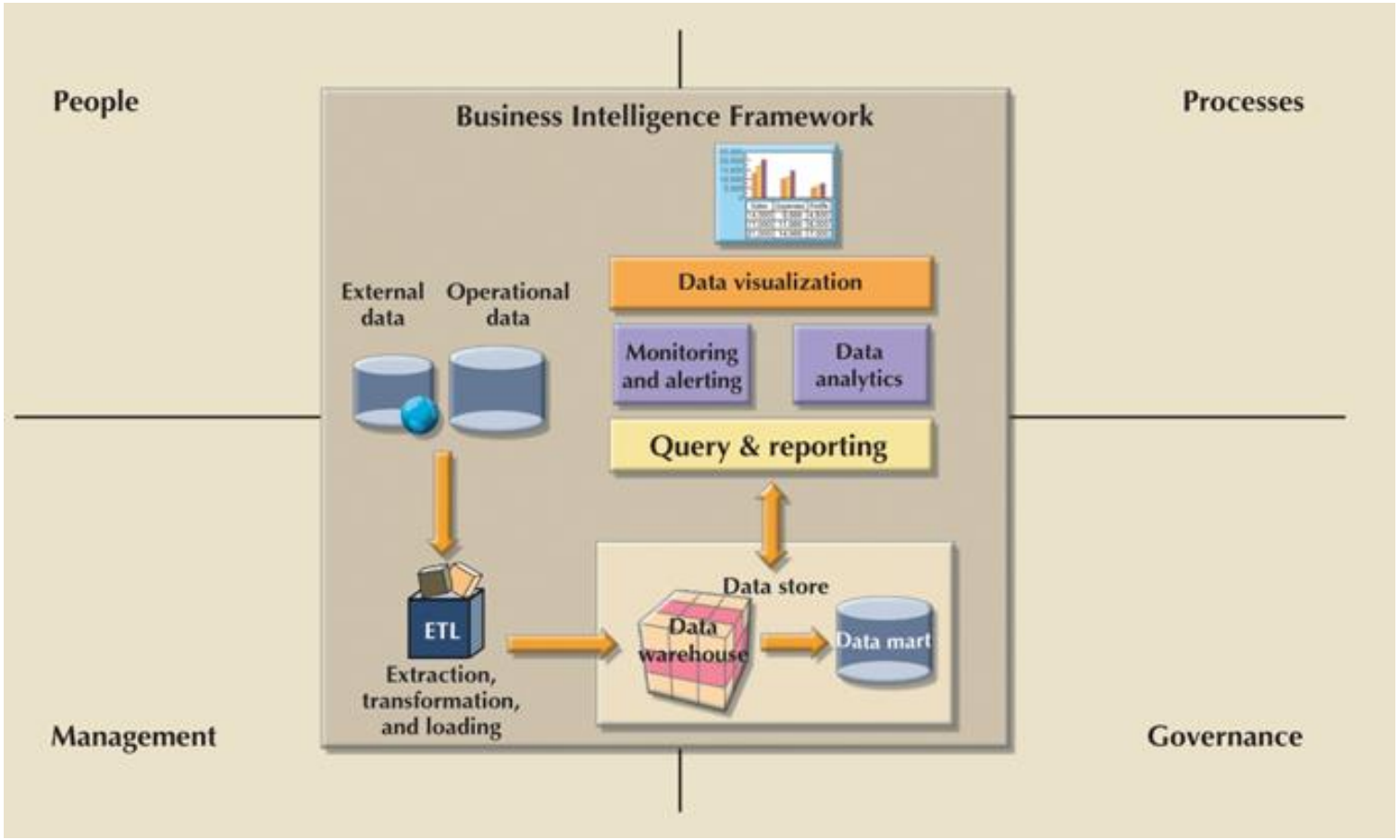
“360 View of Customers” (con’t)

- Optimize staffing levels by predicting changes in customer demand
- Better match employee skills with retail store needs and create the right incentives to drive strong sales performance
- Facilitate better-informed financial decision making by drawing on complete, trustworthy and timely data from a wide array of sources
- Improve fraud detection by analyzing large volumes of transactions

Business Intelligence or Data Analytics

- A broad category of applications and techniques for gathering, storing, analyzing and providing access to basic data and data converted into knowledge
- It helps enterprise users make better business and strategic decisions
- Major applications include the activities of query and reporting, online analytical processing (OLAP), data mining, data visualization, operations research, forecasting, and statistical analysis

BI Framework



BI Architecture Components

Component	Description
ETL tools	Data extraction, transformation, and loading (ETL) tools collect, filter, integrate, and aggregate internal and external data to be saved into a data store optimized for decision support.
Data store	The data store is optimized for decision support and is generally represented by a data warehouse or a data mart. The data is stored in structures that are optimized for data analysis and query speed.
Query and reporting	This component performs data selection and retrieval, and it is used by the data analyst to create queries that access the database and create the required reports.
Data visualization	This component presents data to the end user in a variety of meaningful and innovative ways. This tool helps the end user select the most appropriate presentation format, such as summary reports, maps, pie or bar graphs, mixed graphs, and static or interactive dashboards.
Data monitoring and alerting	This component allows real-time monitoring of business activities. The BI system will present concise information in a single integrated view. This integrated view could include specific metrics about the system performance or activities, such as number of orders placed in the last four hours, number of customer complaints by product by month, and total revenue by region. Alerts can be placed on a given metric; once the value of a metric goes below or above a certain baseline, the system will perform a given action, such as emailing shop floor managers, presenting visual alerts, or starting an application.
Data analytics	This component performs data analysis and data-mining tasks using the data in the data store. This tool advises the user as to which data analysis tool to select and how to build a reliable business data model. Business models are generated by special algorithms that identify and enhance the understanding of business situations and problems.

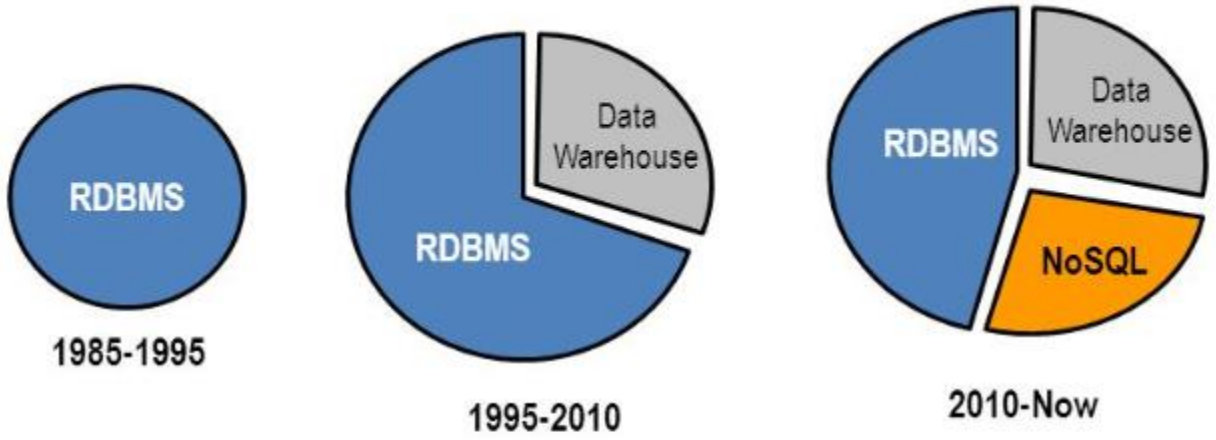
Some Business Intelligence Applications

- Financial modeling and budgeting
- Resource allocation
- Product mix and location
- Customer knowledge
- Marketing, advertising, sales promotions
- Forecasting; seasonality and trends
- Benchmarking (business performance)
- Competitive intelligence

McKinsey Ins't reports that the US faces a shortage of about 200,000 data analytics workers !

Data Warehousing

Database Evolution



- RDBMS for transactions, Data Warehouse for analytics and NoSQL for scalability

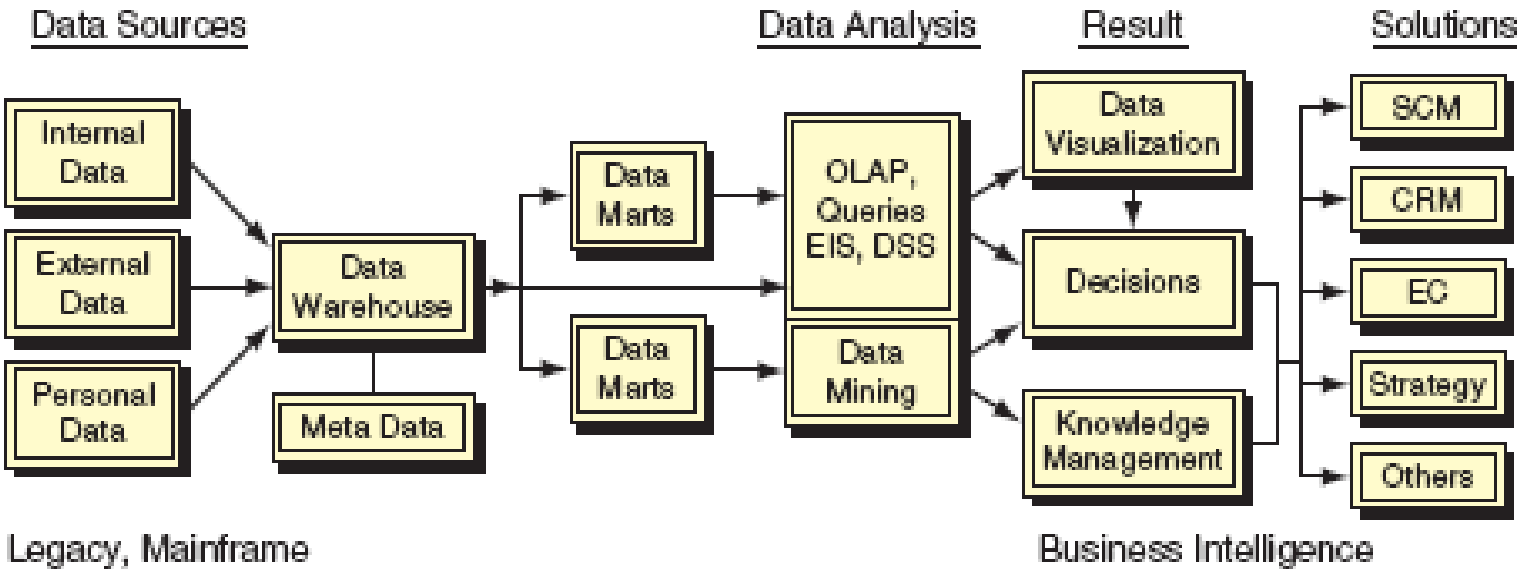
Data Management

BI needs data – lots of data…
However, managing data is difficult for various reasons:

- The amount of data increases exponentially with time
- Data are scattered throughout organizations
- Data are collected by many individuals using several methods
- External data needs to be considered in making organizational decisions
- Some data is unstructured
- Data security, quality, and integrity are critical

Data is an asset; when it is converted to information and knowledge, it gives the firm competitive advantages!

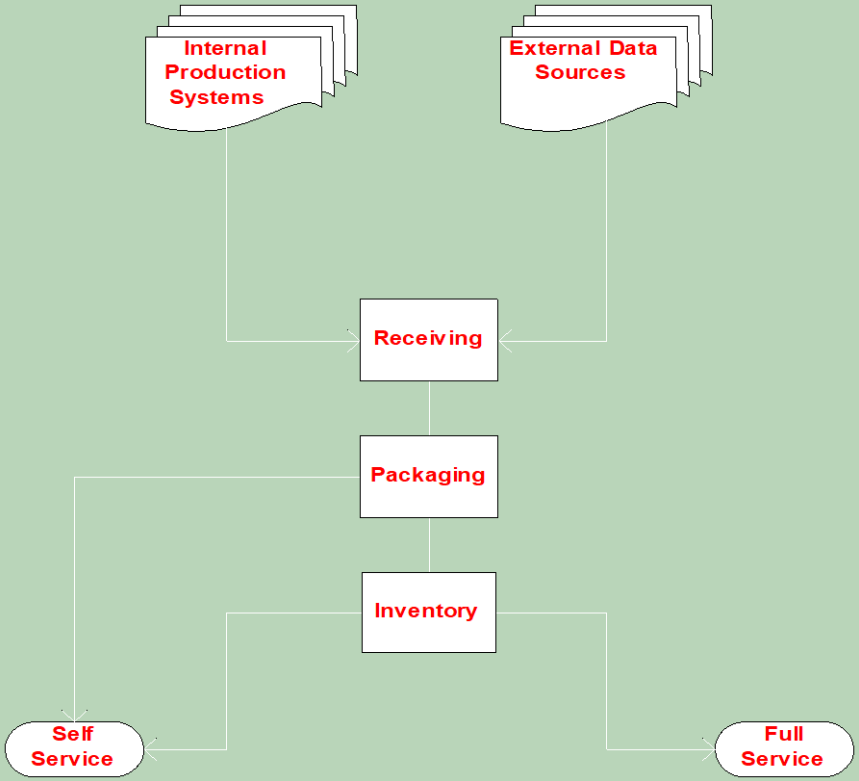
Data Life Cycle Process



“Warehouse” Analogy

- “A collection of data objects that have been received, packaged and inventoried for distribution to a business community”
  - Data objects – relational tables, “cubes” (OLAP, Pivot Tables), and unstructured data
  - Packaged - views (organized for queries)
  - Inventoried - placed in perspective relative to time and other search dimensions
  - Distribution - made available to/on analysis platforms with associated query and analysis tools

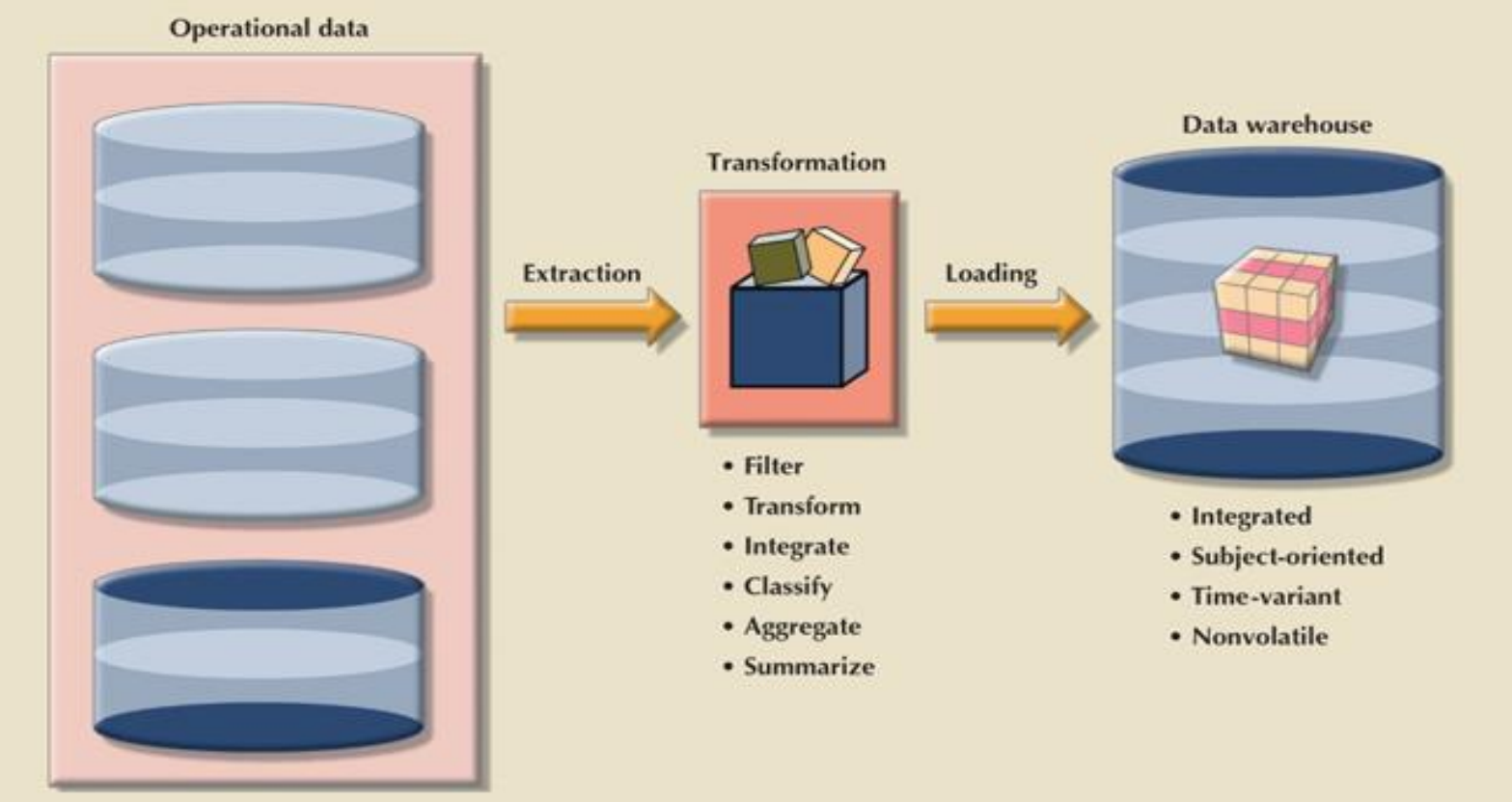
Data Warehouse Analogy



Major Data Warehouse Components
[many jobs at all these levels]

- Data extraction tools (ETL – extract, transform, load)
- Extracted data
- Metadata of warehouse contents
- Warehouse DBMS(s) and OLAP servers
- Warehouse data management tools
- Data delivery programs
- Analysis tools
- User training courses and materials
- Warehouse consultants

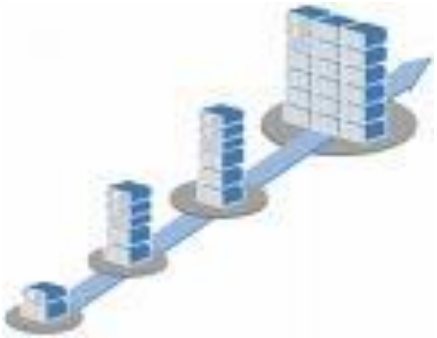
ETL


Operational data
Extraction
Transformation
• Filter
• Transform
• Integrate
• Classify
• Aggregate
• Summarize
Loading
Data warehouse
• Integrated
• Subject-oriented
• Time-variant
• Nonvolatile

Stores, Warehouses, Marts, Catalog, and Cubes

- A data warehouse is a collection of integrated databases designed to support DSS products or systems
- An operational data store (ODS) stores data for a specific application, before that data goes into the DW
- A data mart is a lower-cost, scaled-down version of a data warehouse, usually designed to support a small group of users (rather than the entire firm).
- A Data Catalog shows users where data is located in the organization instead of consolidating data in a warehouse
- Cubes (pivot tables) are arrangements of multi-dimensional data for business intelligence (covered in next session)

Mart vs Warehouse



	Data Mart	Warehouse
Size	50GB	Terabytes
Purpose	Subject Specific	Enterprise Repository
Initiation	Bottom up	Top down
Control	Departmental	Central IS
Imp. Time	3 – 6 months	1 – 2 years
Cost	10000 - 1000000	Millions

The Data Warehouse Environment

- The organization's legacy systems and data stores provide data to the data warehouse or mart(s)
- During the transfer of data from the various sources, **cleansing** or **transformation** may occur, so the data in the DW is more uniform
- Simultaneously, metadata is recorded
- Finally, the DW or mart may be used to create one or more data marts

“Cleaning” or “Scrubbing” of Data

- The process of fixing or eliminating individual pieces of data that are incorrect, incomplete, or duplicated
- Cleaning accounts for about 70% of the cost and effort of implementing most data warehouses
- Dirty data cost U.S. Businesses one trillion dollars annually
- Only about 20% of business with data warehouses use comprehensive scrubbing software

Dirty Data Sources

- What are the sources for dirty data ? Why do we have dirty data ?

- Do not look ahead !

Sources of Dirty Data

- Poor data entry/capture (misspellings, typos, transpositions, variations in spelling)
- Missing fields and relationships
- Lack of enterprise wide coding standards
- Multiple databases with differing metadata
- Older systems with obsolete fields

Cleaning Data via SQL

- --Separate date and time
- SELECT IncidentNum, Date,
- LEFT (Date,10) AS CleanDate,
- RIGHT (Date, LENGTH (Date) -11) AS CleanTime
- FROM INCIDENTS-2017

- --Trim from left, right, or both
- SELECT Location,
- TRIM (both '()' FROM location)
- FROM INCIDENTS-2017

Cleaning Data via SQL (con’t)

- --Convert to upper case
- SELECT IncidentNum, Address,
- UPPER(Address) AS Address_upper,
- LOWER(Address) AS Address_lower
- FROM INCIDENTS-2017

- --Remove Nulls
- SELECT IncidentNum, Descript,
- COALESCE(Descript, 'No Description')
- FROM INCIDENTS-2017

Key Data Warehouse Metadata
[beyond metadata in a RDBMS]

- Data origin
- Data type
- Data format(s)
- Ownership
- Usage constraints
- Useful life
- Security constraints

Key DW Usage Requirements

- Queries and reports with variable structure
- Transformations
- User-specified data aggregation (“slicing and dicing” as needed) [by time, by location, by product, …]
- User-specified drill down
- Graphical outputs (“vizualization”)
- Integration with data mining programs

Typical Data Transformations

- Expanding - expanding codes into meaningful terms
- Snapshots - captures moments in time
- Aggregating - summing, averaging, min, max, ...
- Grouping - collapsing rows into groups/subtotals
- Extrapolating - filling in missing pieces
- Forecasting - filling in future values
- Sampling - drawing an accurate subset for analysis

Data Administration
[at a higher level than the DBA]

- Data are a vital organizational asset that can improve both operations and management tactical and strategic decision making
- Data can be used to obtain and maintain an competitive advantage
- Data is expensive and time consuming to acquire, clean, format, and store
- The purpose of **formal data administration** is to protect the data from unauthorized use and to ensure that the data is used effectively (to protect this vital asset)

Data Administration (con't)

- Data administration is a function analogous to a financial controller
  - A controller's function is to not only protect financial assets but to ensure that they are used effectively
  - Simply locking the money away in a vault will protect it, but the funds will not have been used effectively (reference the biblical tale of the servant that buries his "talents")
- **Data must be used effectively, and a data warehouse or data mart is a first step in this direction !**

- What is “The Parable of the Talents” and how does it relate to data administration ?

- Jesus often told a story to teach a lesson. Hear the story of the talents.
- One day a man was going on a long trip. He needed his servants to take care of his property while he was gone, so he called them to him.
- To the first servant he gave five talents of money. (A talent was not a coin, but a weight of a precious metal such as silver, and one talent was worth more than $1,000. So this servant received money worth more than $5,000.)
- This man went to work at once using his money until he had doubled it. He now had ten talents instead of five.
- The master gave the second man two talents. He probably thought the man was capable of managing that amount of money. The second man was successful also, and doubled his money. He began with more than $2,000 and now he had twice as much.
- The third man was not as capable as the other two, but the master gave him one talent with the expectation that he would manage it well. He, too, could have increased his money, but he dug a hole and hid it in the ground.
- After a long time the master returned. (Some think the master's trip is a picture of Jesus returning to heaven, and the return from the trip is the judgment of mankind.) He was ready for a report from the servants.
- The man who had received five talents brought his money and showed the master that he had doubled it. He was happy to show his master the results.
- The master was well pleased. He said, "Well done, good and faithful servant! You have been faithful with a few things; I will put you in charge of many things. Come and share your master's happiness!"
- The man that had been given two talents showed the master that he had also doubled his money. He received the same words of praise as the first man who had received five talents.
- The man who had received one talent dug up the talent he had buried and brought it to the master. He accused the master of being a hard man to work for, said he had been afraid, so he just buried his talent. He gave it back to the master saying, "See, here is what belongs to you."
- The master was very angry with him and called him a wicked, lazy servant. He said the man should at least have put the money with bankers and received some interest. (Now the Jews could not charge nor receive interest from a fellow Jew, but they could get interest from a person who was not a Jew.)
- The master took his one talent away from him and gave it to the man who had ten talents, and the one talent man was punished because he had not properly used the talent he had been given.
- What can we learn from this story? We need to use whatever "talent" God has given us. It might be money or ability. If we use it wisely, He will increase it so that our lives will glorify Him.

University Library Analogy

- Consider a university library
- What are the administrative functions that have to be done in regard to the information assets in a library ?

- Do not look ahead !

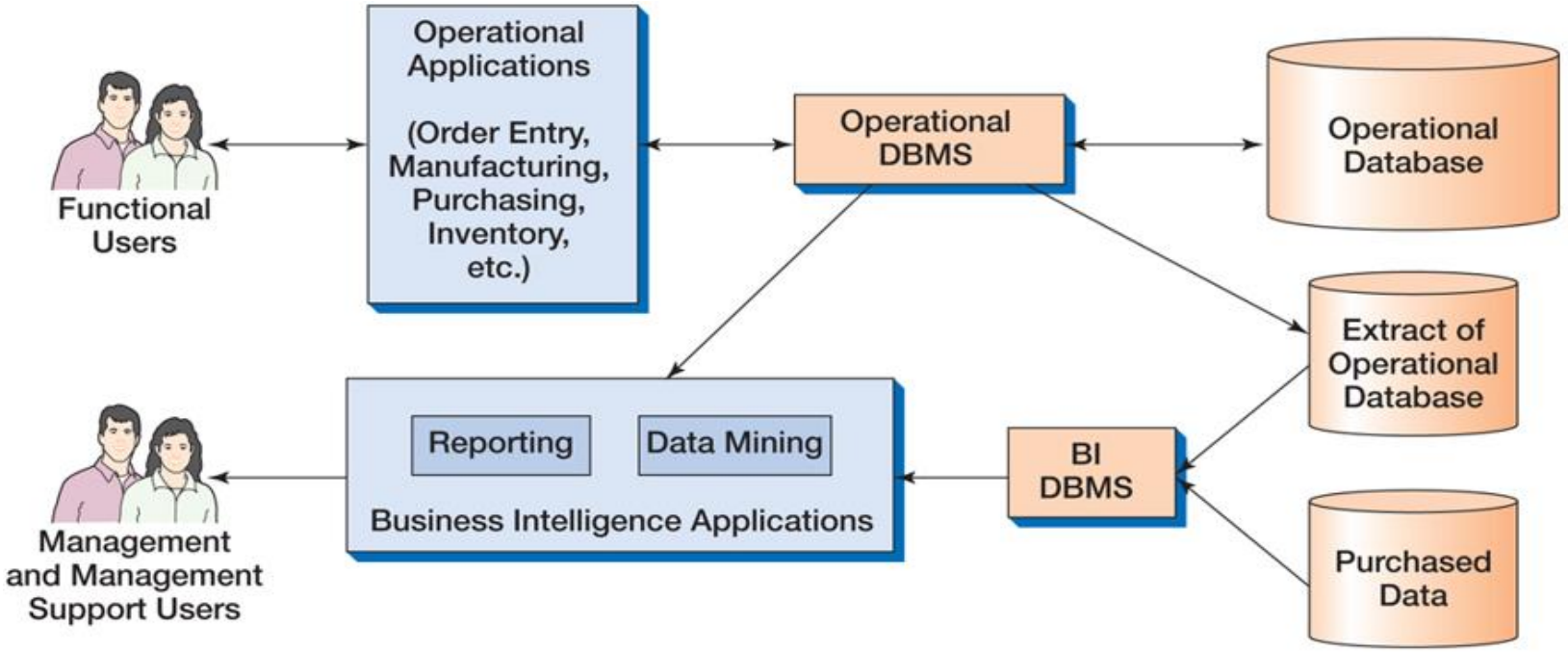
University Library Analogy

- A library contains thousands of books, journals, reports, videos/dvds, cds, etc.
- They are of no value just sitting on the shelf !
- The items must be distributed and used, but at the same time safeguarded
- To distribute these assets, the items must be organized both physically and logically (indexed); they can only be physically arranged one way, but they can be logically arranged many ways (author, title, isbn, loc #, subject, etc.)
- Each item also needs a unique identifier
- Some items may be physically removed form the library and others may not, there may be other constraints
- Users must have an effective and convenient way to search for items (card catalog, pc's in library, internet access)
- How are the library's assets marketed and advertised to potential users ?
- How is effective usage measured ?
- What if there are several libraries on campus ?
- What if some departments have their own libraries ?
- How do we measure the overall utility (and ROI) of the library(s) ?

Data Administration Challenges

- Is a warehouse a place or activity?
- Should the warehouse be separate from production systems?
- Is the relational model sufficient?
- What is the right amount of granularity?
- Should a warehouse permit updates?
- How should data freshness be maintained?
- Many types of data exist
- Basic categories of data are not obvious
- The same data can have many names
- The same data can have many descriptions and formats
- Can a warehouse handle semantic heterogeneity?
- What is the balance between the enterprise versus the workgroup?
- Should a warehouse support distribution?
- Who owns warehouse management?
- Political-organizational issues complicate operational issues

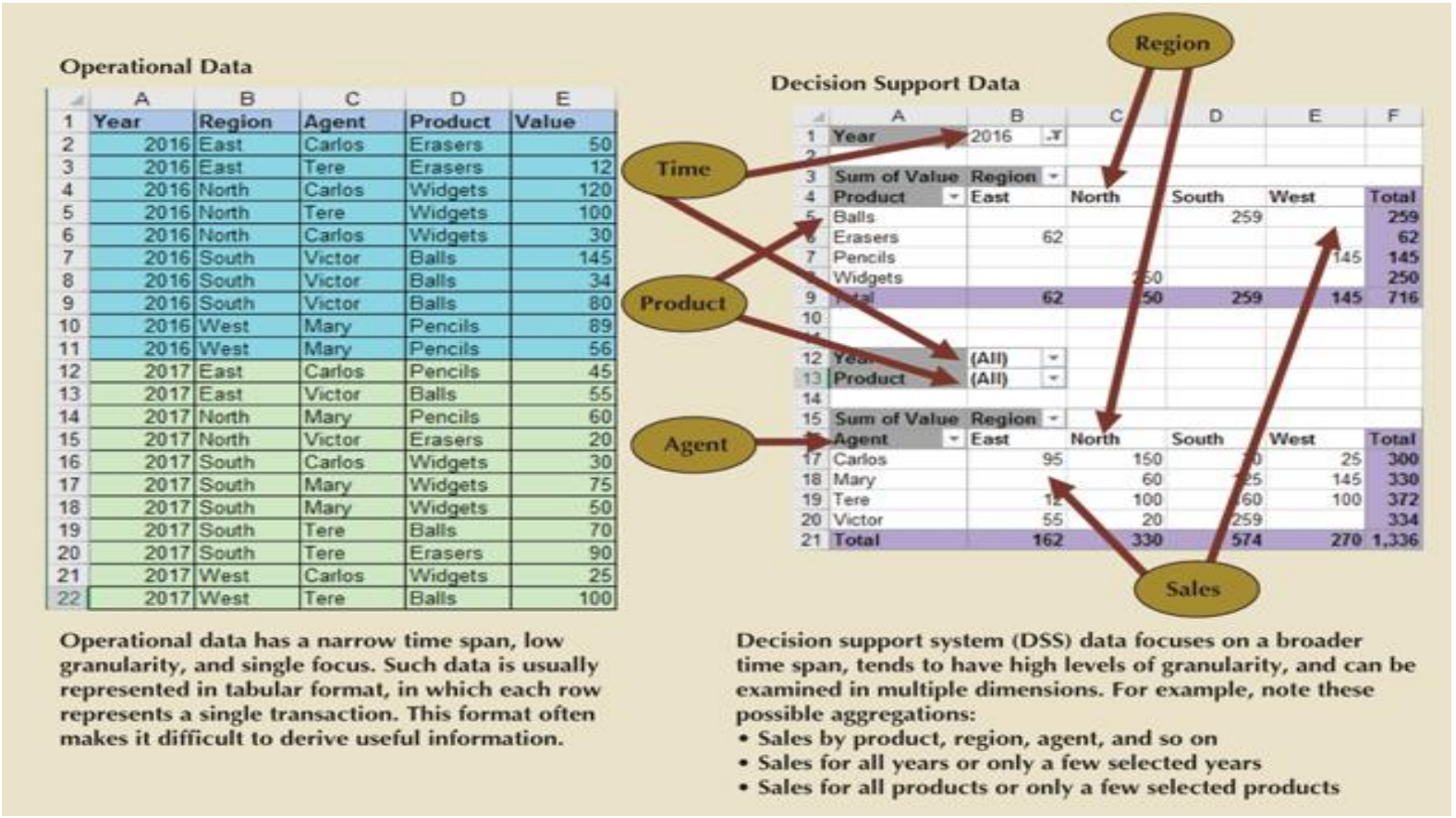
Typically Different DBMS for Operations vs BI



Operational vs BI (con't)

- Operational data and decision support data (BI) serve different purposes
  - Operational data is useful for capturing daily business transactions
  - Decision support data gives tactical and strategic business meaning to the operational data
- Decision support data differs from operational data in three main areas
  - Time span, granularity (level of aggregation), dimensionality

Operational vs BI (con't)



Operational Data

	A	B	C	D	E
1	Year	Region	Agent	Product	Value
2	2016	East	Carlos	Erasers	50
3	2016	East	Tere	Erasers	12
4	2016	North	Carlos	Widgets	120
5	2016	North	Tere	Widgets	100
6	2016	North	Carlos	Widgets	30
7	2016	South	Victor	Balls	145
8	2016	South	Victor	Balls	34
9	2016	South	Victor	Balls	80
10	2016	West	Mary	Pencils	89
11	2016	West	Mary	Pencils	56
12	2017	East	Carlos	Pencils	45
13	2017	East	Victor	Balls	55
14	2017	North	Mary	Pencils	60
15	2017	North	Victor	Erasers	20
16	2017	South	Carlos	Widgets	30
17	2017	South	Mary	Widgets	75
18	2017	South	Mary	Widgets	50
19	2017	South	Tere	Balls	70
20	2017	South	Tere	Erasers	90
21	2017	West	Carlos	Widgets	25
22	2017	West	Tere	Balls	100

Operational data has a narrow time span, low granularity, and single focus. Such data is usually represented in tabular format, in which each row represents a single transaction. This format often makes it difficult to derive useful information.

Decision Support Data

	A	B	C	D	E	F
1	Year	2016				
2						
3	Sum of Value	Region				
4	Product	East	North	South	West	Total
5	Balls			259		259
6	Erasers	62				62
7	Pencils				145	145
8	Widgets		250			250
9	Total	62	250	259	145	716
10						
11						
12	Year	(All)				
13	Product	(All)				
14						
15	Sum of Value	Region				
16	Agent	East	North	South	West	Total
17	Carlos	95	150	30	25	300
18	Mary		60	125	145	330
19	Tere	12	100	160	100	372
20	Victor	55	20	259		334
21	Total	162	330	574	270	1,336

Decision support system (DSS) data focuses on a broader time span, tends to have high levels of granularity, and can be examined in multiple dimensions. For example, note these possible aggregations:

- Sales by product, region, agent, and so on
- Sales for all years or only a few selected years
- Sales for all products or only a few selected products

Operational vs BI (con't)

Characteristic	Operational Data	Decision Support Data
Data currency	Current operations Real-time data	Historic data Snapshot of company data Time component (week/month/year)
Granularity	Atomic-detailed data	Summarized data
Summarization level	Low; some aggregate yields	High; many aggregation levels
Data model	Highly normalized Mostly relational DBMSs	Non-normalized Complex structures Some relational, but mostly multidimensional DBMSs
Transaction type	Mostly updates	Mostly query
Transaction volumes	High-update volumes	Periodic loads and summary calculations
Transaction speed	Updates are critical	Retrievals are critical
Query activity	Low to medium	High
Query scope	Narrow range	Broad range
Query complexity	Simple to medium	Very complex
Data volumes	Hundreds of gigabytes	Terabytes to petabytes

Operational vs BI (con't)

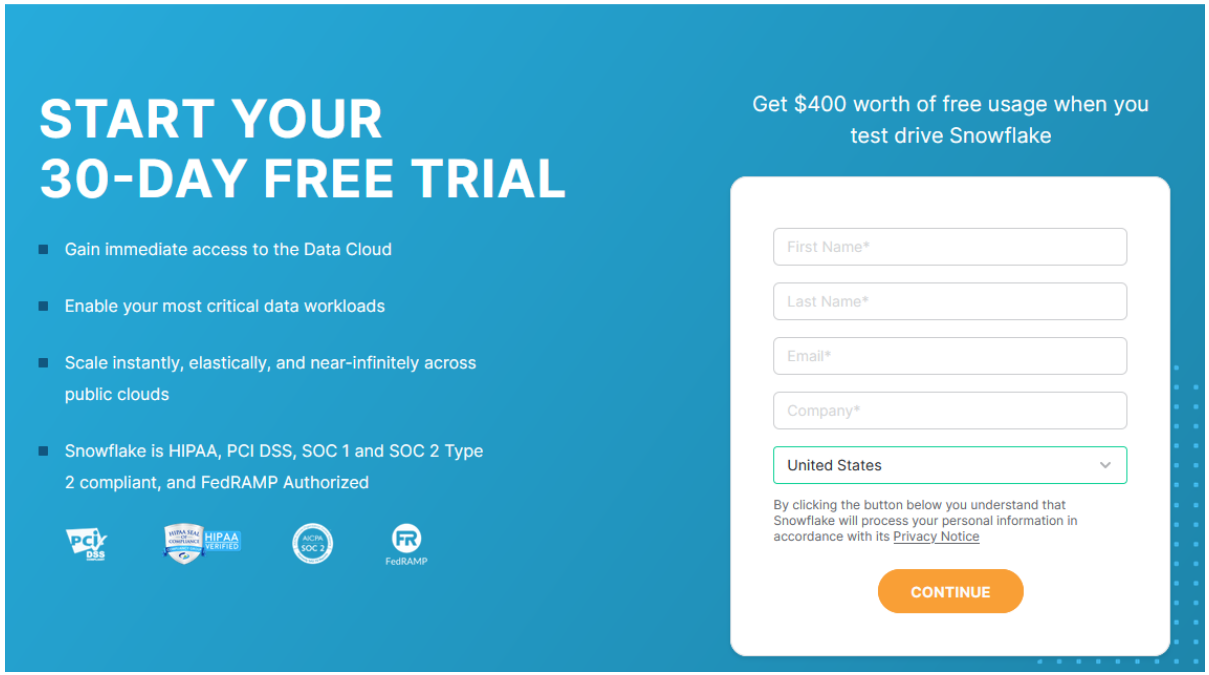
Characteristic	Operational Database Data	Data Warehouse Data
Integrated	Similar data can have different representations or meanings. For example, Social Security numbers may be stored as ###-##-#### or as #########, and a given condition may be labeled as T/F or 0/1 or Y/N. A sales value may be shown in thousands or in millions.	Provide a unified view of all data elements with a common definition and representation for all business units.
Subject-oriented	Data is stored with a functional, or process, orientation. For example, data may be stored for invoices, payments, and credit amounts.	Data is stored with a subject orientation that facilitates multiple views of the data and decision making. For example, sales may be recorded by product, division, manager, or region.
Time-variant	Data is recorded as current transactions. For example, the sales data may be the sale of a product on a given date, such as $342.78 on 12-MAY-2016.	Data is recorded with a historical perspective in mind. Therefore, a time dimension is added to facilitate data analysis and various time comparisons.
Nonvolatile	Data updates are frequent and common. For example, an inventory amount changes with each sale. Therefore, the data environment is fluid.	Data cannot be changed. Data is added only periodically from historical systems. Once the data is properly stored, no changes are allowed. Therefore, the data environment is relatively static.

12 Data Warehouse Rules

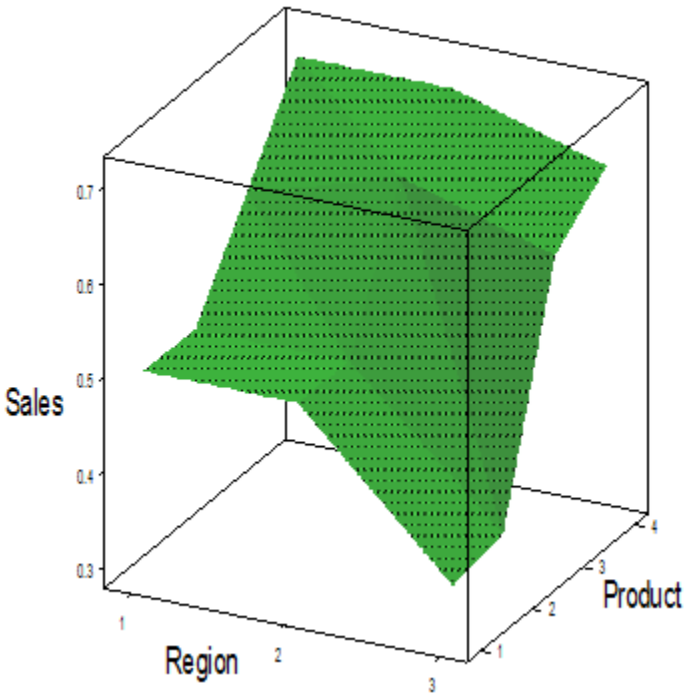
Rule No.	Description
1	The data warehouse and operational environments are separated.
2	The data warehouse data is integrated.
3	The data warehouse contains historical data over a long time.
4	The data warehouse data is snapshot data captured at a given point in time.
5	The data warehouse data is subject oriented.
6	The data warehouse data is mainly read-only with periodic batch updates from operational data. No online updates are allowed.
7	The data warehouse development life cycle differs from classical systems development. Data warehouse development is data-driven; the classical approach is process-driven.
8	The data warehouse contains data with several levels of detail: current detail data, old detail data, lightly summarized data, and highly summarized data.
9	The data warehouse environment is characterized by read-only transactions to very large data sets. The operational environment is characterized by numerous update transactions to a few data entities at a time.
10	The data warehouse environment has a system that traces data sources, transformations, and storage.
11	The data warehouse's metadata is a critical component of this environment. The metadata identifies and defines all data elements. The metadata provides the source, transformation, integration, storage, usage, relationships, and history of each data element.
12	The data warehouse contains a chargeback mechanism for resource usage that enforces optimal use of the data by end users.

Cloud Data Warehouse

- SnowFlake – 4 billion dollar IPO



Online Analytical Processing (OLAP)



Traditional Queries

Queries allow users to request information from the computer that is not available in periodic reports
Query systems are often based on menu/GUI based programs (which generate SQL) or via direct structured query language (SQL) or using a query-by-example (QBE) method

- User requests are stated in a query language and the results are subsets of the data in the relational tables:
  - Sales by department by customer type for specific period
  - Weather conditions for specific date
  - Sales by day of week
  - ...

Changes in Computing

	1950's	1960's	1970's	1980's	1990's	2000's	2010's
Hardware Technology	Vacuum Tubes	Transis tors	Integrated Circuits	LSI	VLSI	ULSI	Nano-systems
Programming Languages	Binary Assembly	Fortran Cobol	Pascal Algol	Ada C Lisp	C++ GUI Java	C# PhP XML	Python, F#
Computing Paradigm	1 user Mainframe	Batch	Time Sharing	Personal Computer	LAN, WEB	.NET, SOA	Mobile
Operating System	none	1 user	multi user	multi user linked	networked	Web, Open source	Cloud, Android, iPhone
Data Base Methods	none	Linear (tapes)	Hier-archical	Relational	Object Oriented	SQL, X Query	SQLJ, OLAP, JDBC
Software Design	pad and pencil	Flow Charts	Structured Design	Data Flow	Object Oriented	RAD, XP, RUP	MDE

NoSql

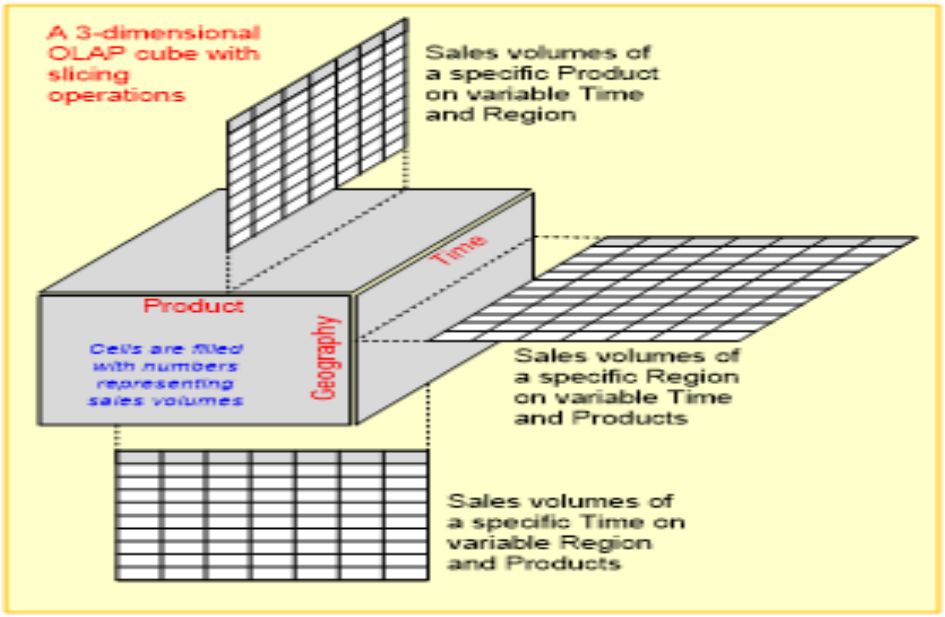
OLAP

- On Line Analytical Processing (OLAP) is a relatively new way of storing, viewing, and presenting information
- With it, data is viewed in cubes
- A two dimensional cube can be viewed as a table
- A three dimensional cube as a “cube”
- A multidimensional cube as a “hypercube”
- These cubes have axes, dimensions, measures, slices, and levels

OLTP vs OLAP

Criteria	OLTP	OLAP
Purpose	To carry out day-to-day business functions	To support decision making and provide answers to business and management queries
Data source	Transaction database (a normalized data repository primarily focused on efficiency and consistency)	Data warehouse or DM (a nonnormalized data repository primarily focused on accuracy and completeness)
Reporting	Routine, periodic, narrowly focused Reports	Ad hoc, multidimensional, broadly focused reports and queries
Resource requirements	Ordinary relational databases	Multiprocessor, large-capacity, specialized databases
Execution speed	Fast (recording of business transactions and routine reports)	Slow (resource intensive, complex, large-scale queries)

OLAP Cube


A 3-dimensional OLAP cube with slicing operations
Sales volumes of a specific Product on variable Time and Region
Time
Product
Geography
Cells are filled with numbers representing sales volumes
Sales volumes of a specific Region on variable Time and Products
Sales volumes of a specific Time on variable Region and Products

Example: Relational Source Data

Category	Type	City	State	Date	Sales Price	Asking Price
New	Single Family	San Francisco	California	1/1/2000	679,000	685,000
Existing	Condo	Los Angeles	California	3/5/2001	327,989	350,000
Existing	Single Family	Elko	Nevada	7/17/2001	105,675	125,000
New	Condo	San Diego	California	12/22/2000	375,000	375,000
Existing	Single Family	Paradise	California	11/19/2001	425,000	449,000
Existing	Single Family	Las Vegas	Nevada	1/19/2001	317,000	325,000
New	Single Family	San Francisco	California	1/1/2000	679,000	685,000
Existing	Condo	Los Angeles	California	3/5/2001	327,989	350,000
Existing	Condo	Las Vegas	Nevada	6/19/2001	297,000	305,000
Existing	Single Family	Los Angeles	California	4/1/2000	579,000	625,000
New	Condo	Los Angeles	California	8/5/2001	321,000	320,000
Etc.						

What is the average sales price for new single family homes in LA in the 2QT of 2001 ?

Example: OLAP Cube for Average Sales Price

[2 "axes" (rows and columns): date "dimensions" and type "dimensions"]

Average Sales Price of Single-Family Dwellings ($thousands)										
			Existing Structures				New Construction			
			California			Nevada	California			Nevada
			San Francisco	Los Angeles	San Diego		San Francisco	Los Angeles	San Diego	
2000	Q1	Jan	408	465	375	179	418	468	371	190
		Feb	419	438	382	180	429	437	382	185
		Mar	427	477	380	195	426	471	387	198
	Q2		433	431	382	188	437	437	380	193
	Q3		437	437	380	190	438	439	382	190
	Q4		435	439	377	193	432	434	370	198
2001	Q1	Jan	452	454	368	198	450	457	367	197
		Feb	450	467	381	187	457	464	388	191
		Mar	432	444	373	188	436	446	371	201
	Q2		437	437	368	190	444	432	363	196
	Q3		436	452	388	196	447	455	385	199
	Q4		441	455	355	198	449	455	355	202

What is the average sales price for new single family homes in LA in the 2QT of 2001 ?

OLAP (con't)

Average Sales Price of Single-Family Dwellings ($thousands)										
			Existing Structures				New Construction			
			California			Nevada	California			Nevada
			San Francisco	Los Angeles	San Diego		San Francisco	Los Angeles	San Diego	
2000	Q1	Jan	408	465	375	179	418	468	371	190
		Feb	419	438	382	180	429	437	382	185
		Mar	427	477	380	195	426	471	387	198
	Q2		433	431	382	188	437	437	380	193
	Q3		437	437	380	190	438	439	382	190
	Q4		435	439	377	193	432	434	370	198
2001	Q1	Jan	452	454	368	198	450	457	367	197
		Feb	450	467	381	187	457	464	388	191
		Mar	432	444	373	188	436	446	371	201
	Q2		437	437	368	190	444	432	363	196
	Q3		436	452	388	196	447	455	385	199
	Q4		441	455	355	198	449	455	355	202

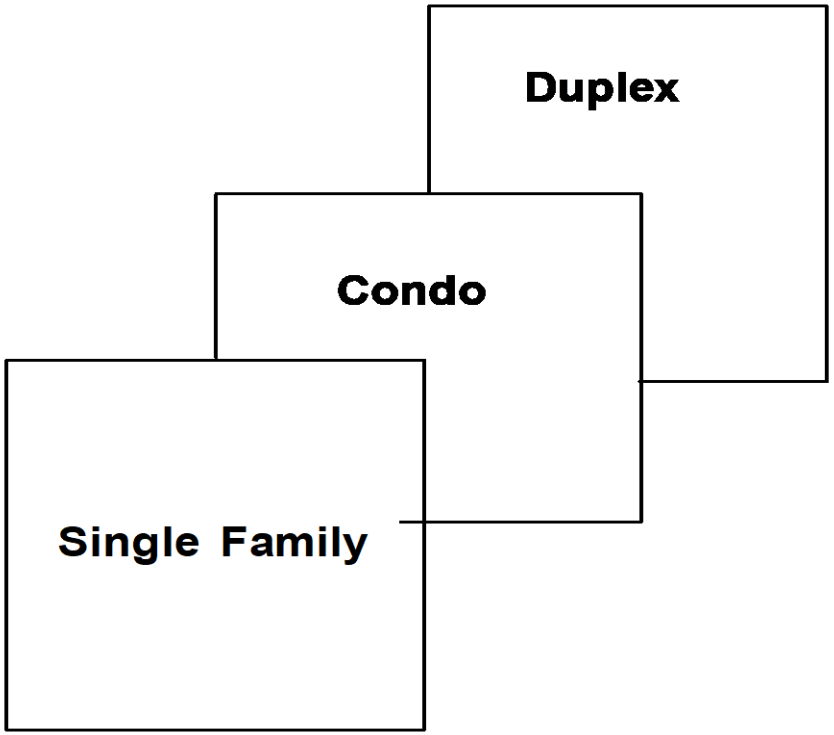
- When 2 or more dimensions are shown on one axis, then every combination (relational column) of one must be shown with the other
- Notice the same sub categories under both existing and new construction categories, and same categories under 2000 and 2001
- The cells of the OLAP cube hold the "measures" (the data); here the measure is sales price for single family homes

OLAP (con't)

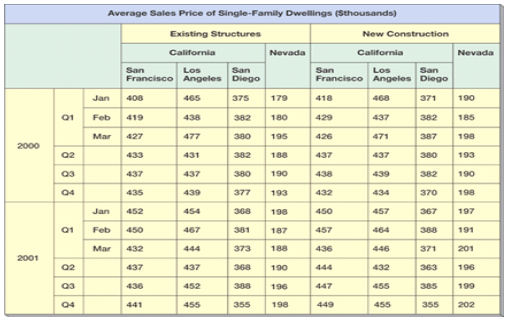
Average Sales Price of Single-Family Dwellings ($thousands)										
			Existing Structures				New Construction			
			California			Nevada	California			Nevada
			San Francisco	Los Angeles	San Diego		San Francisco	Los Angeles	San Diego	
2000	Q1	Jan	408	465	375	179	418	468	371	190
		Feb	419	438	382	180	429	437	382	185
		Mar	427	477	380	195	426	471	387	198
	Q2		433	431	382	188	437	437	380	193
	Q3		437	437	380	190	438	439	382	190
	Q4		435	439	377	193	432	434	370	198
2001	Q1	Jan	452	454	368	198	450	457	367	197
		Feb	450	467	381	187	457	464	388	191
		Mar	432	444	373	188	436	446	371	201
	Q2		437	437	368	190	444	432	363	196
	Q3		436	452	388	196	447	455	385	199
	Q4		441	455	355	198	449	455	355	202

- This OLAP cube is just for the average sales price of single family homes; there would be another cube for the average sales price of condos
- You could think of these two cubes as one behind the other, or as "slices"
- We could also have slices for "sales price" and "asking price"

Slices



“Members” and “Levels”

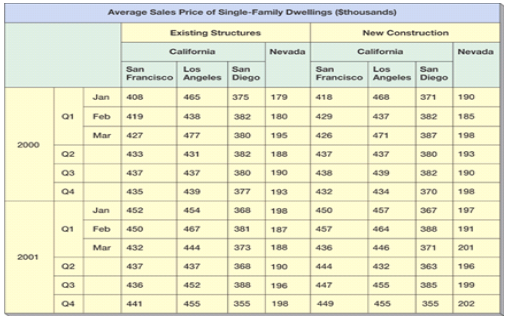


Average Sales Price of Single-Family Dwellings ($thousands)											
				Existing Structures				New Construction			
				California			Nevada	California			Nevada
				San Francisco	Los Angeles	San Diego		San Francisco	Los Angeles	San Diego	
2000	Q1		Jan	408	465	375	179	418	468	371	190
			Feb	419	438	382	180	429	437	382	185
			Mar	427	477	380	195	426	471	387	198
	Q2			433	431	382	188	437	437	380	193
	Q3			437	437	380	190	438	439	382	190
	Q4			435	439	377	193	432	434	370	198
2001	Q1		Jan	452	454	368	198	450	457	367	197
			Feb	450	467	381	187	457	464	388	191
			Mar	432	444	373	188	436	446	371	201
	Q2			437	437	368	190	444	432	363	196
	Q3			436	452	388	196	447	455	385	199
	Q4			441	455	355	198	449	455	355	202

- The values of a dimension are called “members”
- The members of the type dimension are single and condo
- The members of the category dimension are new and existing
- For this data set, the members of the state dimension are CA an NV
- Some members may be computed such as date and/or time
- The “level” of a dimension is its position in the hierarchy; the levels of the date dimension are year, quarter, and month

OLAP Terminology

- OLAP hypercube: means a data display with an unlimited number of axes



Average Sales Price of Single-Family Dwellings ($thousands)

			Existing Structures				New Construction			
			California			Nevada	California			Nevada
			San Francisco	Los Angeles	San Diego		San Francisco	Los Angeles	San Diego	
2000	Q1	Jan	408	465	375	179	418	468	371	190
		Feb	419	438	382	180	429	437	382	185
		Mar	427	477	380	195	426	471	387	198
	Q2		433	431	382	188	437	437	380	193
	Q3		437	437	380	190	438	439	382	190
	Q4		435	439	377	193	432	434	370	198
2001	Q1	Jan	452	454	368	198	450	457	367	197
		Feb	450	467	381	187	457	464	388	191
		Mar	432	444	373	188	436	446	371	201
	Q2		437	437	368	190	444	432	363	196
	Q3		436	452	388	196	447	455	385	199
	Q4		441	455	355	198	449	455	355	202

Term	Description	Example in Figure
Axis	A coordinate of the hypercube	Rows, columns
Dimension	A feature of the data to be placed on an axis	Time, Housing Type, Location
Level	A (hierarchical) subset of a dimension	{California, Nevada} {San Francisco, Los Angeles, Other} {Q1, Q2, Q3, Q4}
Member	A data value in a dimension	{New, Existing}, {Jan, Feb, Mar}
Measure	The source data for the hypercube	Sales Price, Asking Price
Slice	A dimension or measure held constant for the display	Housing Type—all shown are for Single Family—another cube exists for Condo

OLAP Cube Data Definition

[4 dimensions, 2 slices (sales and asking price)]

```
CREATE CUBE HousingSalesCube (
    DIMENSION Time TYPE TIME,
        LEVEL Year TYPE YEAR,
        LEVEL Quarter TYPE QUARTER,
        LEVEL Month TYPE MONTH,
    DIMENSION Location,
        LEVEL USA TYPE ALL,
        LEVEL State,
        LEVEL City,
    DIMENSION HousingCategory,
    DIMENSION HousingType,
    MEASURE SalesPrice,
        FUNCTION AVG
    MEASURE AskingPrice,
        FUNCTION AVG
        )
```

Compare to SQL Data Definition Language: “create table”

Multidimensional SELECT Statement

[produces previously shown view]

```
SELECT CROSSJOIN
        ({Existing Structure, New Construction},
        {California.Children, Nevada})
        ON COLUMNS,
        {2000.Q1.Children, 2000.Q2, 2000.Q3,
            2000.Q4,
        2001.Q1.Children, 2001.Q2, 2001.Q3,
            2001.Q4}
        ON ROWS
FROM HousingSalesCube
WHERE (SalesPrice, HousingType =
    'SingleFamily')
```

Average Sales Price of Single-Family Dwellings ($thousands)										
			Existing Structures				New Construction			
			California			Nevada	California			Nevada
			San Francisco	Los Angeles	San Diego		San Francisco	Los Angeles	San Diego	
2000	Q1	Jan	408	465	375	179	418	468	371	190
		Feb	419	438	382	180	429	437	382	185
		Mar	427	477	380	195	426	471	387	198
	Q2		433	431	382	188	437	437	380	193
	Q3		437	437	380	190	438	439	382	190
	Q4		435	439	377	193	432	434	370	198
2001	Q1	Jan	452	454	368	198	450	457	367	197
		Feb	450	467	381	187	457	464	388	191
		Mar	432	444	373	188	436	446	371	201
	Q2		437	437	368	190	444	432	363	196
	Q3		436	452	388	196	447	455	385	199
	Q4		441	455	355	198	449	455	355	202

Cross Join

```
SELECT CROSSJOIN
    ({Existing Structure, New Construction},
    {California.Children, Nevada})
    ON COLUMNS,
    {2000.Q1.Children, 2000.Q2, 2000.Q3,
      2000.Q4,
    2001.Q1.Children, 2001.Q2, 2001.Q3,
      2001.Q4}
    ON ROWS
FROM HousingSalesCube
WHERE (SalesPrice, HousingType =
  'SingleFamily')
```

- The “crossjoin” ({X,Y}, {A,B}) creates a view
  - where X and Y are the main categories
  - A and B are sub categories under both X and Y
- Crossjoins are created on the columns or rows

Related Tables

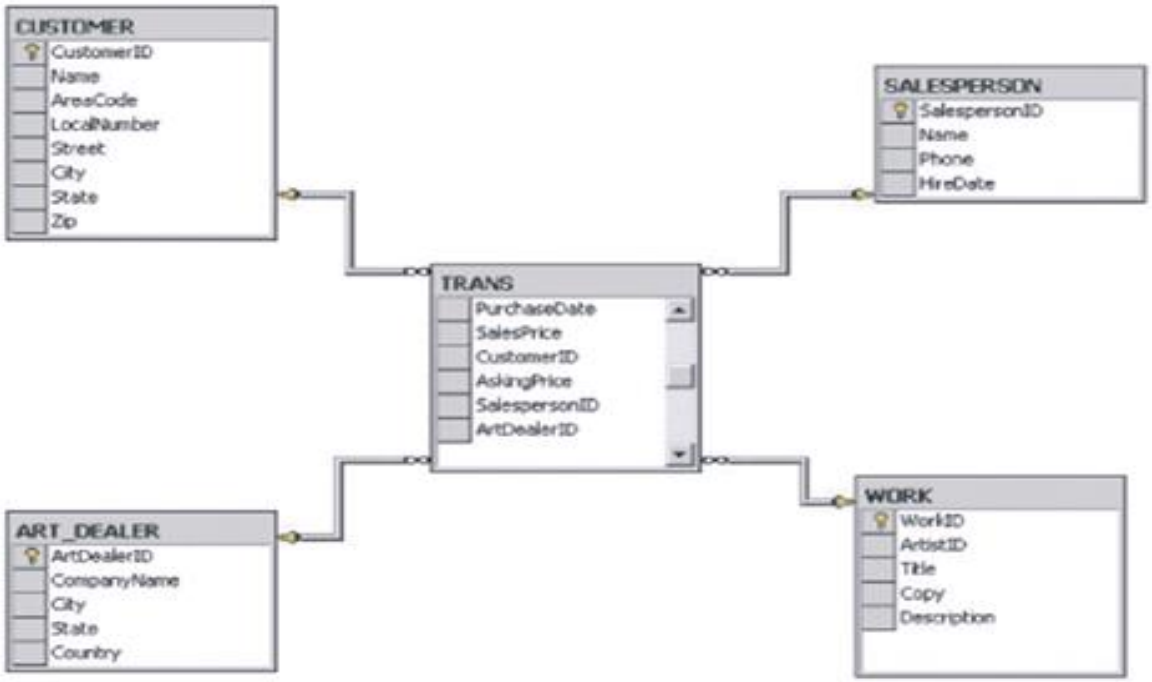
- The previous example only had one relational table
- Most databases have multiple tables with primary/foreign key relationships
- Often the dimensions are held as foreign keys in the “cube” table, with relations to other tables (“member data”) with the details about each dimension

OLAP Schema Structures

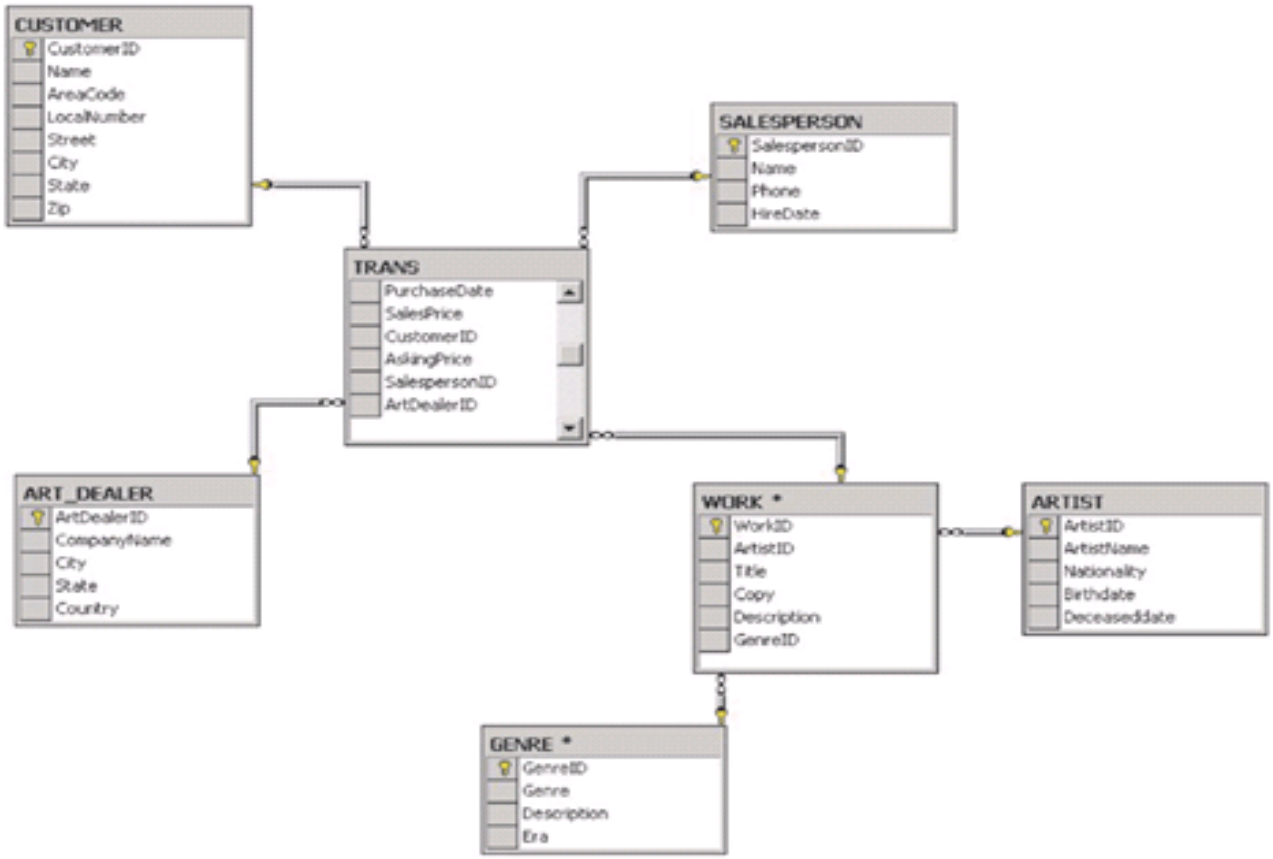
- Star schema: every dimension table is adjacent to the table storing the measure values
  - These tables may or may not be normalized; de-normalization is often done to force a star schema
- Snowflake schema: there can be multilevel, normalized tables

- In general, the star schema requires greater storage, but it is faster to process than the snowflake schema

Example: Star Schema



Example: Snowflake Schema



OLAP Technology/Storage Alternatives

- Three different means for storing OLAP data
- ROLAP (relational OLAP): relational DBMS with extensions is used to meet OLAP requirements
- MOLAP (multidimensional OLAP): a specialized multidimensional processor is used to produce acceptable OLAP performance
- HOLAP (hybrid OLAP): both relational DBMS products and specialized OLAP engines have a role and can be used to advantage

ROLAP vs MOLAP

Characteristic	ROLAP	MOLAP
Schema	Uses star schema Additional dimensions can be added dynamically	Uses data cubes Multidimensional arrays, row stores, column stores Additional dimensions require re-creation of the data cube
Database size	Medium to large	Large
Architecture	Client/server Standards-based	Client/server Open or proprietary, depending on vendor
Access	Supports ad hoc requests Unlimited dimensions	Limited to predefined dimensions Proprietary access languages
Speed	Good with small data sets; average for medium-sized to large data sets	Faster for large data sets with predefined dimensions

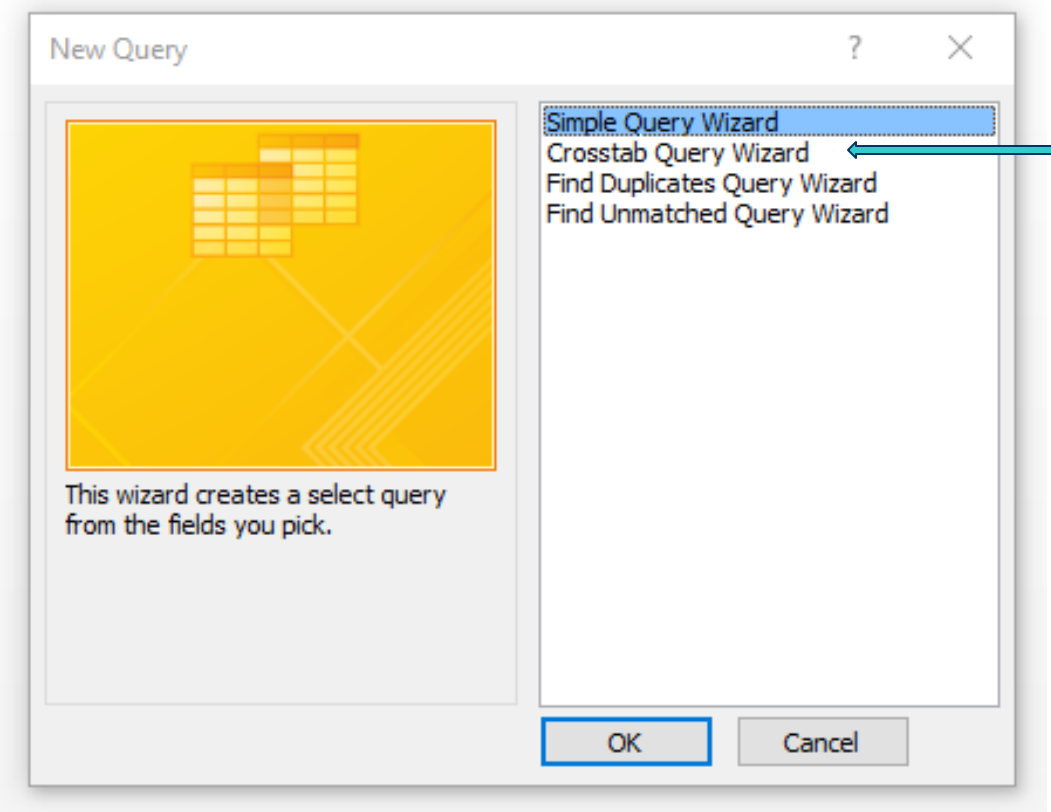
Vendor OLAP Support

- All the major database vendors now provide some type of OLAP support:
  - IBM
  - ORACLE
  - Microsoft (Access and SQL/Server)
    - Newer versions of Access have crosstab, but not full Pivot tables
  - MySQL (open source extensions)
  - Many specialized OLAP and data warehouse vendors

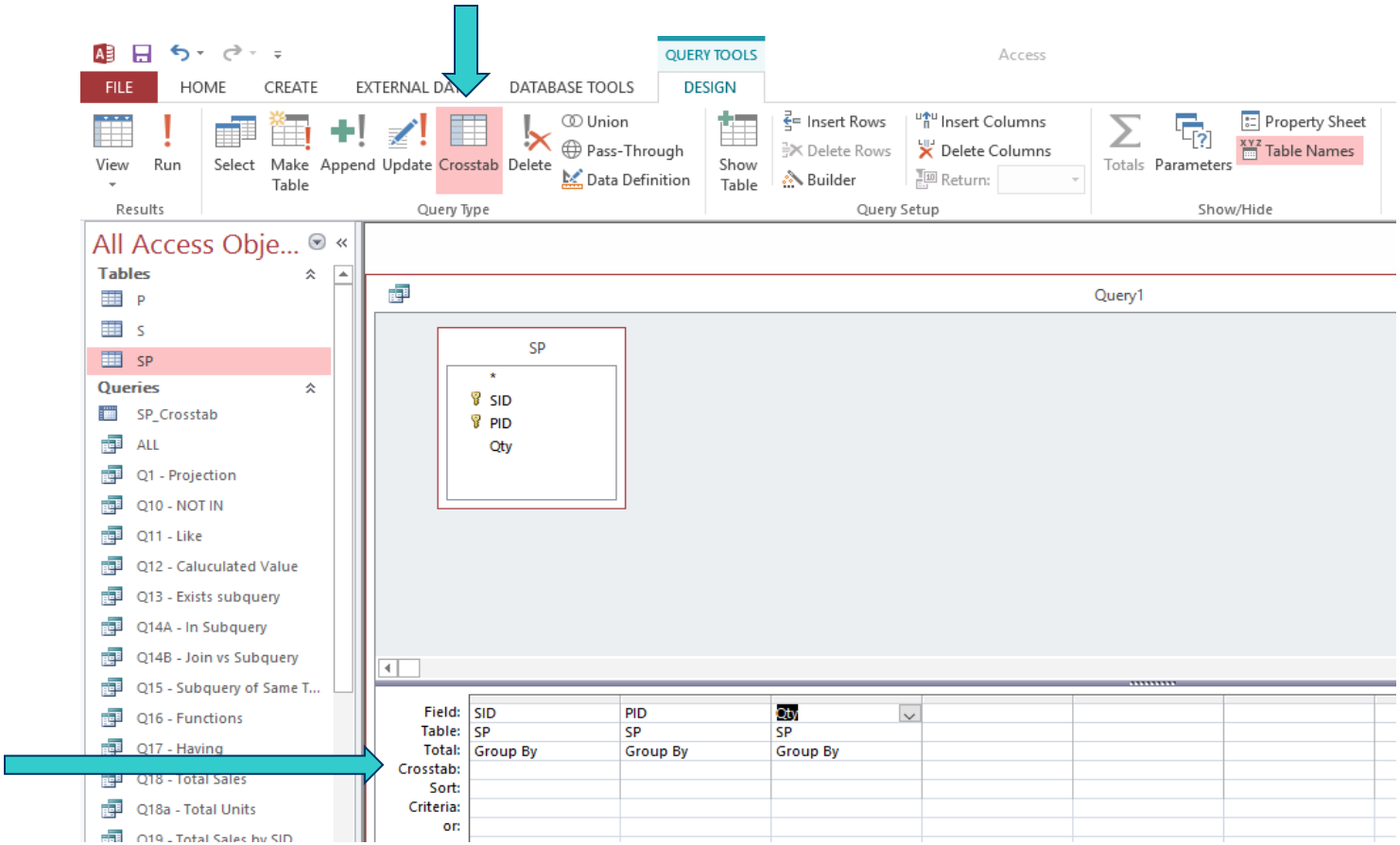
ACCESS CROSSTAB

- Access has a crosstab feature which is like a static pivot table (no dynamic drag and drop)
- These can be created in query design or in the query wizard
- The wizard lets you select a table or query the cross tab is based on
- You then choose at least one field for the Row Heading, another for the Column Heading and a third for the data analysis
- When complete, the cross tab query groups the rows and columns data in a table layout showing one row for each data value from the row heading, one column for each data value for the column heading and the calculations for each row and column intersect
- If you want to change the cross tab design you need to edit the query in design view, and select, for example, different fields for the row and column headings, or a different calculation for the intersections

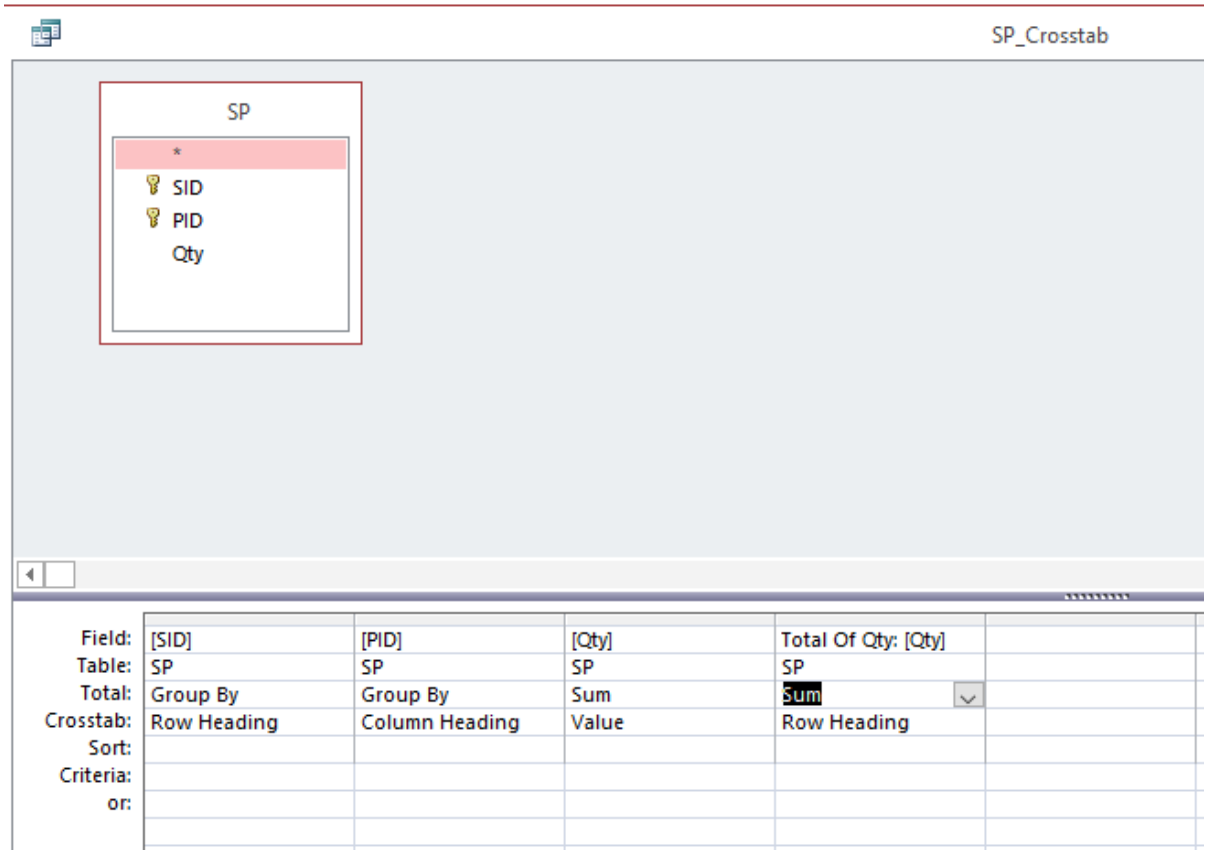
CROSSTAB in Access Query Wizard



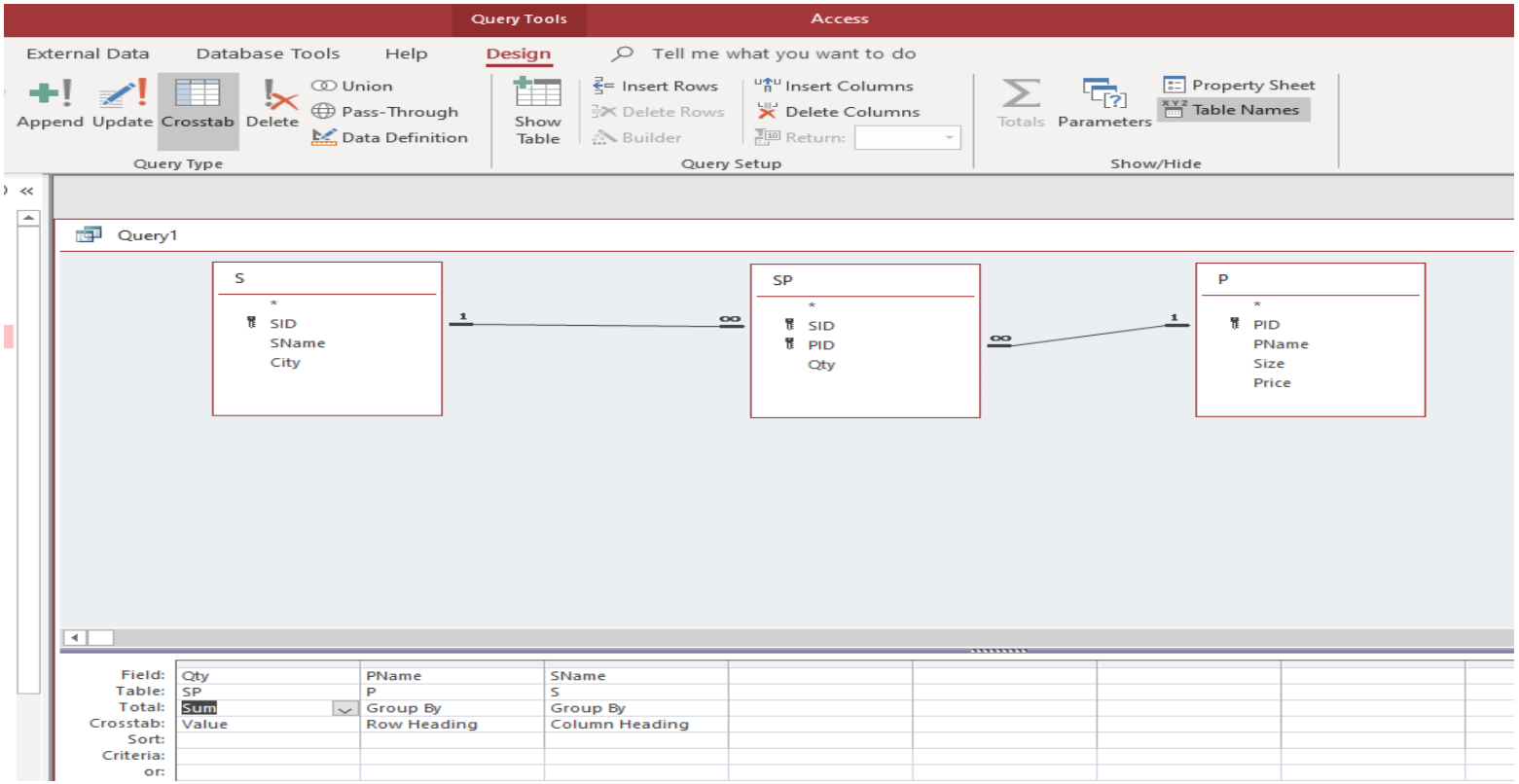
CROSSTAB in Access QBE



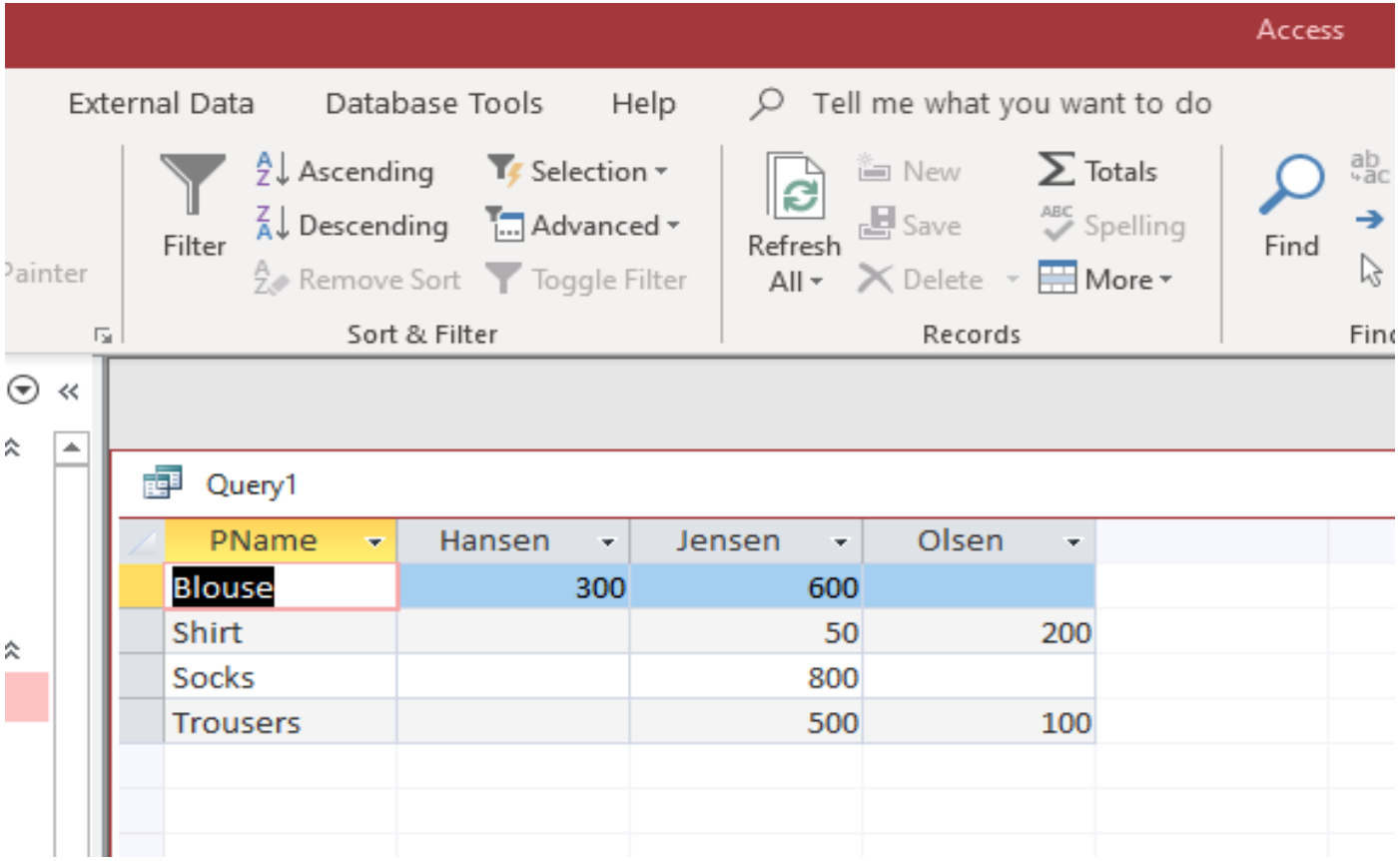
CROSSTAB in Access QBE (con't)



Access Crosstab Using Joined Tables

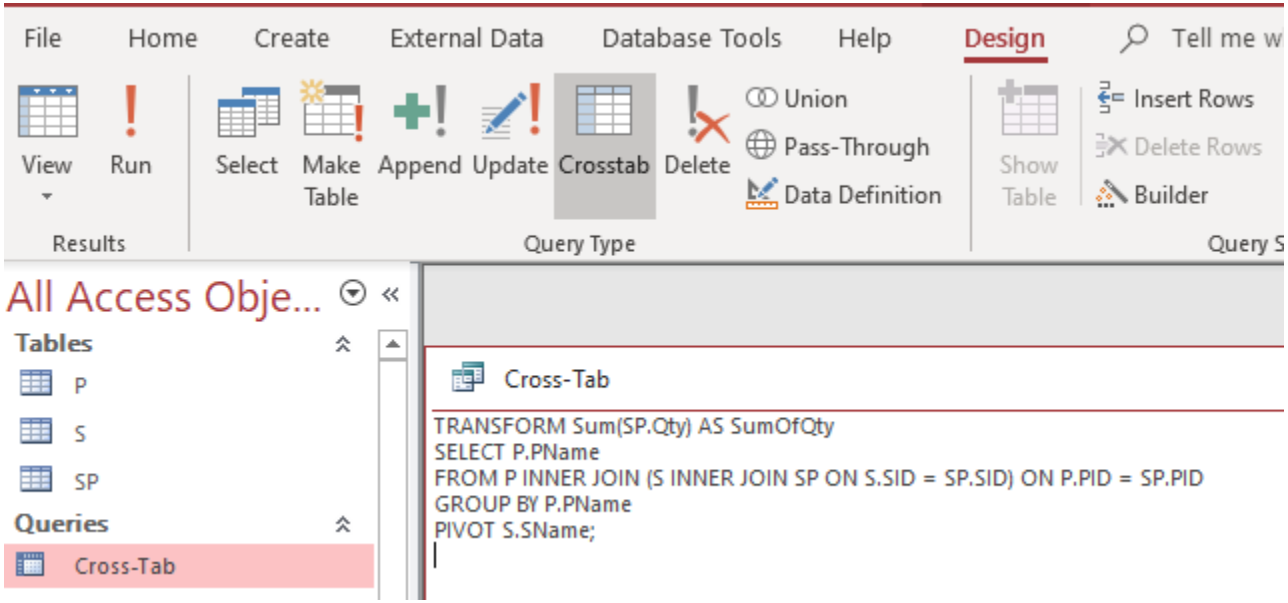


Access Crosstab Using Joined Tables (con't)



Access Crosstab Using Joined Tables (con't)

Switching to SQL view:



SQL TRANSFORM/PIVOT in Access

- Crosstab queries can also be created in SQL
- When you summarize data using a crosstab query, you select values from fields or expressions as column headings so you can view data in a more compact format than with a select query
- TRANSFORM is optional but when included is the first statement in an SQL string
- It precedes a SELECT statement that specifies the fields used as row headings and a GROUP BY clause that specifies row grouping
- Optionally, you can include other clauses, such as WHERE, that specify additional selection or sorting criteria; you can also use subqueries as predicates — specifically, those in the WHERE clause — in a crosstab query

TRANSFORM/PIVOT in Access (con't)

- The values returned in *pivotfield* are used as column headings in the query's result set
- For example, pivoting the sales figures on the month of the sale in a crosstab query would create 12 columns
- You can restrict *pivotfield* to create headings from fixed values (*value1*, *value2*) listed in the optional IN clause
  - You can also include fixed values for which no data exists to create additional columns

TRANSFORM/PIVOT in Access (con't)

TRANSFORM Statement

Creates a crosstab query.

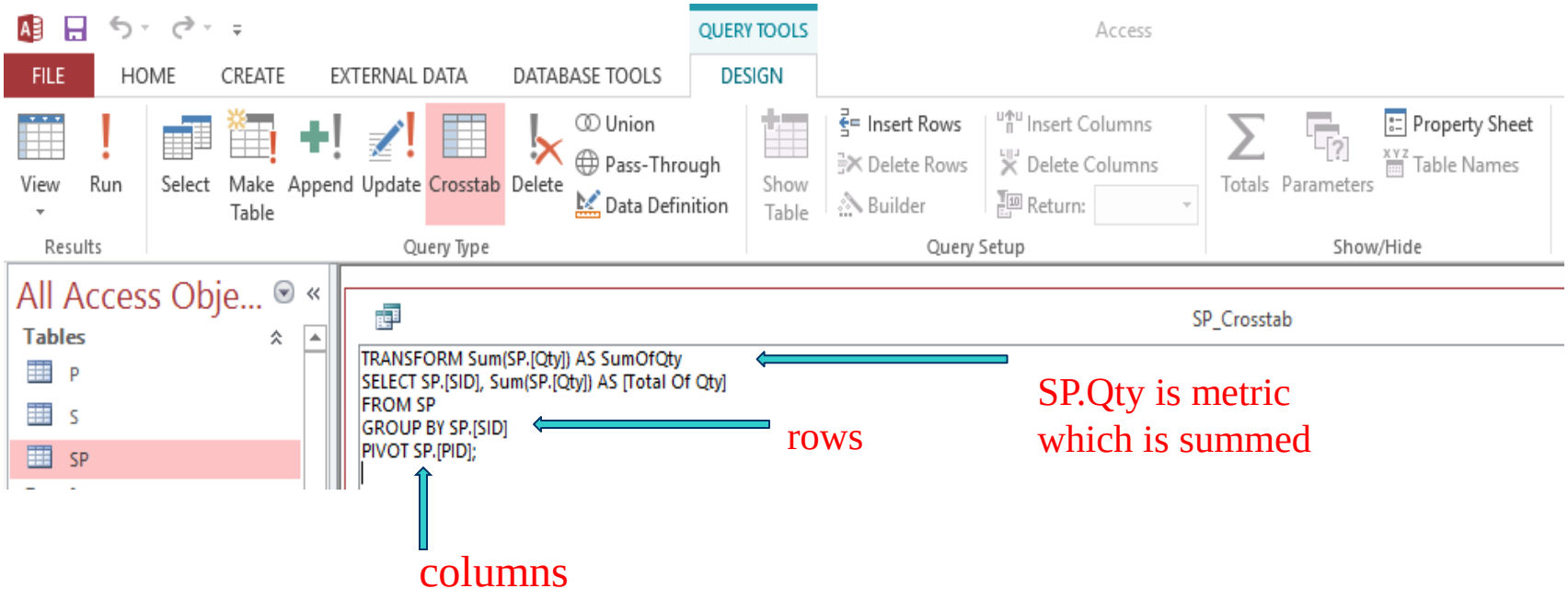
Syntax

TRANSFORM *aggfunction*
selectstatement
PIVOT *pivotfield* [IN (*value1*[, *value2*[, ...]])]

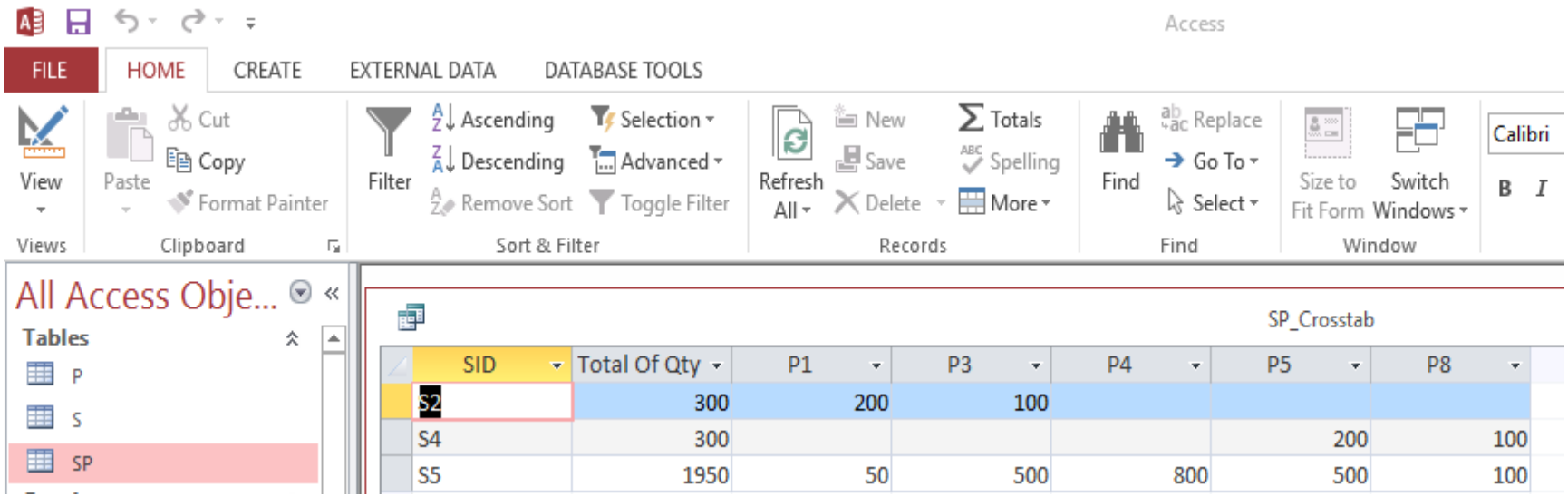
The TRANSFORM statement has these parts:

Part	Description
aggfunction	An SQL aggregate function that operates on the selected data.
selectstatement	A SELECT statement.
pivotfield	The field or expression you want to use to create column headings in the query's result set.
value1, *value2*	Fixed values used to create column headings.

TRANSFORM/PIVOT in Access (con't)



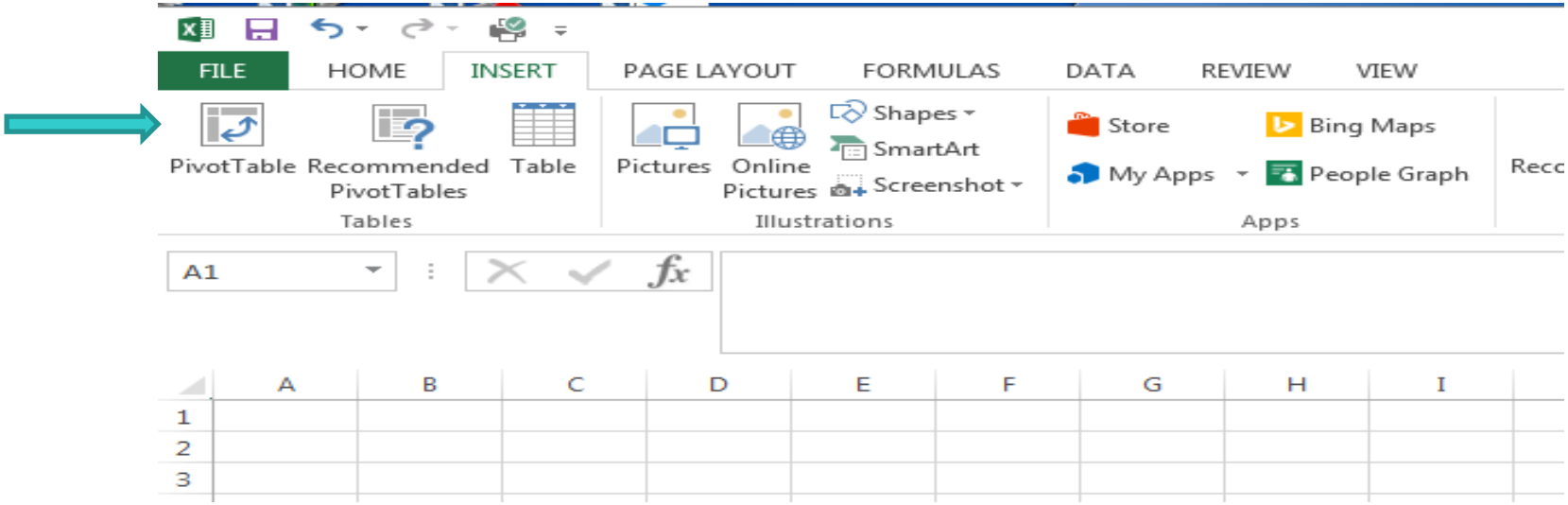
TRANSFORM/PIVOT in Access (con't)



SID	Total Of Qty	P1	P3	P4	P5	P8
S2	300	200	100			
S4	300				200	100
S5	1950	50	500	800	500	100

Excel Pivot Table

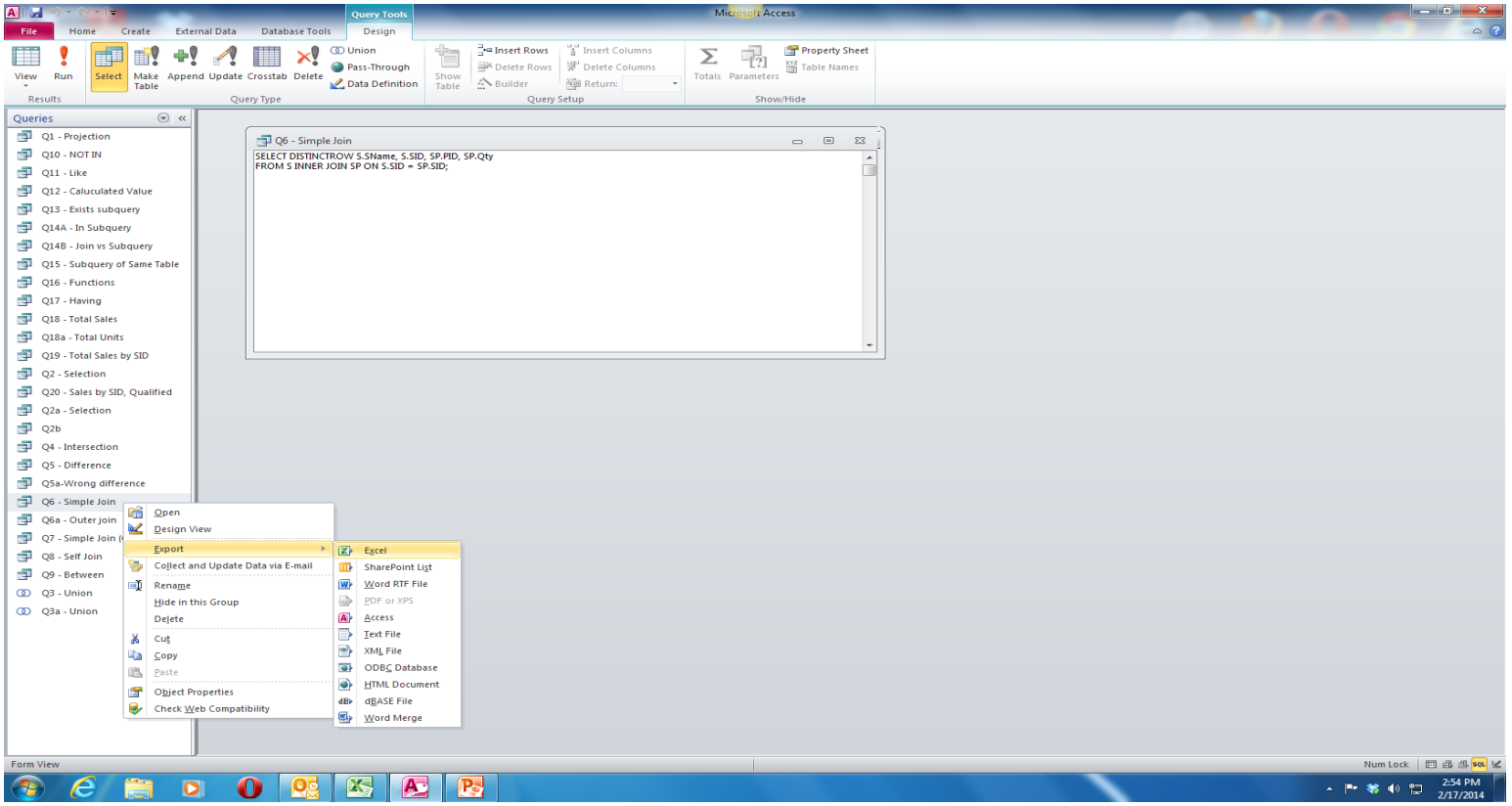
- Excel provides more comprehensive support for pivot tables, so one can export from Access (or another RDBMS) into Excel



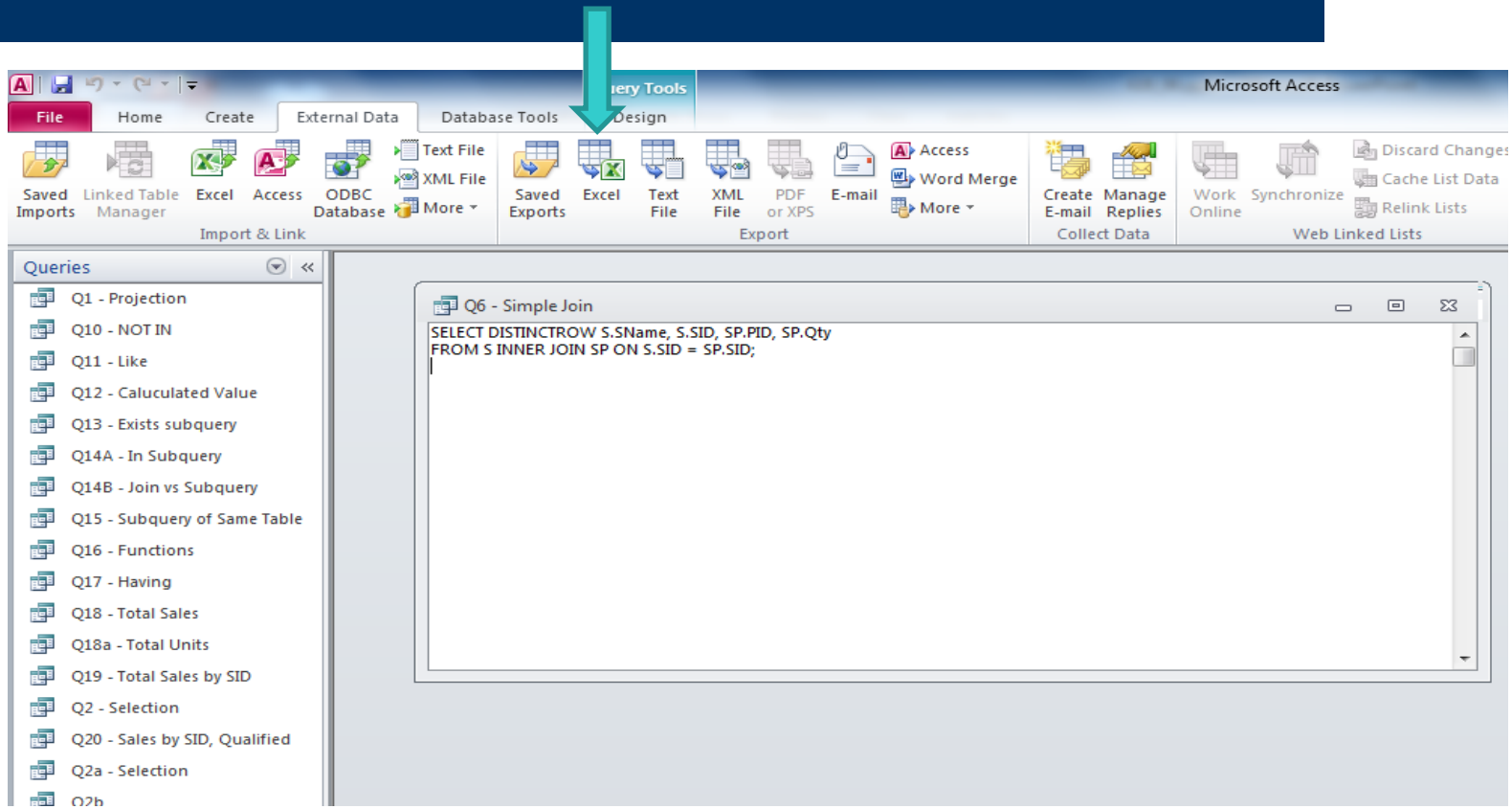
Database → Spreadsheet

- Data is often exported from data warehouses or databases into Excel or another analysis tool
- Objects (such as queries) can be defined as the source of an export to Excel or other tool

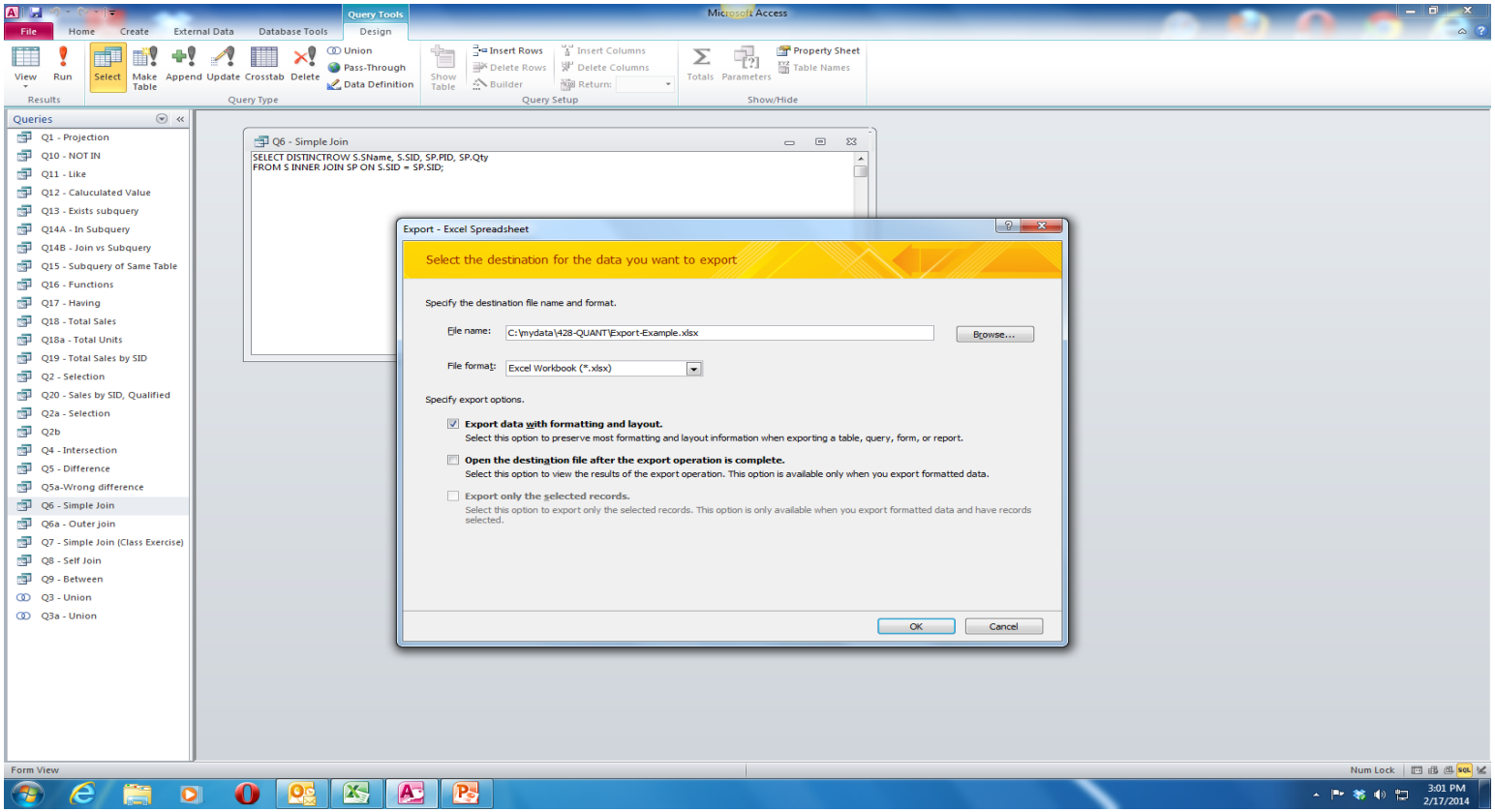
Export Database Object to Excel [right click object]



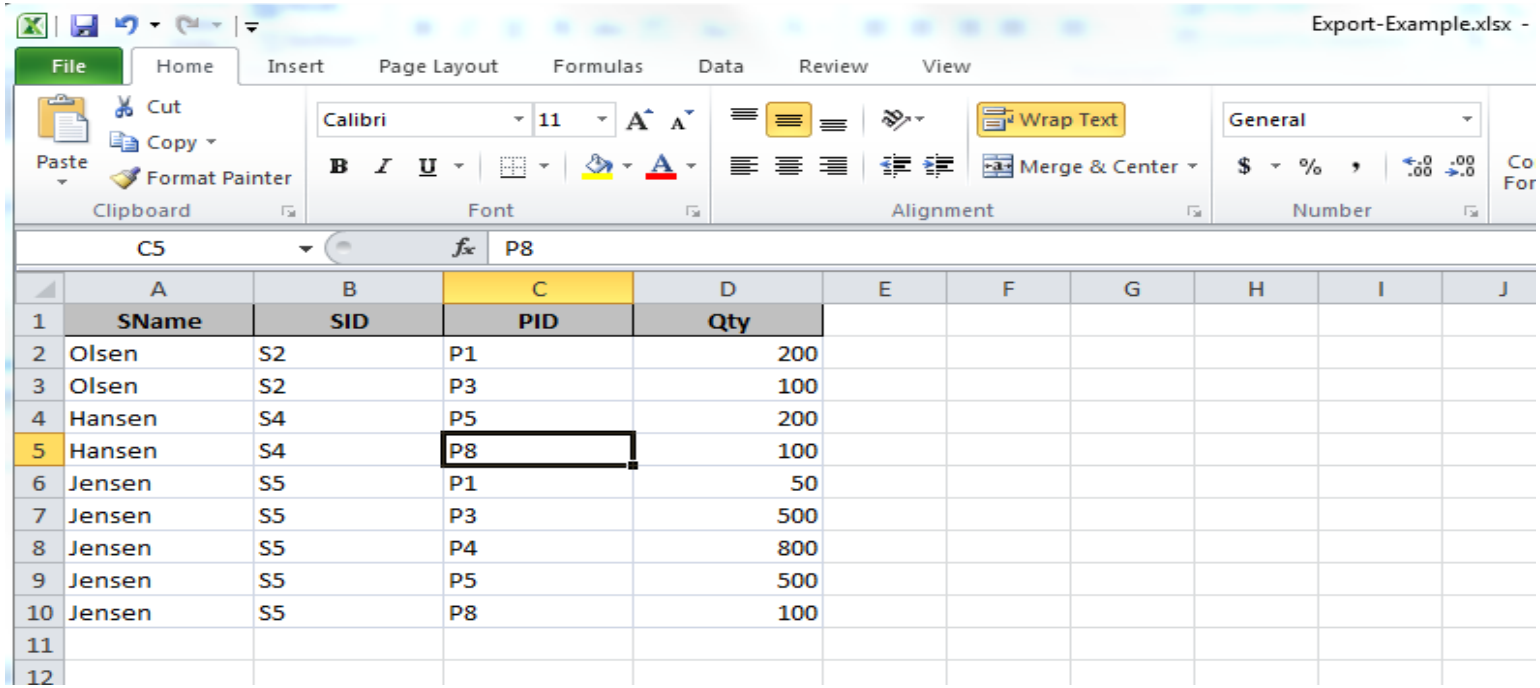
Or from ribbon:
External Data Tab, Export Group, Excel



Export Database Object to Excel (con't)



Export Database Object to Excel (con't)



	A	B	C	D	E	F	G	H	I	J
1	SName	SID	PID	Qty						
2	Olsen	S2	P1	200						
3	Olsen	S2	P3	100						
4	Hansen	S4	P5	200						
5	Hansen	S4	P8	100						
6	Jensen	S5	P1	50						
7	Jensen	S5	P3	500						
8	Jensen	S5	P4	800						
9	Jensen	S5	P5	500						
10	Jensen	S5	P8	100						
11										
12										

Export Database Object to Excel (con't)

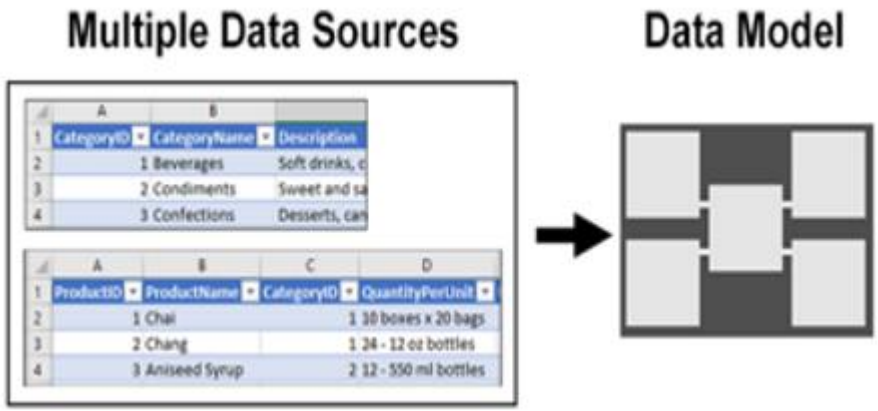
EXPORT	SOURCE OBJECT	FIELDS AND RECORDS	FORMATTING
Without formatting	Table or query NOTE Forms and reports cannot be exported without their formatting.	All fields and records in the underlying object are exported.	The **Format** property settings are ignored during the operation. For lookup fields, only the lookup ID values are exported. For hyperlink fields, the contents are exported as a text column that displays the links in the format **displaytext#address#**.
With formatting	Table, query, form, or report	Only fields and records that are displayed in the current view or object are exported. Filtered records, hidden columns in a datasheet, and fields not displayed on a form or report are not exported.	The wizard respects the **Format** property settings. For lookup fields, the lookup values are exported. For hyperlink fields, the values are exported as hyperlinks. For rich text fields, the text is exported but the formatting is not.

Export Database Object to Excel (con't)

IF THE DESTINATION WORKBOOK	AND THE SOURCE OBJECT IS	AND YOU WANT TO EXPORT	THEN
Does not exist	A table, query, form, or report	The data, with or without the formatting	The workbook is created during the export operation.
Already exists	A table or query	The data, but not the formatting	The workbook is not overwritten. A new worksheet is added to the workbook, and is given the name of the object from which the data is being exported. If a worksheet having that name already exists in the workbook, Access prompts you to either replace the contents of the corresponding worksheet or specify another name for the new sheet.
Already exists	A table, query, form, or report	The data, including the formatting	The workbook is overwritten by the exported data. All existing worksheets are removed, and a new worksheet having the same name as the exported object is created. The data in the Excel worksheet inherits the format settings of the source object.

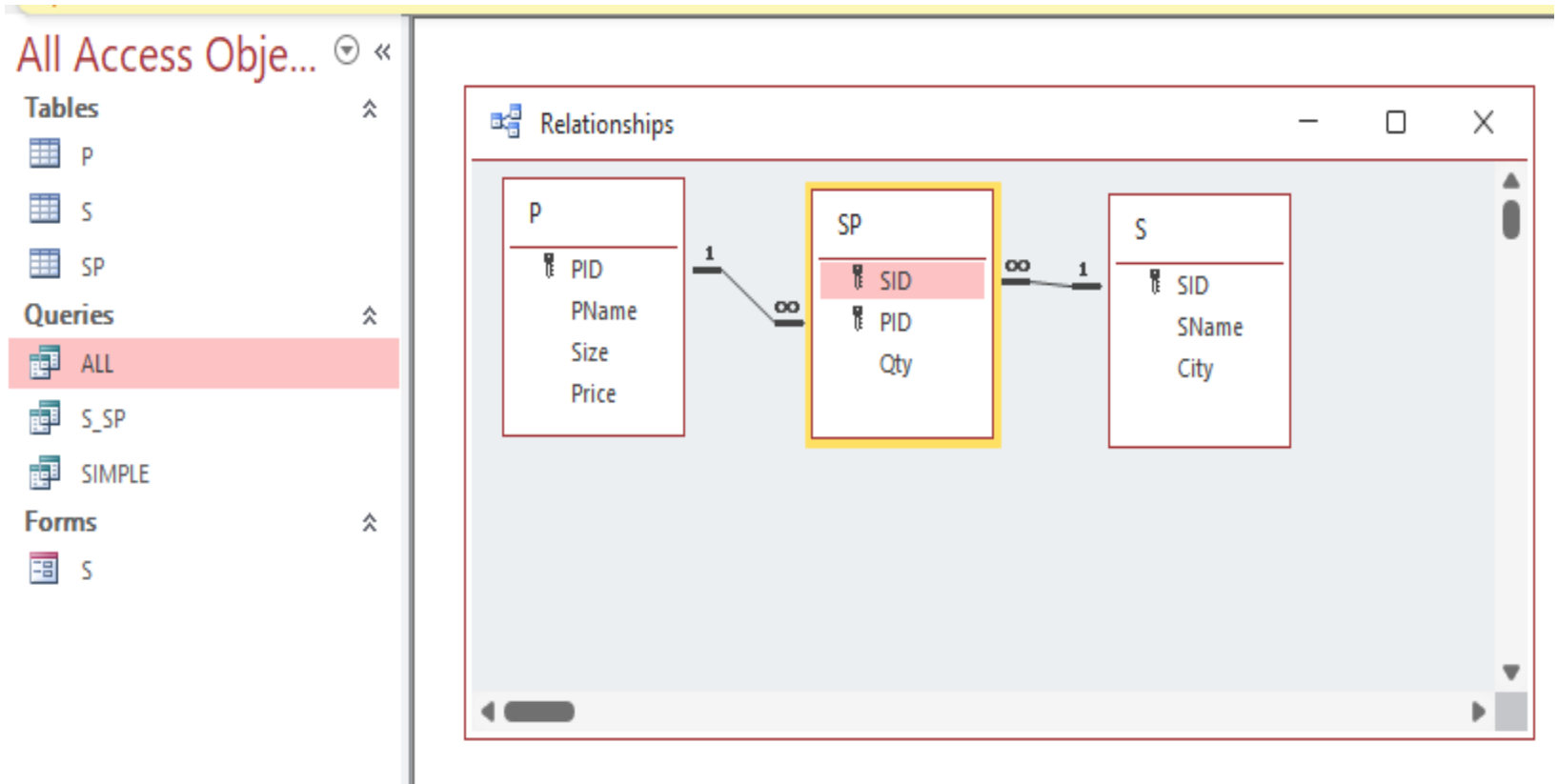
Excel OLAP (Data Models)

- Latest versions of Excel have “Get & Transform” which can import directly from relational databases such as Access, SQLServer, MySQL, Oracle, etc.
- Import can be to a table, data model (Power PivotTable), or PivotChart
- Tables and relationships can be imported



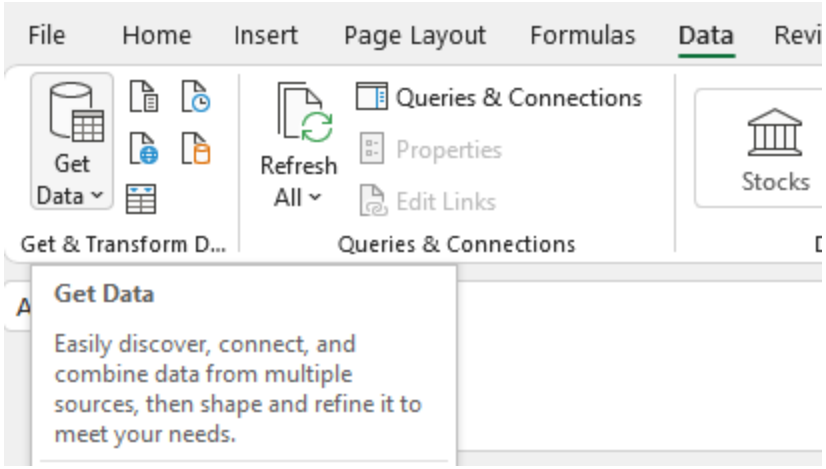
Data Models (con't)

- Example Access database:



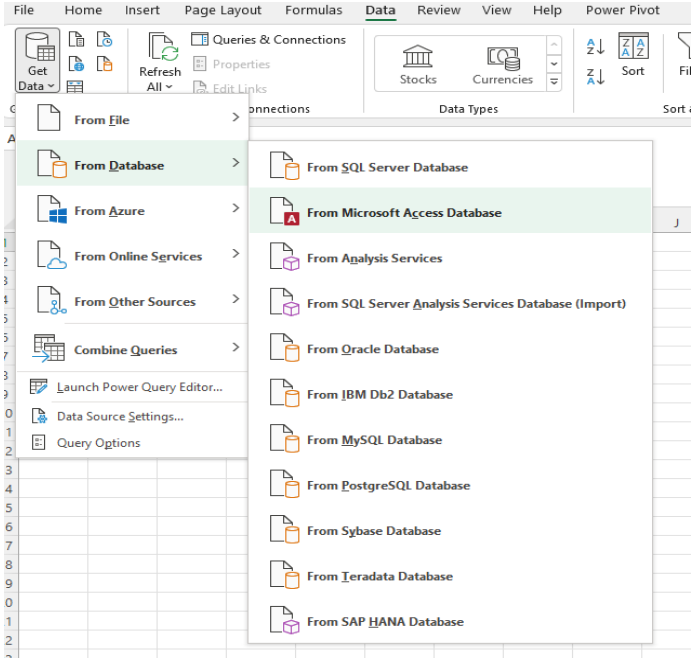
Data Models (con't)

- Open a new Excel workbook, then:
  - Data → Get & Transform → Get Data → From Database → from Microsoft Access Database



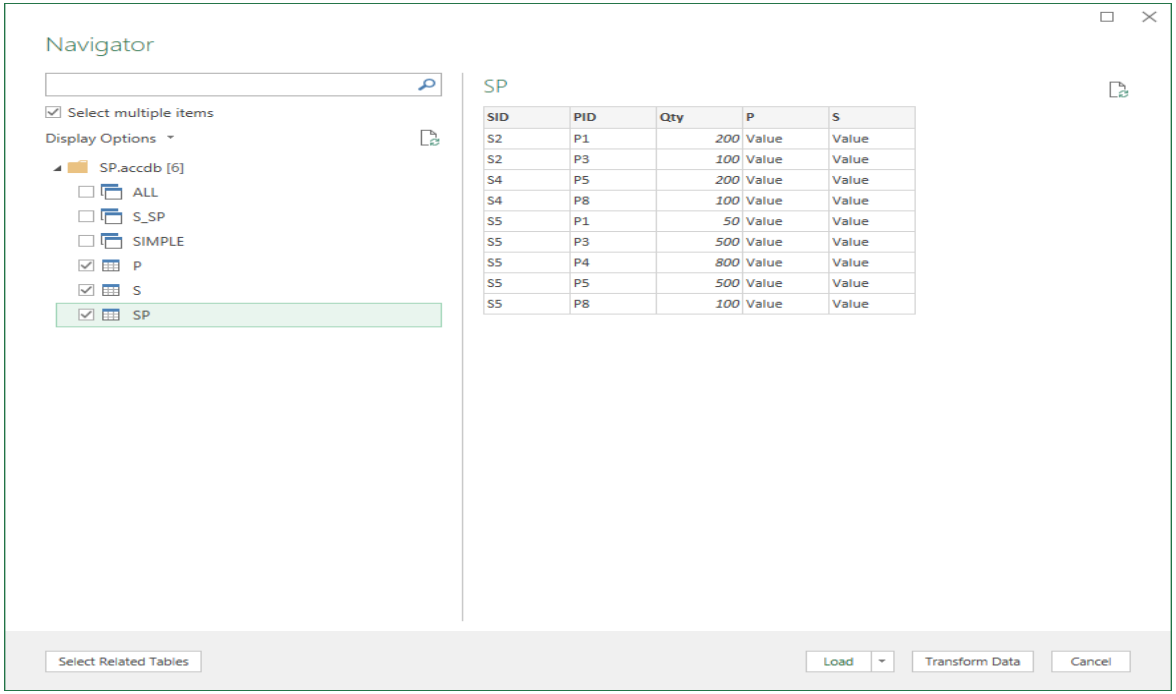
Data Models (con’t)

- From Microsoft Access Database then chose database file in Windows file selection window



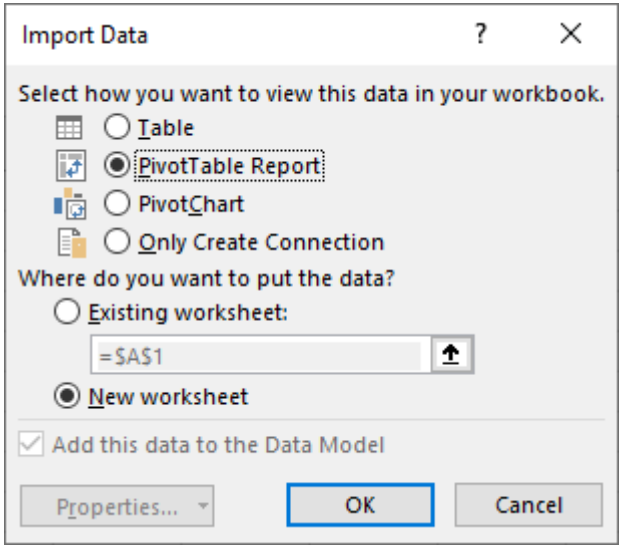
Data Models (con't)

- Navigator window opens – select multiple items – click load button



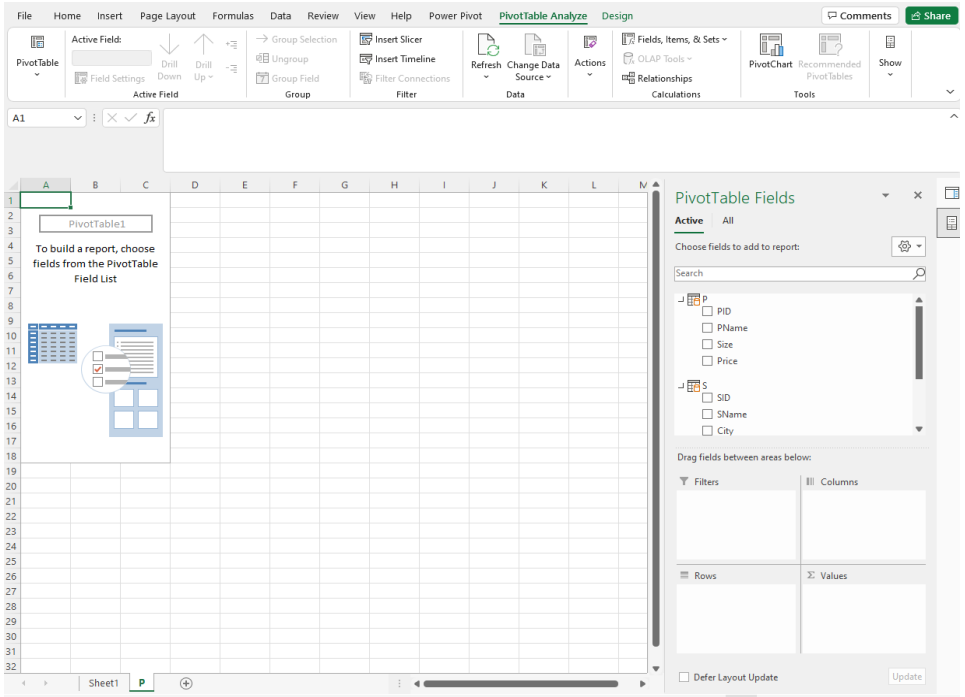
Data Models (con't)

- Click Load → Load To …
  - Click Pivot Table Report (Add this to the Data Model is automatically checked)
  - Click OK



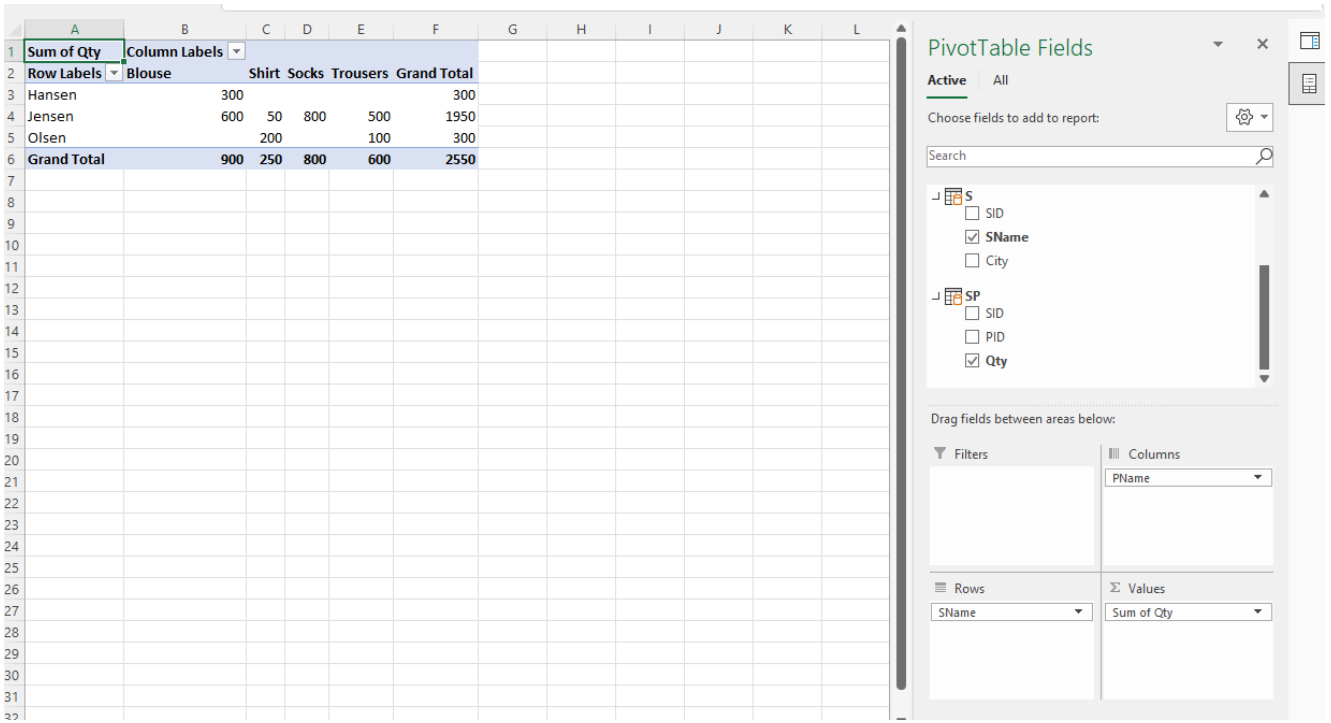
Data Models (con’t)

- An empty OLPA pivot table is created, and you can now design your pivot table



Data Models (con’t)

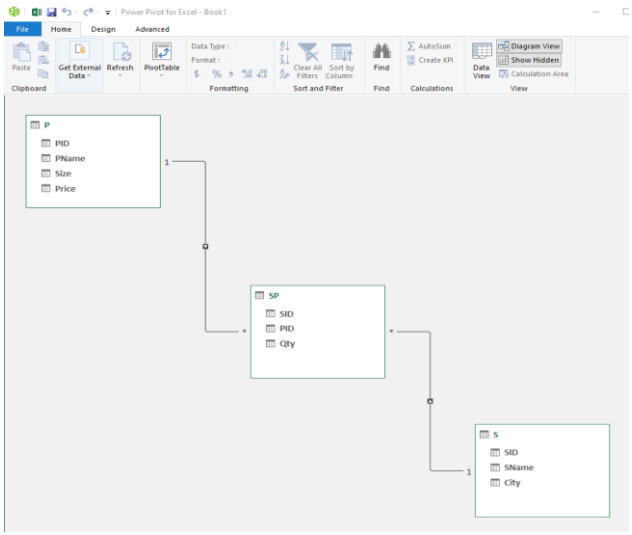
- Qty by Salesperson name and Product name



Sum of Qty	Column Labels				
Row Labels	Blouse	Shirt	Socks	Trousers	Grand Total
Hansen	300				300
Jensen	600	50	800	500	1950
Olsen		200		100	300
Grand Total	900	250	800	600	2550

Data Models (con't)

- One can see the imported Data Model:
  - Power Pivot → Data Model → Manage
  - The Power Pivot window opens
  - Home → View → Diagram View



PIVOT in SQL Server (known PID's)

```
SELECT *
FROM
(
   SELECT SID, PID, QTY
   FROM SP
) SRC
PIVOT
(
   SUM(QTY)
   FOR PID IN ([1],[2],[3],[4],[5],[6],[7],[8])
)PIV;
```

PIVOT in SQL Server
(unknown PID's – need dynamic query)

```
DECLARE @COLS AS NVARCHAR(MAX),
   @query AS NVARCHAR(MAX)
SELECT @COLS = (SELECT DISTINCT PID FROM SP)
SET @query = 'SELECT *
FROM
(
   SELECT SID, PID, QTY
   FROM SP
) SRC
PIVOT
(
   SUM(QTY)
   FOR PID IN (@COLS)
)PIV'
execute(@query)
```

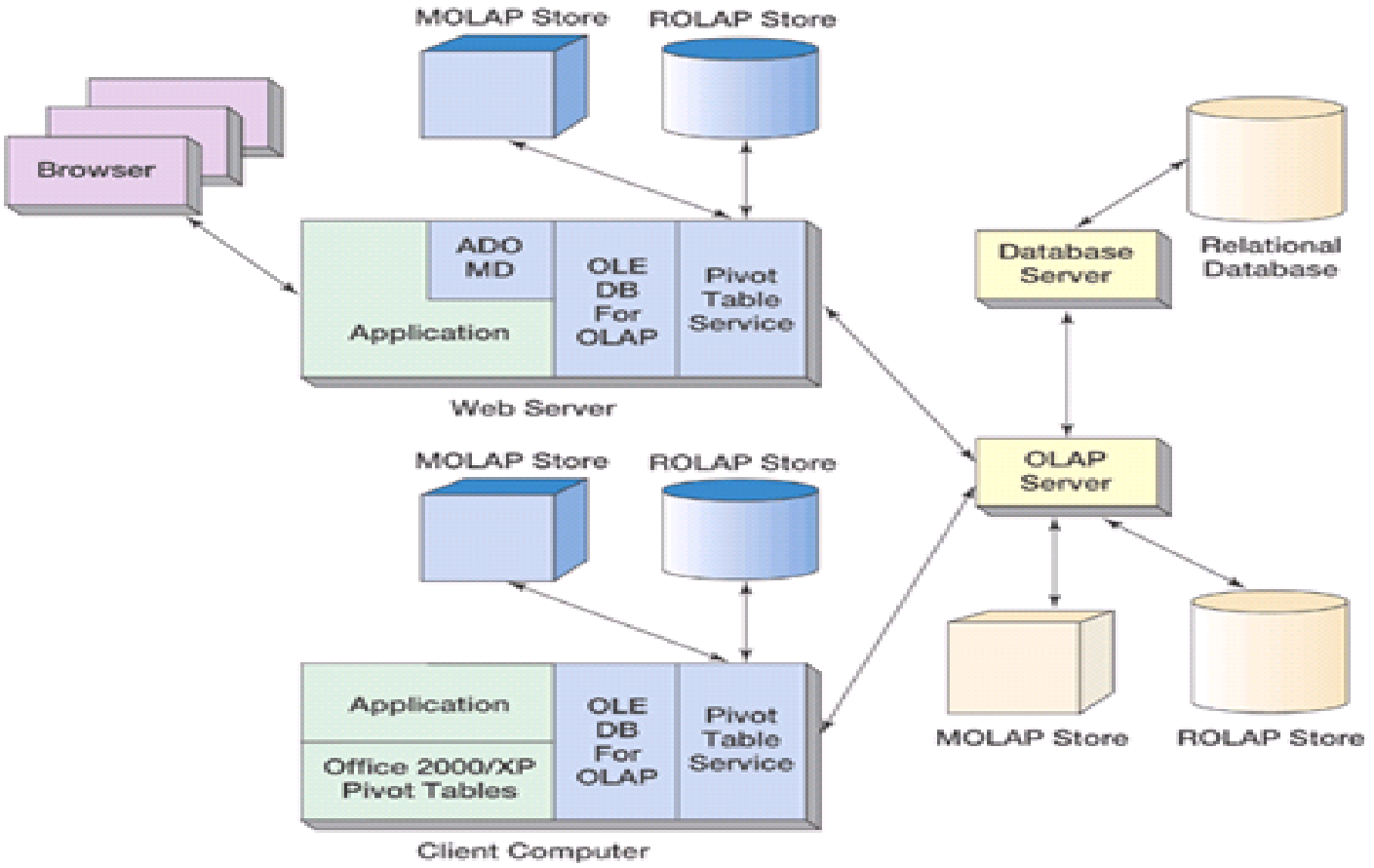
Other SQL Analytics Functions

- The ROLLUP extension to GROUP BY
  - Used with GROUP BY clause to generate aggregates by different dimensions
  - Enables subtotal for each column listed except for the last one, which gets a grand total
    - SELECT column1 [,column2, …], aggregate function (expression)
    - FROM table1[,table2,…]
    - [WHERE …]
    - GROUP BY column1[,column2, …] WITH ROLLUP
    - [HAVING condition]
    - [ORDER by column1[,column2, …]

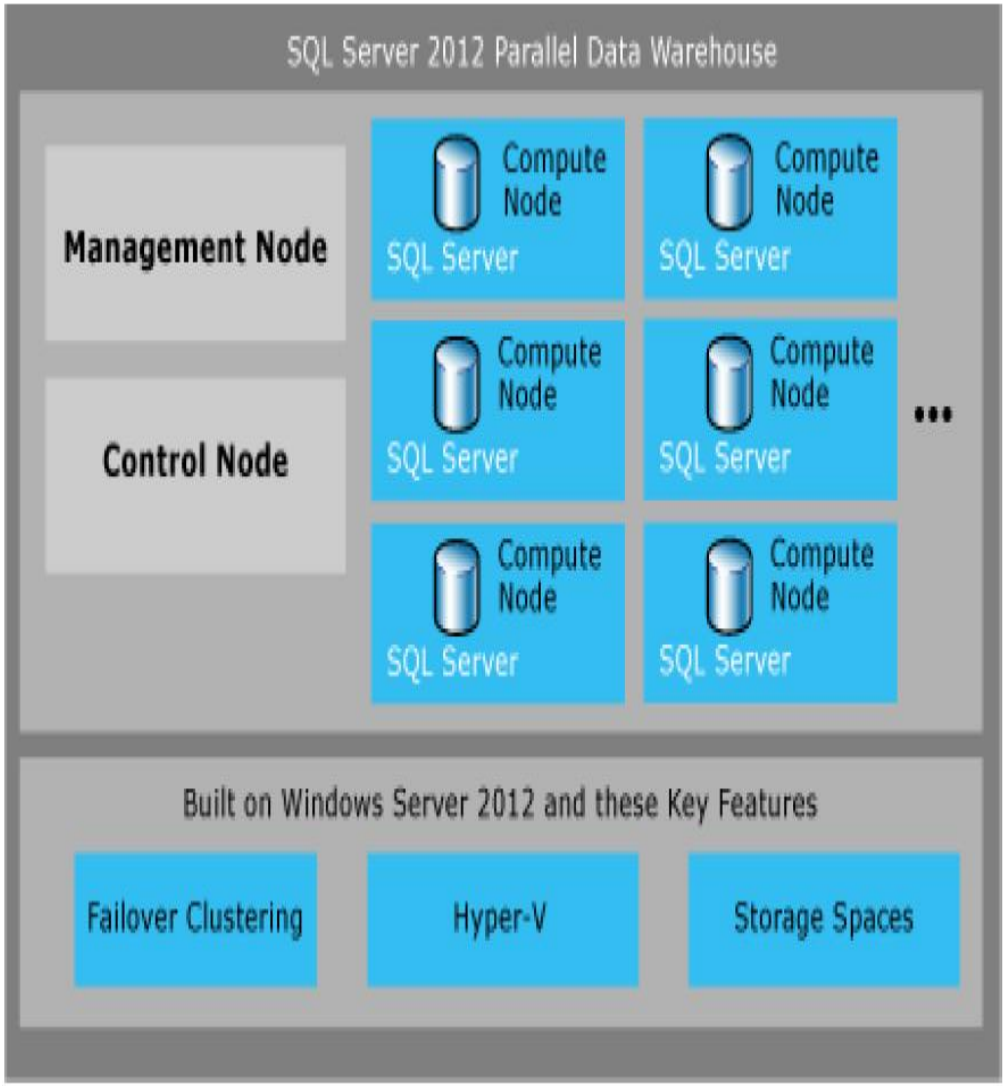
Other SQL Analytics Functions (con't)

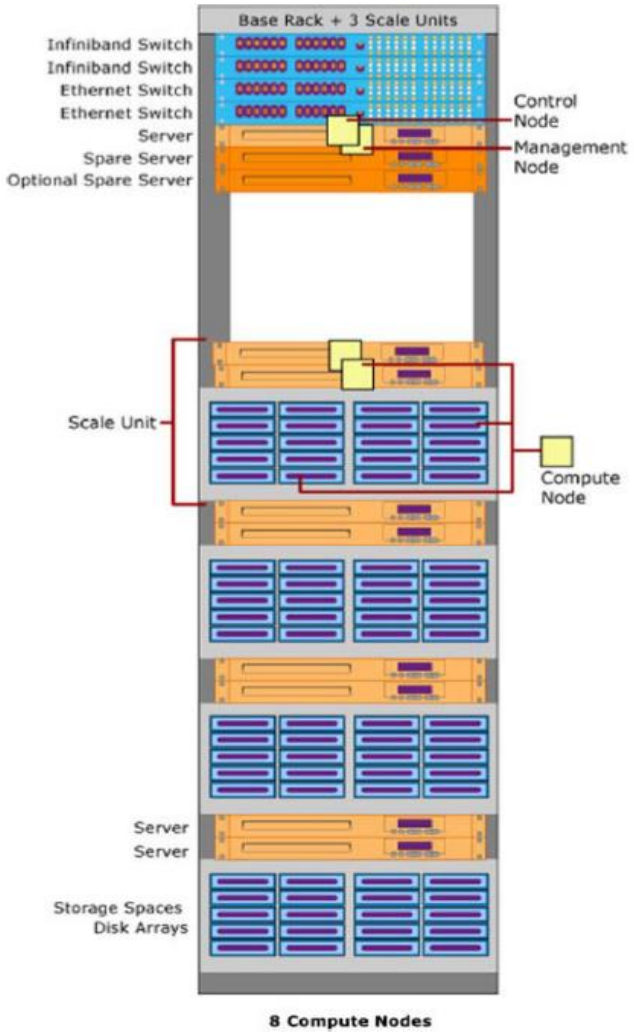
- The CUBE extension to GROUP BY
  - Used with GROUP BY clause to generate aggregates by the listed columns
  - Enables you to get a subtotal for each column listed in the expression, in addition to a grand total for the last column listed
    - SELECT column1 [,column2, …], aggregate function (expression)
    - FROM table1[,table2,…]
    - [WHERE …]
    - GROUP BY column1[,column2, …] WITH CUBE
    - [HAVING condition]
    - [ORDER by column1[,column2, …]

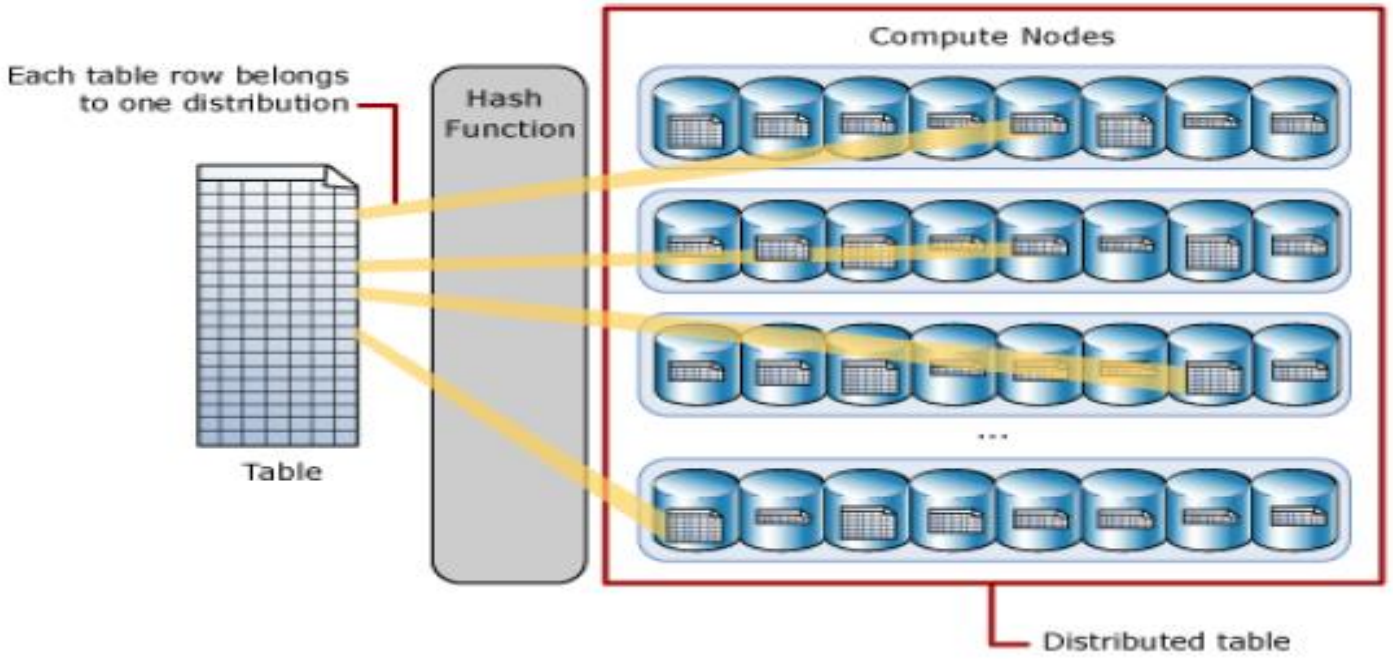
Microsoft OLAP Architecture



SQL Server with Multiple Nodes





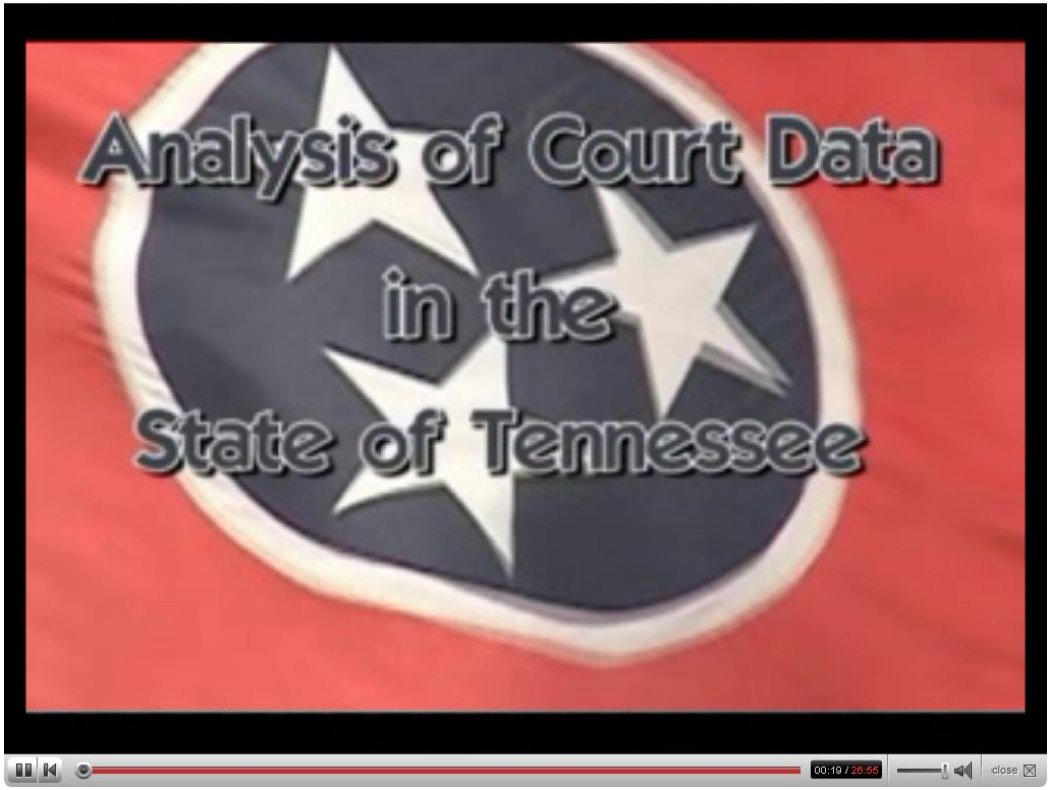


When PDW distributes a fact table, it uses one of the columns as the key for determining the distribution to which the row belongs. A hash function assigns each row to a distribution according to the key value in the distribution column. Every row in a table belongs to one and only one distribution. If you don't choose the best distribution column when create the table, it is easy to re-create the table to use a different distribution column.

PDW doesn't require that all tables get distributed. Small dimension tables are usually replicated to each Compute node. Replicating small tables speeds query processing since the data is always available on each Compute node and there is no need to waste time transferring the data among the nodes in order to satisfy a query.

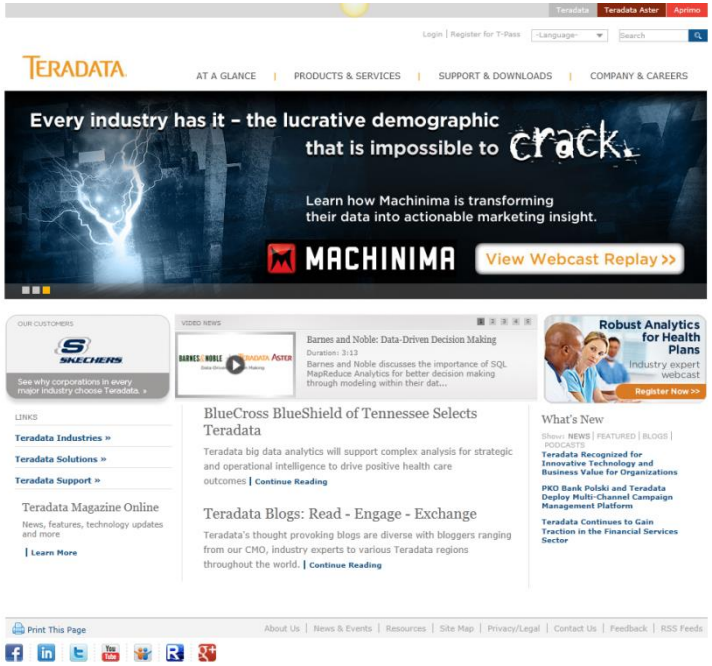
PDW's cost-based query optimizer is the **"secret sauce"** that makes parallel queries run fast and return accurate results. A result of Microsoft's extensive research and development efforts, the query optimizer uses proprietary algorithms to successfully choose a high performing parallel query plan.

OLAP


Analysis of Court Data
in the
State of Tennessee
00:19 / 26:55
close

Teradata

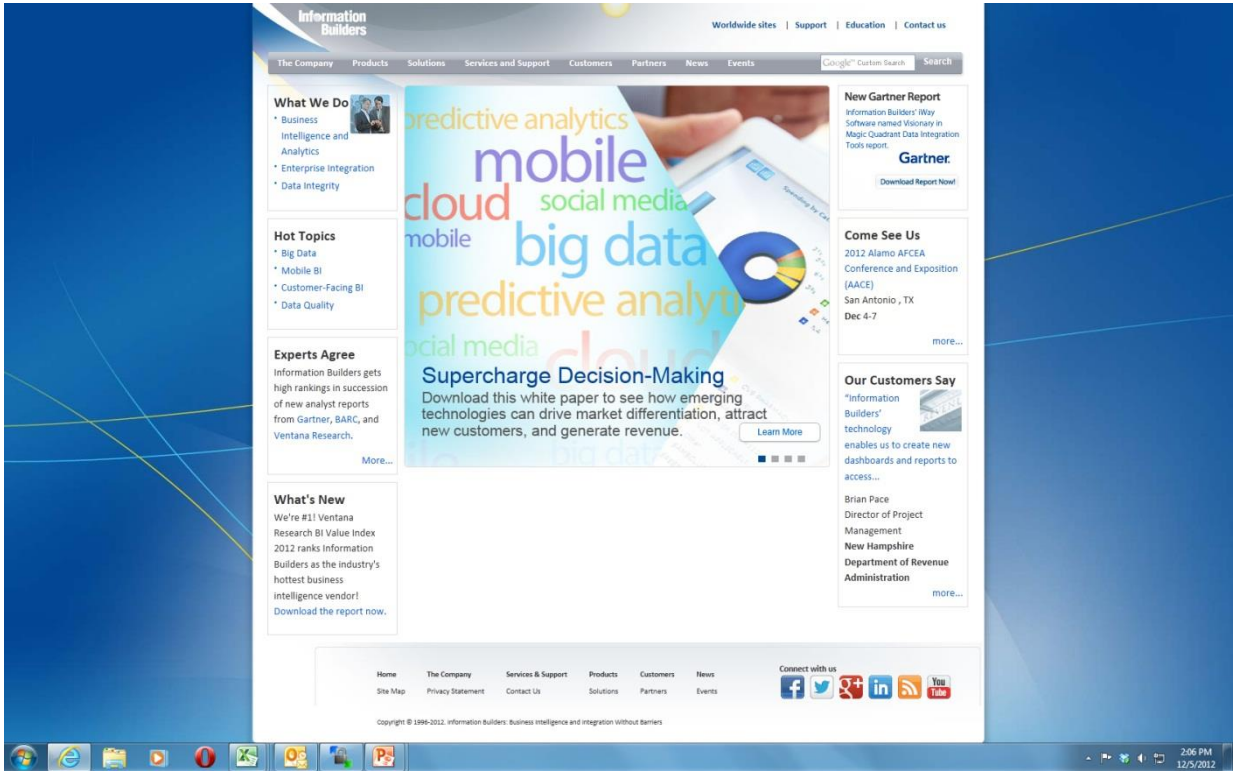
[NYSE: TDC]



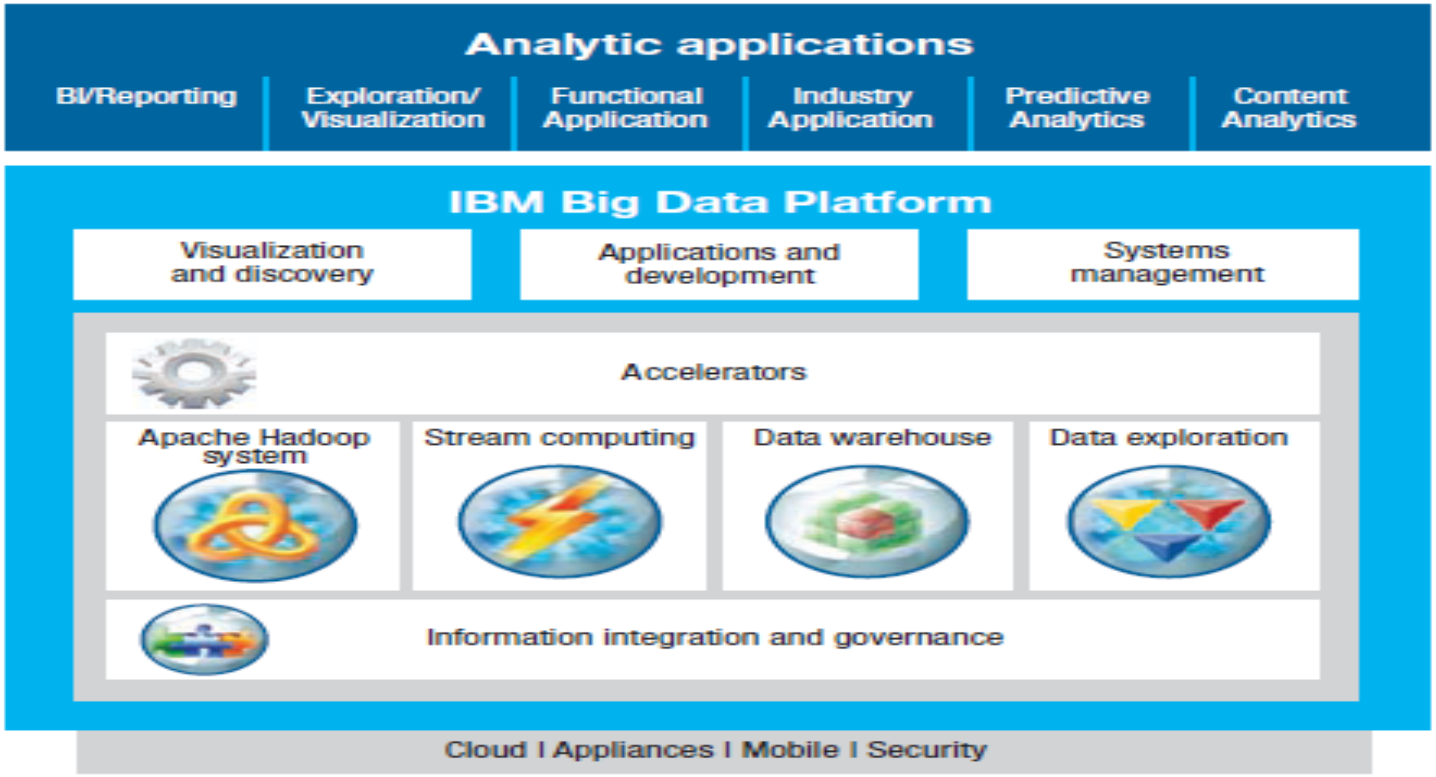
1010 Data


1010 data
Contact Us | Support | Username Password LOGIN
Forgot Your Password?
Business Empowerment | Community | Solutions & Services | Underlying Technology | News | About
THE
WORLD'S DATA
AT YOUR
FINGERTIPS
Learn More >
1 2 3 4 5
"You truly are a whole new approach to data analytics."
CIO, Rite Aid
More Business Insight Through Better (and Easier) Data Analysis
Join the hundreds of blue-chip companies who have found an easy way to get more insight out of more data.
For well over a decade, 1010data has pushed the limits of analytics on large amounts of data, including "Big Data". From routine reporting to advanced analytics, the 1010data system allows businesses like yours to hone their tactics and strategy, while reducing technology overhead, costs and risk.
ANALYST REPORT
NEW Gartner Survey Analysis
1010data earns high scores in Survey: Customers Rate Their BI Platform Vendors, 2012
Gartner Data Warehouse Database Management Systems
1010data Positioned as a Challenger in 2012 DWDM Gartner Magic Quadrant
EVENTS
*IE
December 6 - 7, 2012
NRF
January 13 - 16, 2013
NEWS
11.13.12
NYSE Delivers Analyzed Data To Clients (Forbes)
11.12.12
Security Precautions Prevent Disruption in wake of Hurricane Sandy
10.29.12
IT Spending Slowdown, Emerging Trends Reshuffles Megavendor Deck (ZDNet)
DEMO
Rows 1-16 of 47,304,770,932
RESOURCE LIBRARY
Extensive set of case studies, white papers, data sheets, etc.
2:04 PM
12/5/2012

Information Builders



IBM “Big Data” Platform



Tableau

Tableau is business intelligence software that helps people see and understand their data.

FAST ANALYTICS
Connect and visualize your data in minutes. Tableau is 10 to 100x faster than existing solutions.

BIG DATA, ANY DATA
From spreadsheets to databases to Hadoop to cloud services, explore any data with Tableau.

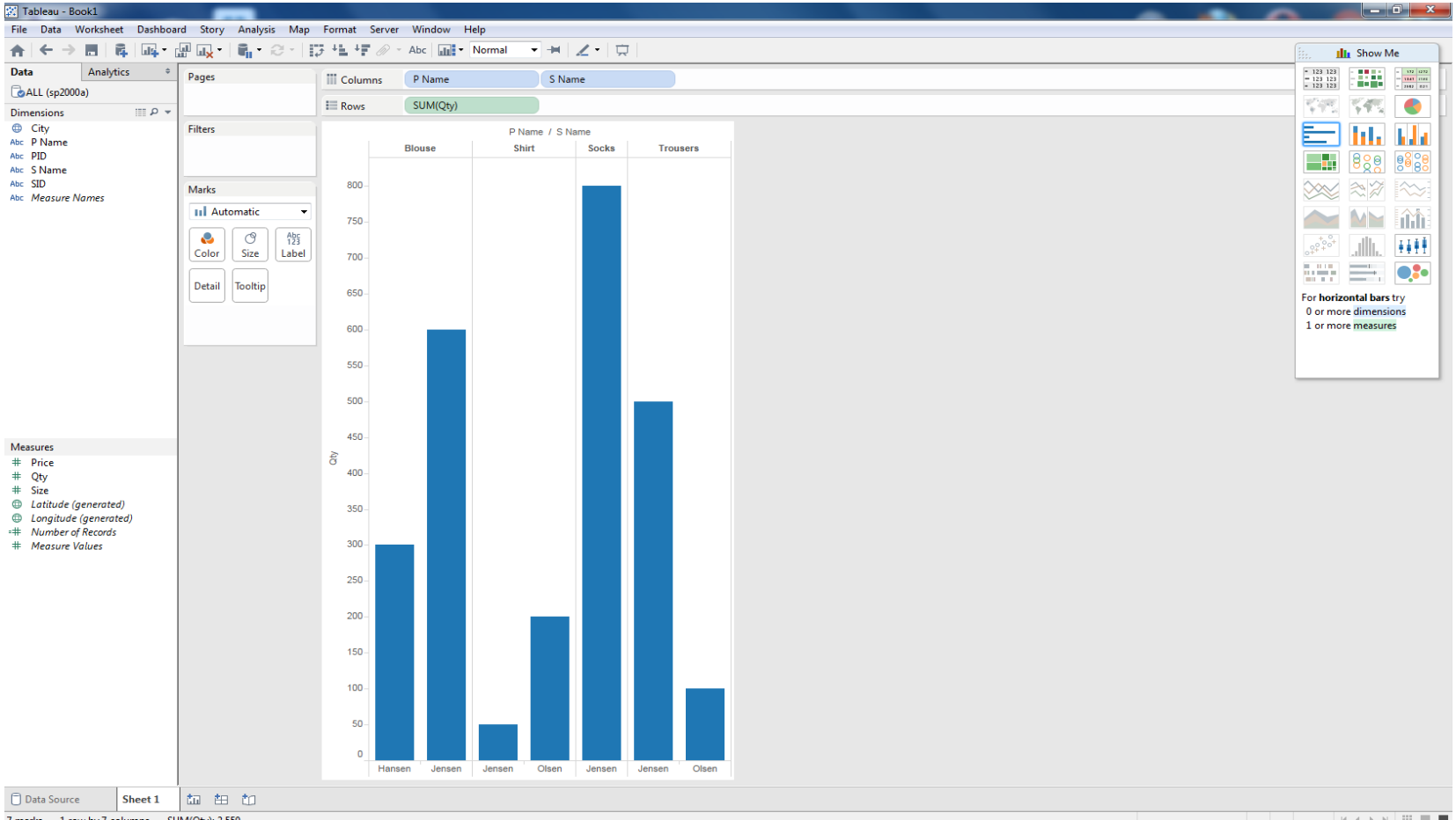
UPDATE AUTOMATICALLY
Get the freshest data with a live connection to your data or get automatic updates on a schedule you define.

EASE OF USE
Anyone can analyze data with Tableau's intuitive drag & drop products. No programming, just insight.

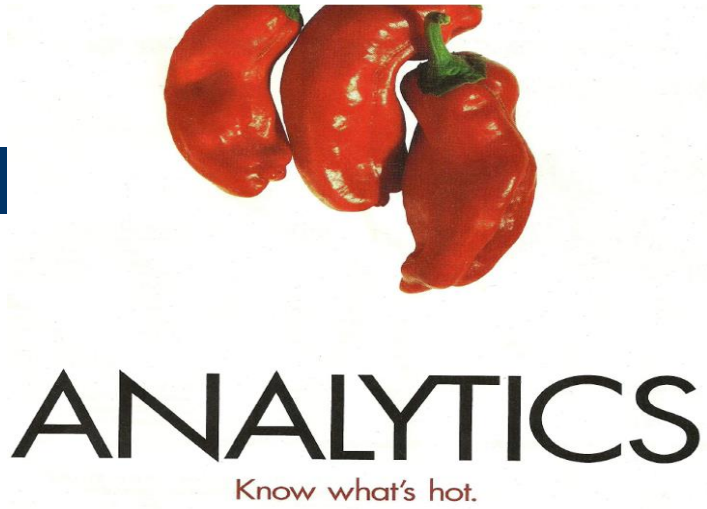
SMART DASHBOARDS
Combine multiple views of data to get richer insight. Best practices of data visualization are baked right in.

SHARE IN SECONDS
Publish a dashboard with a few clicks to share it live on the web and on mobile devices.

Tableau (con’t)



Analytics & Data Mining

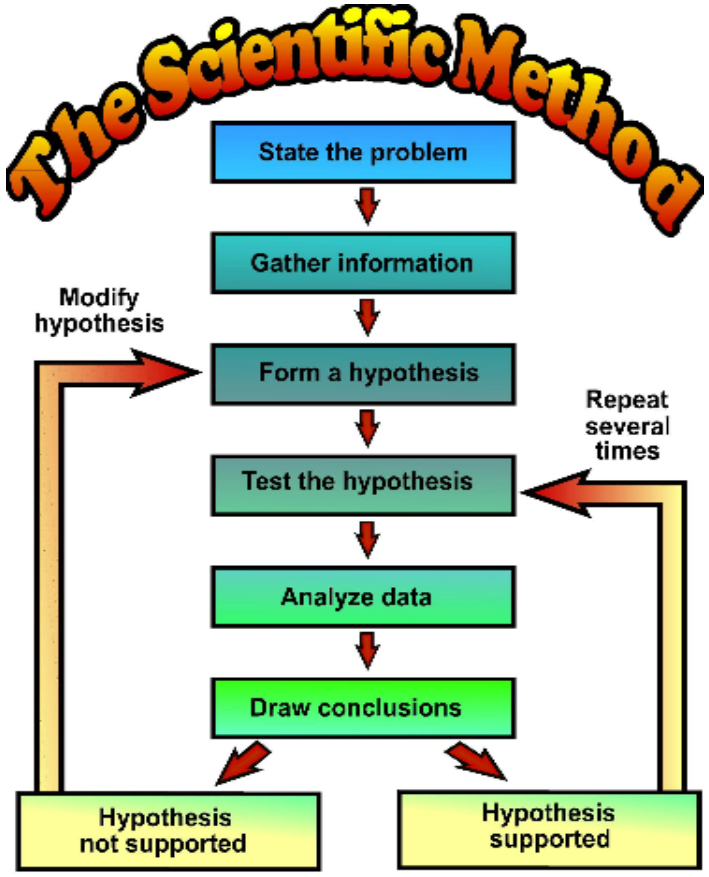


Data Analytics & Data Mining

- Subset of business intelligence (BI) functionality that encompasses a wide range of mathematical, statistical, and modeling techniques with the purpose of extracting knowledge from data
  - Explanatory analytics: focuses on discovering and explaining data characteristics and relationships based on existing data
  - Predictive analytics: focuses on predicting future data outcomes with a high degree of accuracy
- Data mining focuses on the discovery and explanation stages of knowledge acquisition
  - Analyzing massive amounts of data to uncover hidden trends, patterns, and relationships; to form computer models to simulate and explain the findings; and to use such models to support business decision making

The “Scientific Method”

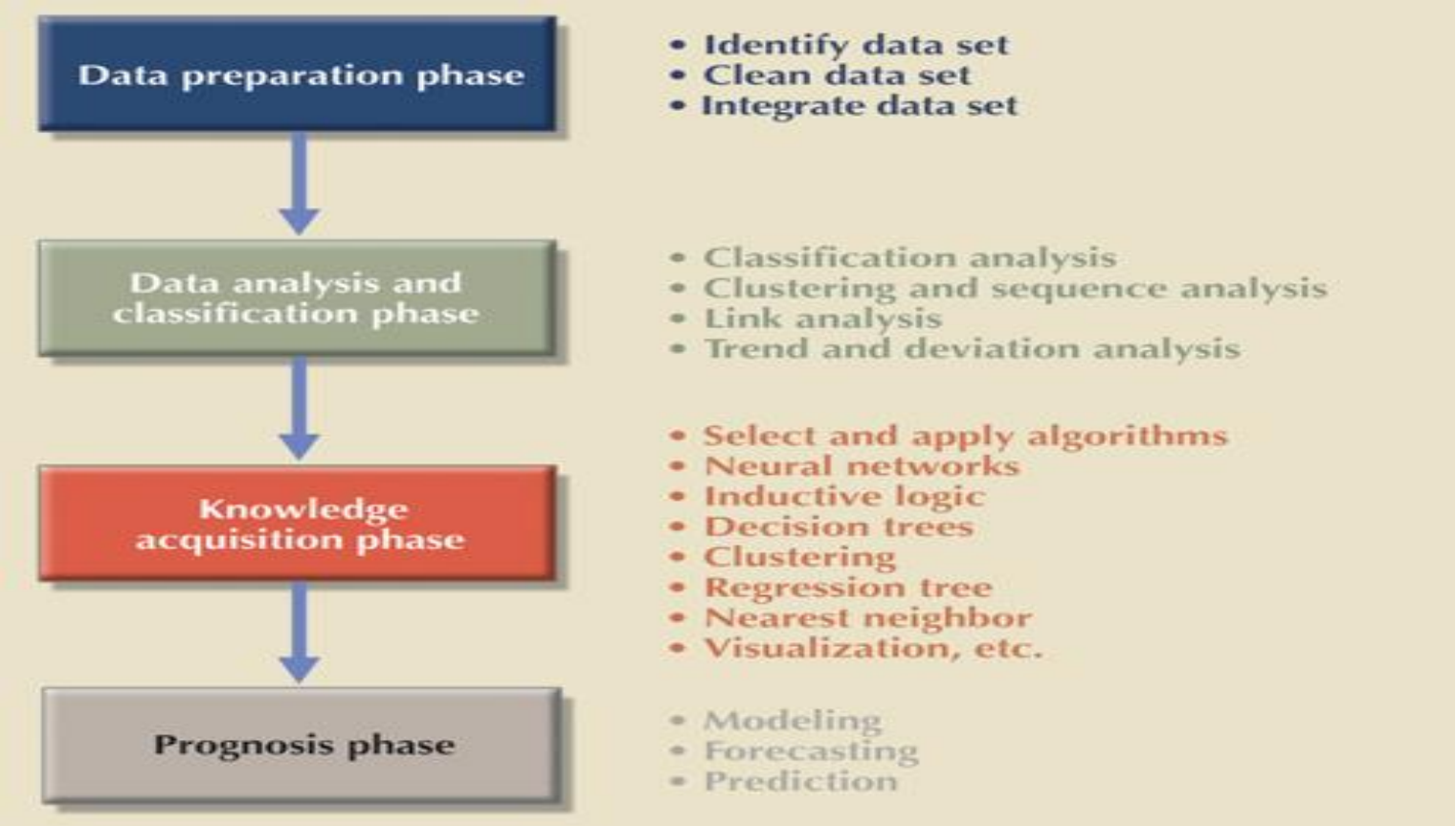
- Formulate a hypothesis
- Gather data:
  - Experiments
  - Surveys
  - Observations
- Use inferential statistics to see if the data supports the hypothesis



Data Mining

- **Data mining** is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems
- The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use
- Aside from the raw analysis step, it involves database and data management aspects, data pre-processing, model and inference considerations, interestingness metrics, complexity considerations, post-processing of discovered structures, visualization, and online updating

Data Mining Phases



Data Mining Techniques

- Nearest-neighbor, clustering, and classification methods
- Text mining and context analysis
- Decision trees and forests
- Neural network computing is a machine learning approach which examines data for patterns
- Intelligent agents retrieving information from the Internet or from intranet-based databases
- Predictive analytics
- Association or affinity analysis uses a specialized set of algorithms that sort through large data sets and express statistical rules among items

Predictive Analytics

- Predictive analytics focuses on creating actionable models to predict future behaviors and events
  - Employs mathematical and statistical algorithms, neural networks, decision trees, and other advanced machine learning tools to create actionable predictive models based on available data
  - Used in areas such as customer relationships, customer service, customer retention, fraud detection, targeted marketing, and optimized pricing

Affinity Analysis
[Market Basket Analysis]

- This is the most widely used and, in many ways, most successful data mining algorithm
- It essentially determines what products people purchase together
- Stores can use this information to place these products in the same area (particularly preferred brands)
- Direct marketers can use this information to determine which new products to offer to their current customers
- Inventory policies can be improved if reorder points reflect the demand for the complementary products

Purchase Information

- Purchase patterns of customers (transaction data) contain a huge wealth of information that many business now use for a variety of purposes:
  - Marketing
  - Up selling
  - Cross selling
  - Recommendations
  - Inventory & logistics
  - Store management
  - This is often combined with shopper ID information

Association Rules for Market Basket Analysis

Rules are derived in the form “left-hand side implies right-hand side” and an example is:

Yellow Peppers IMPLIES Red Peppers, Bananas

Unidirectional Rules

- The rules are unidirectional
- The following is an “obvious” rule:
  - Caviar IMPLIES Vodka
- But the reverse is not true:
  - Vodka IMPLIES Caviar

Measures of Predictive Ability

1. *Support* (prevalence) refers to the percentage of baskets where both left and right side products were present
2. *Confidence* measures what percentage of baskets that contained the left-hand product also contained the right
3. *Lift* measures how much more frequently the left-hand item is found with the right than pure chance (the product of their individual probabilities of occurrence)

Example rule:

- Green Peppers IMPLIES Bananas
  - Confidence – 85.96
    - About 86% of the baskets with green peppers also had bananas
  - Support – 3.77
    - About 4% of the baskets had both green peppers and bananas
  - Lift – 1.37
    - It is 1.37 times more likely to find green peppers with bananas than the product of their individual probabilities (probability of green peppers AND bananas)

Market Basket Analysis Methodology

- We first need a list of transactions of what was purchased - this is readily available with electronic cash registers
- Next, we choose a list of products to analyze, and tabulate how many times each was purchased with the others
- The diagonals of the table shows how often a product is purchased in any combination, and the off-diagonals show which combinations were bought

A Small Simple Store Example

Consider the following simple example about five transactions at a convenience store:

Transaction 1: Frozen pizza, cola, milk
Transaction 2: Milk, potato chips
Transaction 3: Cola, frozen pizza
Transaction 4: Milk, pretzels
Transaction 5: Cola, pretzels

Transaction 1: Frozen pizza, cola, milk
Transaction 2: Milk, potato chips
Transaction 3: Cola, frozen pizza
Transaction 4: Milk, pretzels
Transaction 5: Cola, pretzels

Cross Tabulation in a Table

Product Bought	Pizza also	Milk also	Cola also	Chips also	Pretzels also
Pizza	2	1	2	0	0
Milk	1	3	1	1	1
Cola	2	1	3	0	1
Chips	0	1	0	1	0
Pretzels	0	1	1	0	2

- Pizza and Cola sell together more often than any other combo; a cross-marketing opportunity?
- Milk sells well with everything – people probably come here specifically to buy it

Market Basket Concepts

- Transaction – the purchase of one or more items by a customer at one point in time and space – a “shopping cart” or “market basket”
- Association Rule – a rule which suggests a relationship between items in the transaction, written as for single items A and B:
  - A IMPLIES B (or A → B)

Support

- Support – the % of transactions (baskets) where an association rule applies – where we see both item A and B in the same basket
  - For example, if 500 baskets contain both A and B out of a total of 1000 baskets, then the support is 50%
  - A→B and B→A both have the same support

Confidence

- Confidence – measures the predictive accuracy of a rule
- Confidence is the probability that item B is in the basket if item A is in the basket ("conditional probability") → P(B|A) = P(AB)/P(A)
- Calculated as:
  - Support (A & B)/P(A) where support (A) is the % of baskets containing A
  - For example, if 500 baskets contain both A and B out of a total of 1000 baskets, then the support of A & B is 50%
  - If A is in 75% of baskets, the confidence is 50/75 or 67%

Lift

- Lift - the ratio of support to a product to the individual probabilities of both sides
  - P(AB)/(P(A) * P(B))
- For example:
  - For example, if 500 baskets contain both A and B out of a total of 1000, then the support of A & B is 50%
  - If A is in 75% of baskets and B is in 20% of the baskets, then the lift is:
    - .50/(.75*.20) = 3.33

Computing Support

	Pizza	Milk	Cola	Chips	Pretzels
Pizza	2	1	2	0	0
Milk	1	3	1	1	1
Cola	2	1	3	0	1
Chips	0	1	0	1	0
Pretzels	0	1	1	0	2

The support measure for Cola IMPLIES Pizza is 40% (2/5).
Of the 5 transactions 2 have both cola and pizza.
Note support does not consider direction (Pizza IMPLIES Cola is also 40%).

Computing Confidence

	Pizza	Milk	Cola	Chips	Pretzels
Pizza	2	1	2	0	0
Milk	1	3	1	1	1
Cola	2	1	3	0	1
Chips	0	1	0	1	0
Pretzels	0	1	1	0	2

Milk IMPLIES Chips has a confidence of 33%, since the support of "Milk plus Chips" is 20% (1/5) and Milk is in 60% of baskets (3/5).

Thus 20%/60% is 33. Confidence is unidirectional !

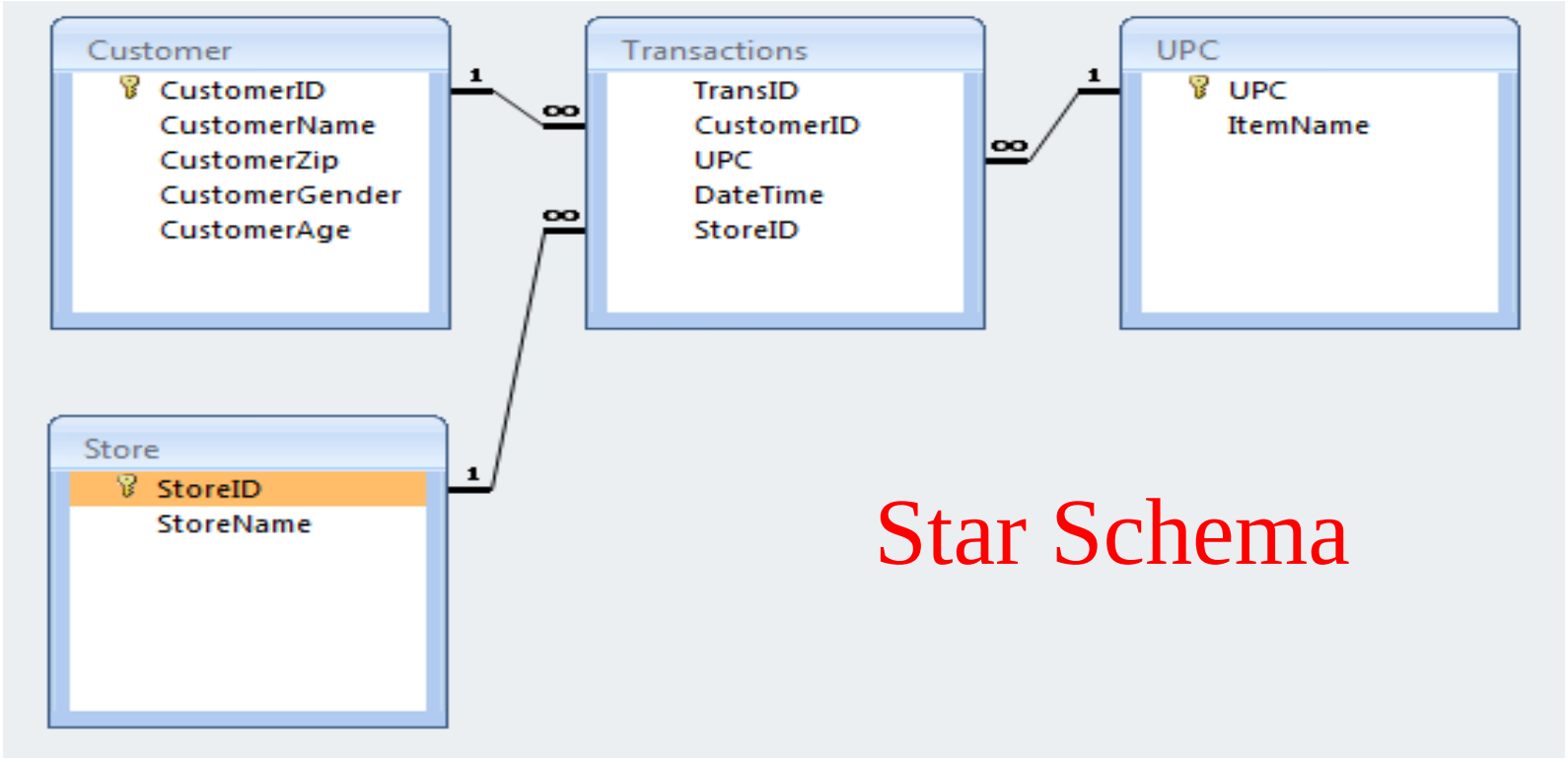
Computing Lift

	Pizza	Milk	Cola	Chips	Pretzels
Pizza	2	1	2	0	0
Milk	1	3	1	1	1
Cola	2	1	3	0	1
Chips	0	1	0	1	0
Pretzels	0	1	1	0	2

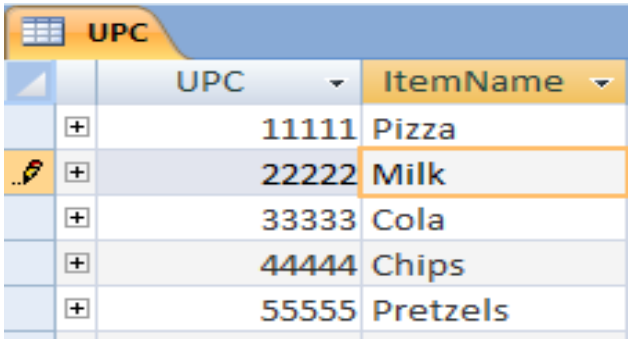
Lift is the ratio of support of a product to the individual joint probabilities of both sides.

Cola IMPLIES Pizza lift is .40/(.60 * .40) = 1.67.

Example Database

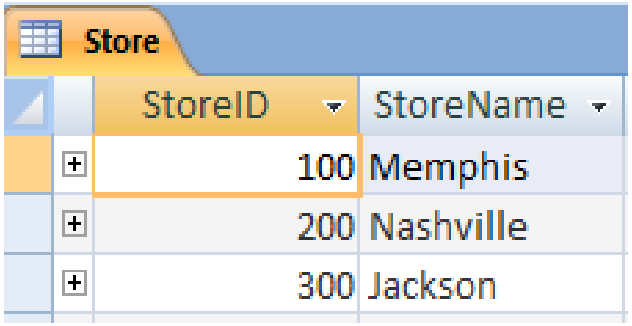


Example Access Data



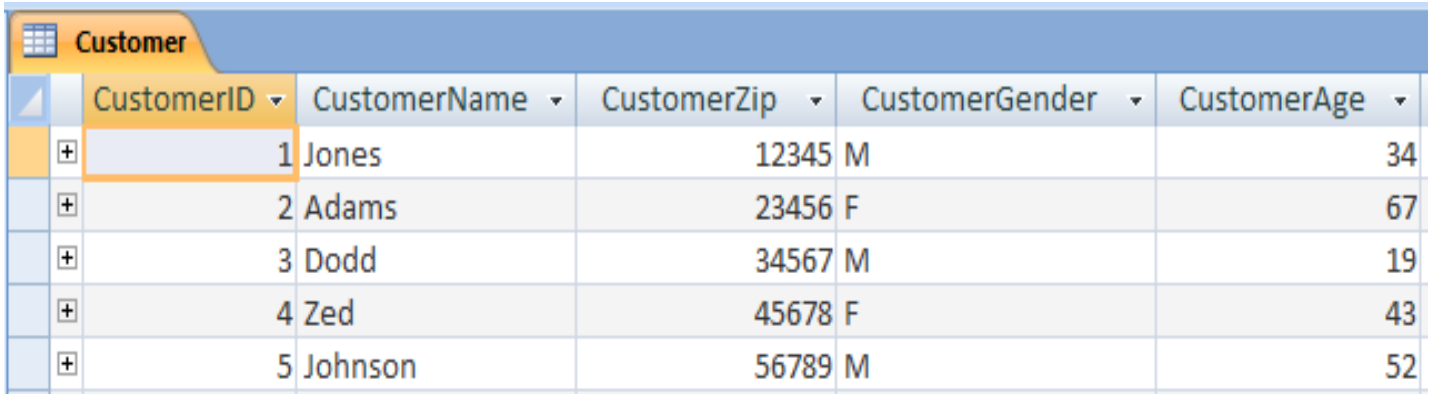
UPC

UPC	ItemName
11111	Pizza
22222	Milk
33333	Cola
44444	Chips
55555	Pretzels



Store

StoreID	StoreName
100	Memphis
200	Nashville
300	Jackson



Customer

CustomerID	CustomerName	CustomerZip	CustomerGender	CustomerAge
1	Jones	12345	M	34
2	Adams	23456	F	67
3	Dodd	34567	M	19
4	Zed	45678	F	43
5	Johnson	56789	M	52

Transaction Example Data

Transaction 1: Frozen pizza, cola, milk
Transaction 2: Milk, potato chips
Transaction 3: Cola, frozen pizza
Transaction 4: Milk, pretzels
Transaction 5: Cola, pretzels

Transactions

TransID	CustomerID	UPC	DateTime	StoreID
1	5	11111		100
1	5	33333		100
1	5	22222		100
2	4	22222		100
2	4	44444		100
3	3	33333		100
3	3	11111		100
4	2	22222		100
4	2	55555		100
5	1	33333		100
5	1	55555		100

Baskets View

[SELECT Transactions.TransID, UPC.ItemName AS Item FROM UPC INNER JOIN Transactions ON UPC.UPC=Transactions.UPC;]

Baskets

TransID	Item
1	Pizza
3	Pizza
1	Milk
2	Milk
4	Milk
1	Cola
3	Cola
5	Cola
2	Chips
4	Pretzels
5	Pretzels

Cross Product to Find Products Selling Together

[each row in the first table combined with each row in the second table]

Table TABA

Field 1	Field 2
1	Text 1
2	Text 2

Table TABB

Field 3	Field 4	Field 5
1	A	Text 3
1	B	Text 4
2	A	Text 5
2	B	Text 6

Field 1	Field 2	Field 3	Field 4	Field 5
1	Text 1	1	A	Text 3
1	Text 1	1	B	Text 4
1	Text 1	2	A	Text 5
1	Text 1	2	B	Text 6
2	Text 2	1	A	Text 3
2	Text 2	1	B	Text 4
2	Text 2	2	A	Text 5
2	Text 2	2	B	Text 6

Cross product of tables TABA and TABB

Pairs View (same basket [transaction])

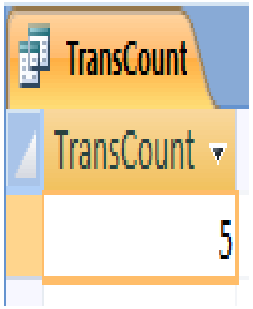
**[SELECT T1.Item AS Item1, T2.Item AS Item2
FROM Baskets AS T1, Baskets AS T2
WHERE T1.transID=T2.transID And T1.Item<>T2.Item;]**

Pairs

Item1	Item2
Cola	Pizza
Milk	Pizza
Pizza	Cola
Milk	Cola
Pizza	Milk
Cola	Milk
Chips	Milk
Milk	Chips
Pizza	Cola
Cola	Pizza
Pretzels	Milk
Milk	Pretzels
Pretzels	Cola
Cola	Pretzels

Transaction Count

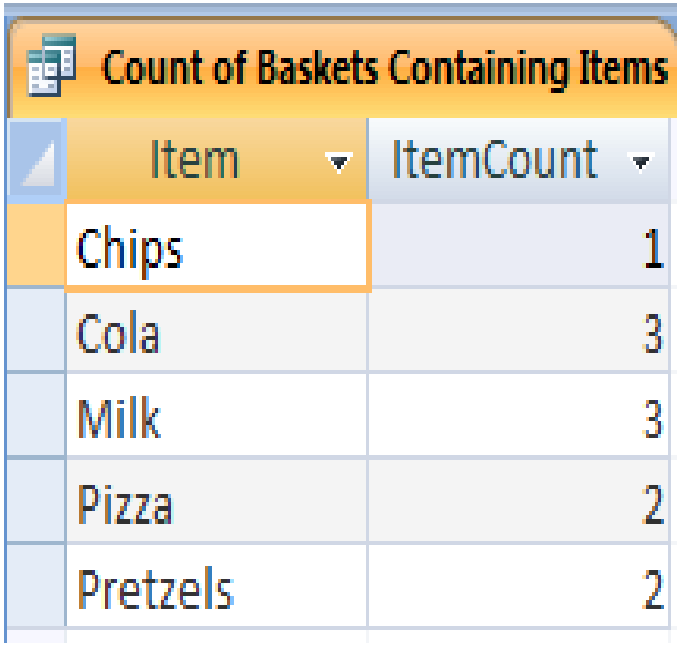
- Standard SQL
  - SELECT count(DISTINCT TransID) AS TransCount
    FROM Baskets
- Access SQL
  - SELECT count(*) AS TransCount
    FROM (SELECT DISTINCT TransID FROM baskets)



Grouping View

- SELECT Item, count(*) AS ItemCount
- FROM Baskets
- GROUP BY Item;

Transaction 1: Frozen pizza, cola, milk
Transaction 2: Milk, potato chips
Transaction 3: Cola, frozen pizza
Transaction 4: Milk, pretzels
Transaction 5: Cola, pretzels



Count of Baskets Containing Items

Item	ItemCount
Chips	1
Cola	3
Milk	3
Pizza	2
Pretzels	2

Support Count

- From the transactions, how many are for each pair:
  - SELECT Item1, Item2, count(*) AS SupportCount
  - FROM Pairs
  - GROUP BY Item1, Item2;

SupportCount

Item1	Item2	SupportCount
Chips	Milk	1
Cola	Milk	1
Cola	Pizza	2
Cola	Pretzels	1
Milk	Chips	1
Milk	Cola	1
Milk	Pizza	1
Milk	Pretzels	1
Pizza	Cola	2
Pizza	Milk	1
Pretzels	Cola	1
Pretzels	Milk	1

Support

- From the transactions, how many are for each pair as a percentage of the total transactions:
  - SELECT Item1, Item2, count(*) AS SupportCount, count(*)/(SELECT count(*) AS TransCount FROM (SELECT DISTINCT transID FROM transactions)) AS Support
  - FROM Pairs
  - GROUP BY Item1, Item2;

Support (con't)

Support

Item1	Item2	SupportCount	Support
Chips	Milk	1	0.2
Cola	Milk	1	0.2
Cola	Pizza	2	0.4
Cola	Pretzels	1	0.2
Milk	Chips	1	0.2
Milk	Cola	1	0.2
Milk	Pizza	1	0.2
Milk	Pretzels	1	0.2
Pizza	Cola	2	0.4
Pizza	Milk	1	0.2
Pretzels	Cola	1	0.2
Pretzels	Milk	1	0.2

Cola IMPLIES Pizza support is 40%; of the 5 transactions, 2 have both Cola and Pizza.
Pizza IMPLIES Cola is also 40% (support does not consider direction)

Confidence

- Support divided by % of baskets containing the first product in the rule
  - SELECT Item1, Item2, count(*) AS SupportCount, count(*)/(SELECT count(*) AS TransCount FROM (SELECT DISTINCT transID FROM baskets)) AS Support, (select count(*) from baskets where Item=Item1)/(SELECT count(*) AS TransCount FROM (SELECT DISTINCT transID FROM baskets)) AS Item1inBaskets, Support/Item1inBaskets AS Confidence
  - FROM Pairs
  - GROUP BY Item1, Item2;

Confidence (con't)

Support-Confidence

Item1	Item2	SupportCount	Support	Item1inBaskets	Confidence
Chips	Milk	1	0.2	0.2	1
Cola	Milk	1	0.2	0.6	0.333333333333333
Cola	Pizza	2	0.4	0.6	0.666666666666667
Cola	Pretzels	1	0.2	0.6	0.333333333333333
Milk	Chips	1	0.2	0.6	0.333333333333333
Milk	Cola	1	0.2	0.6	0.333333333333333
Milk	Pizza	1	0.2	0.6	0.333333333333333
Milk	Pretzels	1	0.2	0.6	0.333333333333333
Pizza	Cola	2	0.4	0.4	1
Pizza	Milk	1	0.2	0.4	0.5
Pretzels	Cola	1	0.2	0.4	0.5
Pretzels	Milk	1	0.2	0.4	0.5

Milk IMPLIES Chips has a confidence of .33 [.2 divided by .6]
Chips IMPLIES Milk has a confidence of 1

Lift

- Lift is the ratio of support to the product of the individual probabilities
  - SELECT Item1, Item2, count(*) AS SupportCount, count(*)/(SELECT count(*) AS TransCount FROM (SELECT DISTINCT transID FROM baskets)) AS Support, (select count(*) from baskets where Item=Item1)/(SELECT count(*) AS TransCount FROM (SELECT DISTINCT transID FROM baskets)) AS Item1inBaskets, Support/Item1inBaskets AS Confidence, (select count(*) from baskets where Item=Item2)/(SELECT count(*) AS TransCount FROM (SELECT DISTINCT transID FROM baskets)) AS Item2inBaskets, Support/(Item1inBaskets*Item2inBaskets) AS Lift
  - FROM Pairs
  - GROUP BY Item1, Item2;

Lift (con't)

Support-Confidence-Lift

Item1	Item2	SupportCount	Support	Item1inBaskets	Confidence	Item2inBaskets	Lift
Chips	Milk	1	0.2	0.2	1	0.6	1.66666666666667
Cola	Milk	1	0.2	0.6	0.333333333333333	0.6	0.555555555555556
Cola	Pizza	2	0.4	0.6	0.666666666666667	0.4	1.66666666666667
Cola	Pretzels	1	0.2	0.6	0.333333333333333	0.4	0.833333333333333
Milk	Chips	1	0.2	0.6	0.333333333333333	0.2	1.66666666666667
Milk	Cola	1	0.2	0.6	0.333333333333333	0.6	0.555555555555556
Milk	Pizza	1	0.2	0.6	0.333333333333333	0.4	0.833333333333333
Milk	Pretzels	1	0.2	0.6	0.333333333333333	0.4	0.833333333333333
Pizza	Cola	2	0.4	0.4	1	0.6	1.66666666666667
Pizza	Milk	1	0.2	0.4	0.5	0.6	0.833333333333333
Pretzels	Cola	1	0.2	0.4	0.5	0.6	0.833333333333333
Pretzels	Milk	1	0.2	0.4	0.5	0.6	0.833333333333333

The lift for the rule "Cola IMPLIES Pizza" is .4/ (.6 * .4) = 1.67

Selecting Rules ("mining")

- To select the relevant rules, one would select rows from the previous tables where the support, confidence, and lift met minimum criteria, such as WHERE Support>=0.4 AND Confidence>=1 AND Lift>=1;

Rules

Item1	Item2	Support	Confidence	Lift
Pizza	Cola	0.4	1	1.66666666666667

Performing Analysis with Virtual Items

- The sales data can be augmented with the addition of “virtual items” -- For example, we could record that the customer was new to us, or had children
- The transaction record might look like:

  Item 1: Sweater Item 2: Jacket Item 3: New
- This might allow us to see what patterns new customers have versus old customers

Multidimensional Market Basket Analysis

- Rules can involve more than two items, for example Plant and Clay Pot IMPLIES Soil
- These rules are built iteratively -- first, pairs are found, then relevant sets of three or four
- In our example here, one would join the “pairs” table to itself, to formulate a “triples” table
- These are then pruned by removing those that occur infrequently
- In an environment like a grocery store, where customers commonly buy over 100 items, rules could involve as many as 10 items

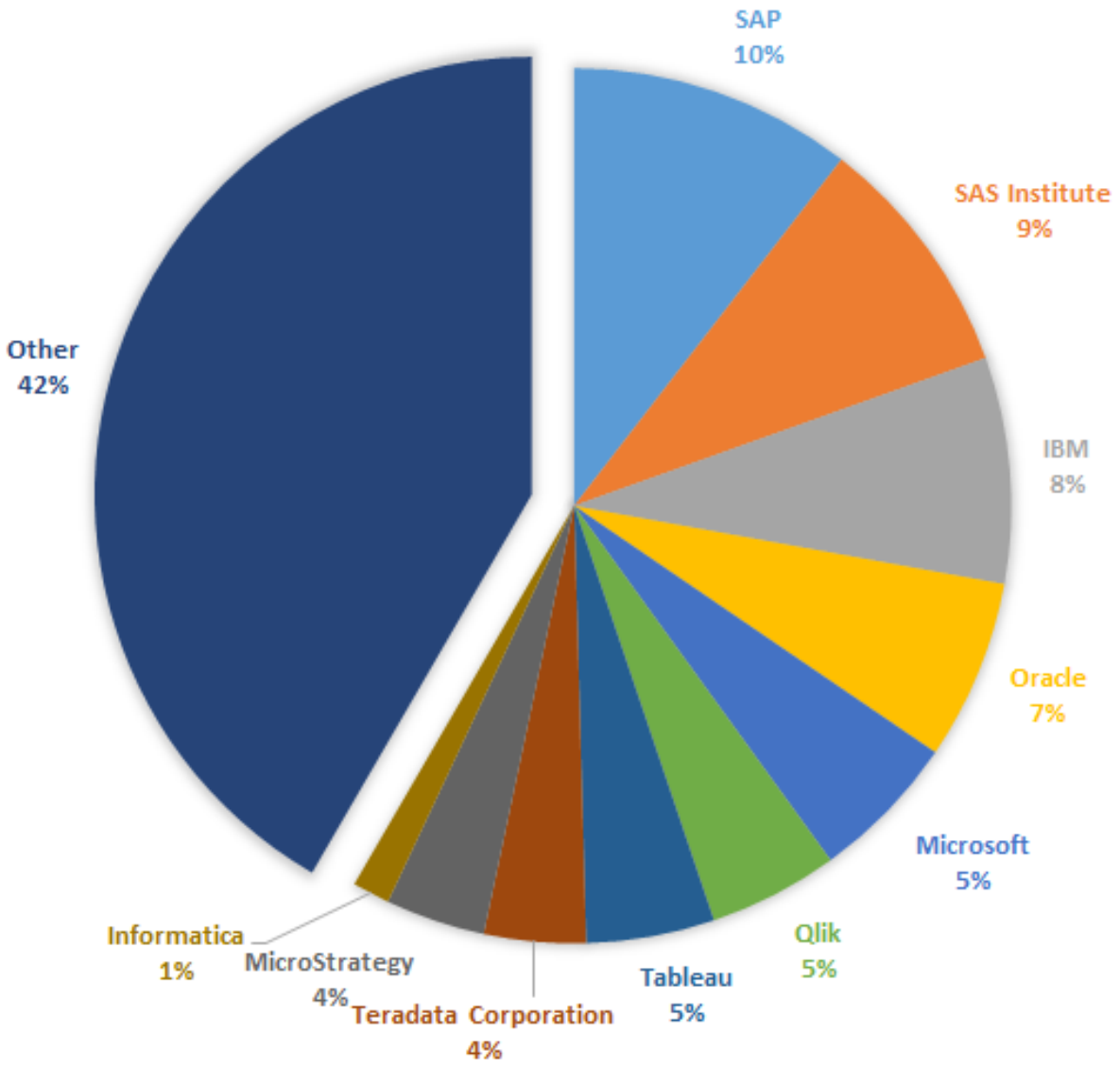
Using the Results

- The tabulations can immediately be translated into association rules and the numerical measures computed
- Comparing this week's table to last week's table can immediately show the effect of this week's promotional activities
- Some rules are going to be *trivial* (hot dogs and buns sell together) or *inexplicable (*toilet rings sell only when a new hardware store is opened)

Data Analytics Tools

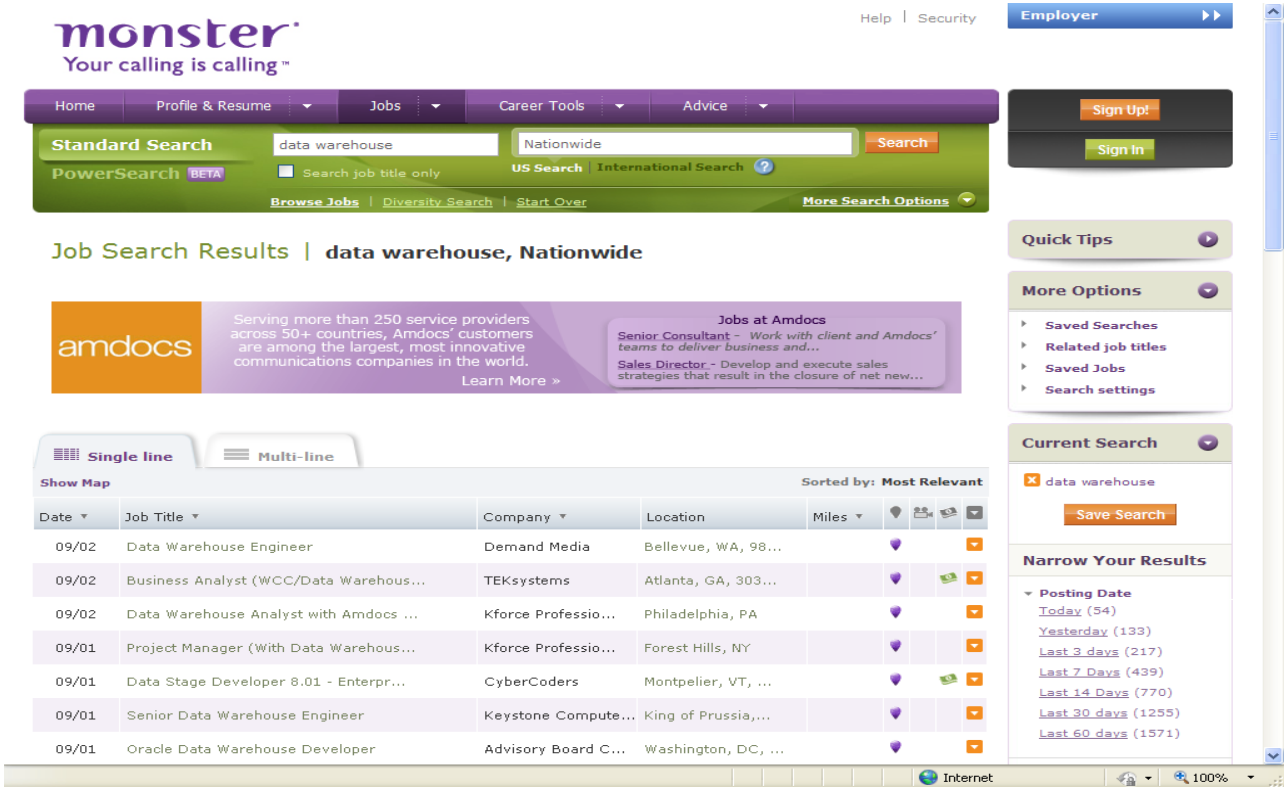
- To perform data analytics tasks one can use SQL for some tasks such as affinity analysis
- There are also many tools with built-in procedures for common algorithms such as SAS
- For the most sophisticated and powerful machine learning algorithms, data scientists typically use programing languages such as R and Python

2015 BI Revenue by Vendor



Data Warehousing Careers

[search data warehouse, business intelligence, analytics, etc.]



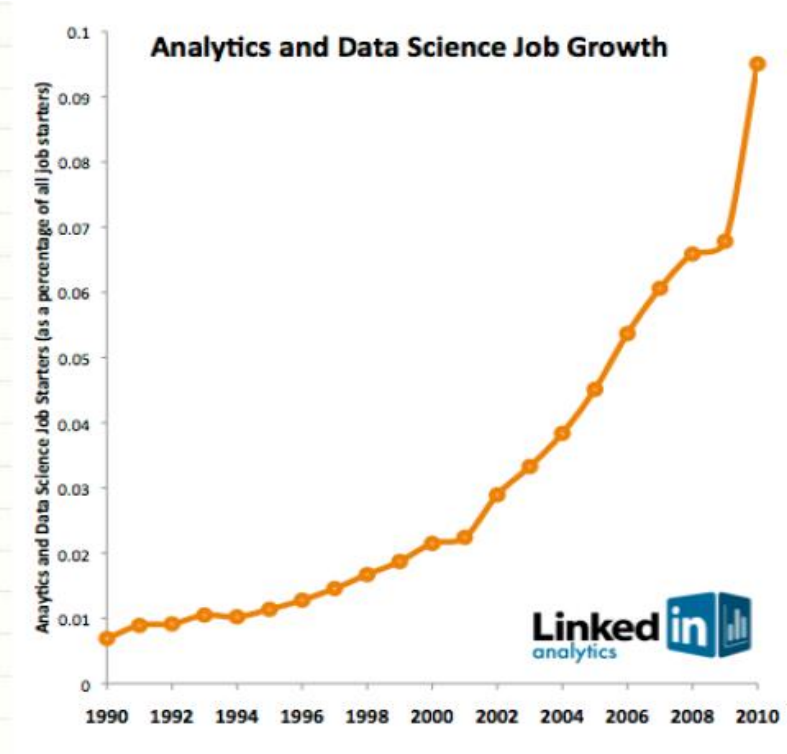
BI Jobs

- ComputerWorld's Survey of its 100 IT leaders ranked their top five priorities as:
  - Business analytics
  - Mobility (tablets, apps, etc)
  - Application development
  - Cloud computing
  - Security

Data Analytics Jobs

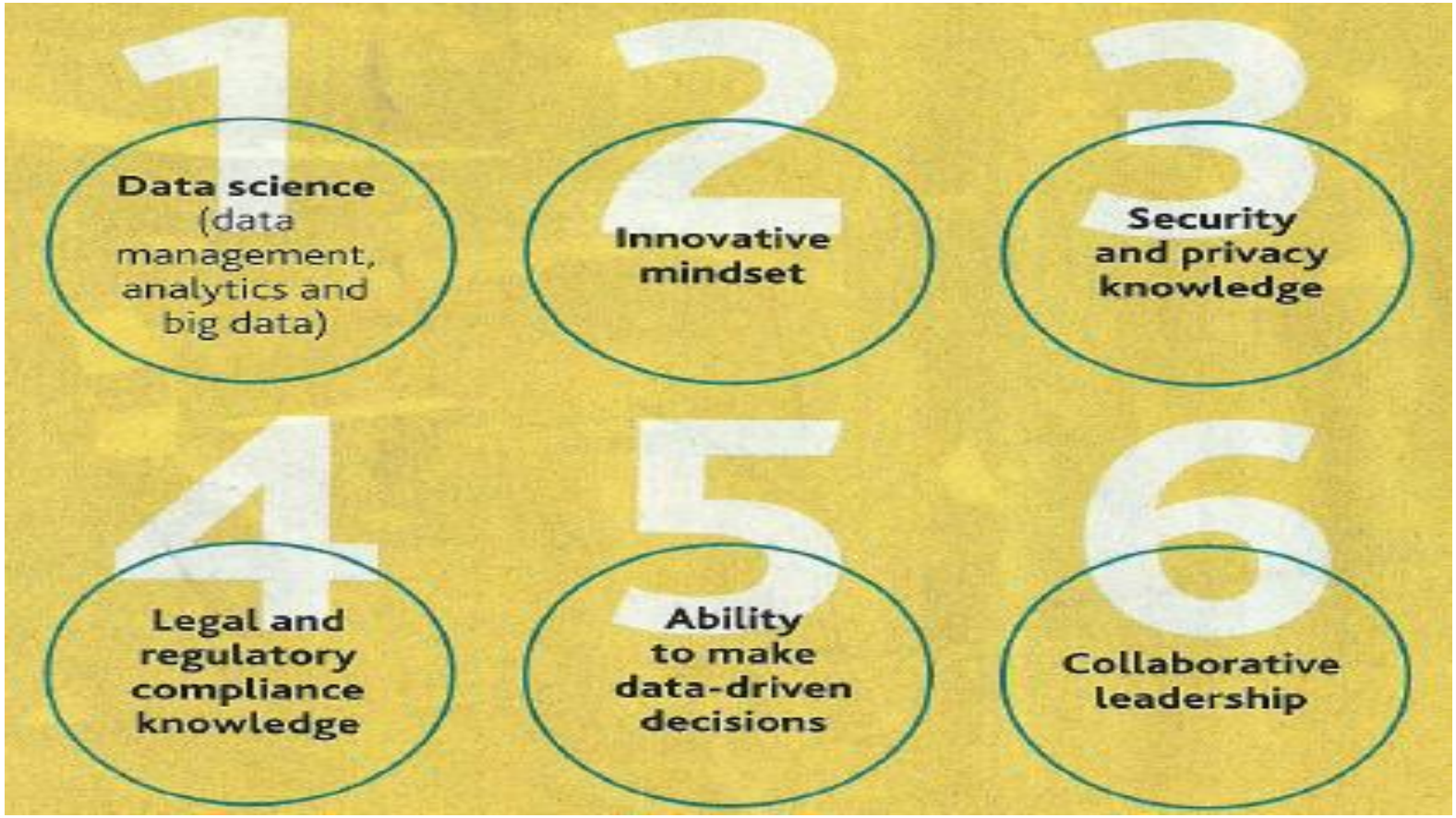
A report released by Glassdoor says that data scientists have the best jobs in the U.S., according to that company's analysis.

With a median base salary of $116,840, more than 1,700 job openings on Glassdoor's site, and a user-provided career opportunities rating of 4.1, "data scientist" took the prize for most highly rated job title in America.

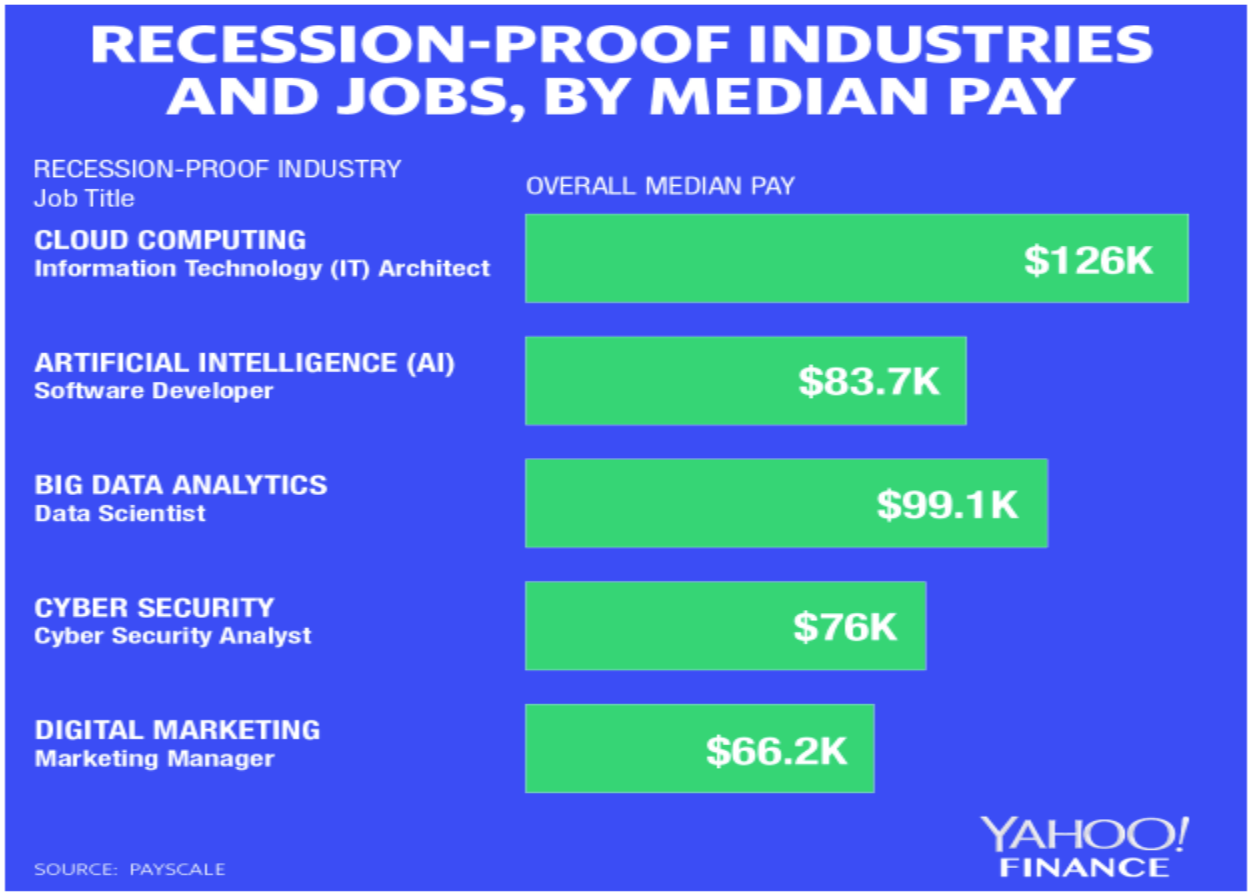


Top Skills Needed

[Skills to Thrive in Disruptive Times, PMI, 2018 Interviews of 500 HR and 500 PM Professionals]



Yahoo Finance, 2019



The Sexiest Job of the 21st Century

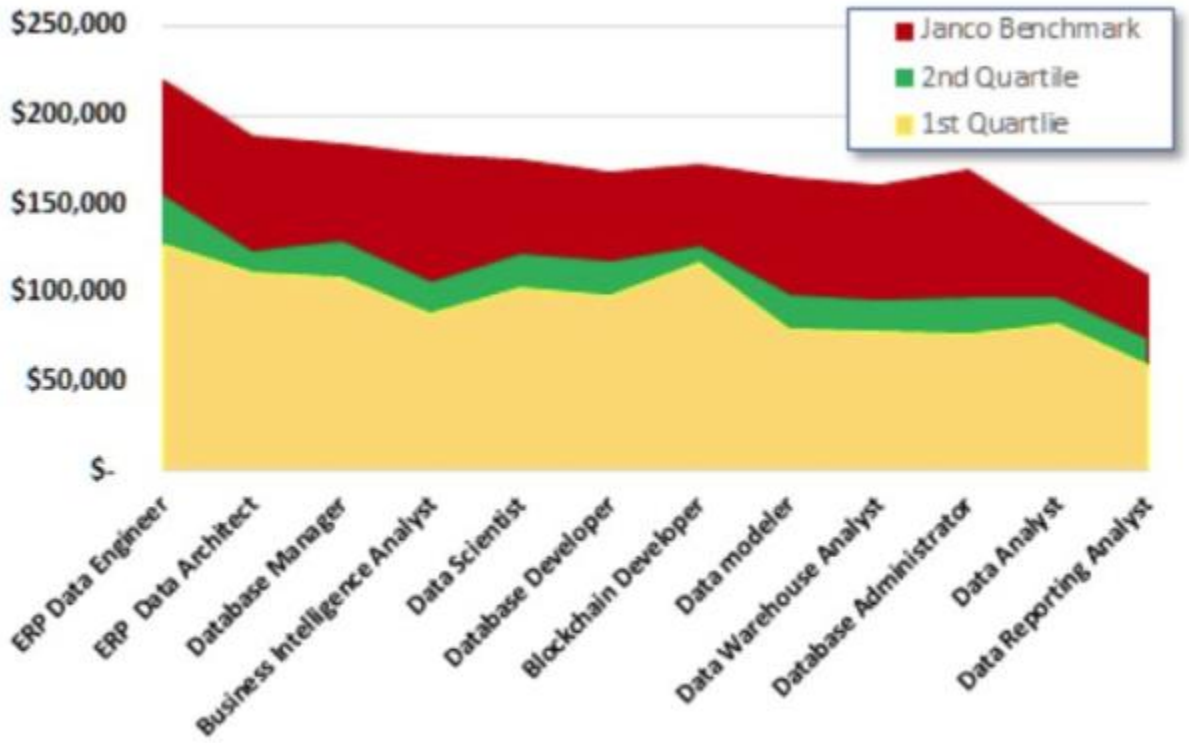
- The sexiest job of the 21st century is not a social media entrepreneur nor Hollywood producer - if you take a cue from the Harvard Business Review, the title goes to data scientists
- Just how in demand are these candidates?
- PEHub cites a tech recruiter who says he's aware of thousands of data science jobs awaiting candidates in the Silicon Valley area. Nationally, McKinsey & Co. estimates the U.S. has as many as 190,000 fewer people with analytic expertise than needed.
- PEHub reports pay for data scientists is upwards of $225,000 even for people straight out of graduate school, up from $125,000 just a few years ago. For someone with a few years of experience working in the field, pay can reach much higher. One Seattle software-company CEO describes candidates with these skillsets "almost like unicorns." One got away from the executive when Microsoft approached the data scientist with a $650,000 salary plus bonuses.

GlassDoor's 2020 Best Jobs

- **1. Front end engineer**
- Job satisfaction rating: 3.9, Number of job openings: 13,122, Median base salary: $105,240
- **2. Java developer**
- Job satisfaction rating: 3.9, Number of job openings: 16,136, Median base salary: $83,589
- **3. Data scientist**
- Job satisfaction rating: 4.0, Number of job openings: 6,542, Median base salary: $107,801
- **4. Product manager**
- Job satisfaction rating: 3.8, Number of job openings: 12,173, Median base salary: $117,713
- **5. Devops engineer**
- Job satisfaction rating: 3.9, Number of job openings: 6,603, Median base salary: $107,310
- **6. Data engineer**
- Job satisfaction rating: 3.9, Number of job openings: 6,941, Median base salary: $102,472
- **7. Software engineer**
- Job satisfaction rating: 3.6, Number of job openings: 50,438, Median base salary: $105,563
- **8. Speech language pathologist**
- Job satisfaction rating: 3.8, Number of job openings: 29,167, Median base salary: $71,867
- **9. Strategy manager**
- Job satisfaction rating: 4.3, Number of job openings: 3,515, Median base salary: $133,067
- **10. Business development manager**
- Job satisfaction rating: 4.0, Number of job openings: 6,560, Median base salary: $78,480


Big Data Market Forecast, 2011-2026 ($US B)
$90.00
$80.00
$70.00
$60.00
$50.00
$40.00
$30.00
$20.00
$10.00
$-
WIKIBON
$7.60
$12.25
$19.60
$27.36
$33.31
$37.97
$43.40
$49.28
$55.22
$60.91
$66.11
$70.76
$74.88
$78.53
$81.78
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026

Hottest IT Jobs



Data Scientist

Data Scientist

- Robert Half
  - It pays to be a big data engineer
  - Not only will salaries soar, rising about 6 percent year over year, but they'll also generate more income than anyone else in IT
  - Big data engineers at the lower end will make $135,000 next year
  - At the top of the scale, big data engineers can expect to make $196,000

References

- Data Warehousing For Dummies by Thomas C. Hammergren
- The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling by Ralph Kimball and Margy Ross
- Data Warehousing Essentials by Julio Bolton | Jun 28, 2019

Homework

- Textbook Chapter 13
- Textbook Review Questions 1 thru 7
- Appendix: Excel Power Pivot →
- Create a crosstab query for the Access S-O-C model
  - Count of orders is the measure
  - Customers are the rows, and salespersons are the columns

Excel Power Pivot

OLAP

- Excel Power Pivot provides a limited form of *OLAP* (OnLine Analytical Processing) – a form of Multidimensional analysis
- The OLAP method of data analysis requires two steps:
  - Create an OLAP database (Excel calls this a Data Model)
  - Use reporting tools to analyze and visualize the data residing in the model
- *Reference: Excel 365 Expert Skills with the Smart Method*

Tools To Create A Data Model

- Get & Transform (previously named: Power Query)
  - A tool for cleaning (transforming) source data and loading the cleaned data into Power Pivot tables; it is a tool for ETL (Extract, Transform and Load)
- Power Pivot
  - A tool used to take the data tables that Get & Transform has pre-prepared and define relationships between them to create data models (also called OLAP databases)

Tools To Analyze And Visualize The Data

- OLAP Pivot Table
  - A tool that is able to present the summarized data contained in a data model
- 3-D Maps (previously named: Power Maps)
  - An easy-to-learn application that provides a way to visualize geographical data contained in a data model
- Excel itself
  - Excel can directly access summarized data residing in a data model; Excel's extensive analysis, charting and other visualization features can then be used

OLTP vs OLAP

- Most corporate data is held in a relational database (like SQL Server, MySQL, Oracle and Access)
- Relational databases consist of a series of tables linked by relationships
- Relational databases are designed for finding, adding, deleting and editing rows of data stored in tables
- This type of database is called an OLTP database (OnLine Transaction Processing)

OLTP vs OLAP (con't)

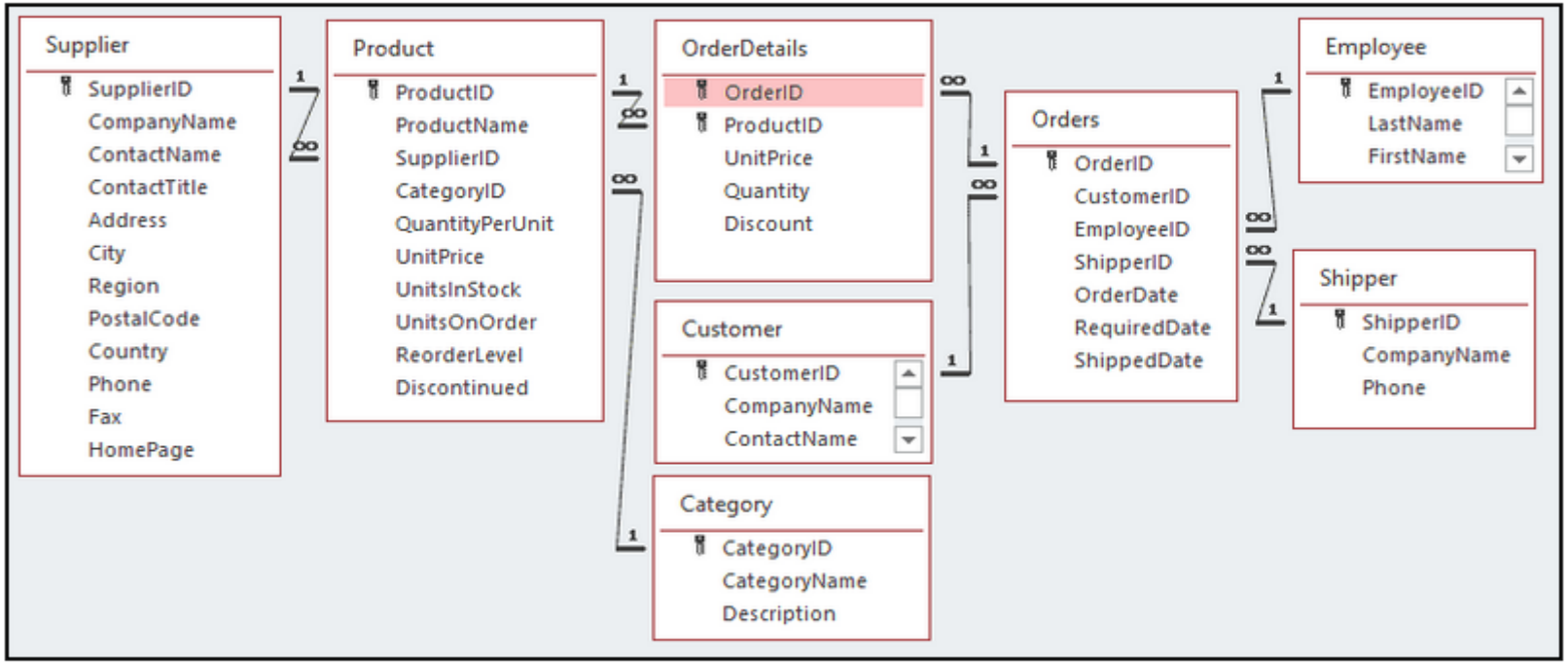
- For reporting and analyzing data you need a different type of database called an OLAP database
- OLAP databases are designed for quickly creating summary reports for large data sets
- Excel Power Pivot calls this type of database a *Data Model*

OLTP vs OLAP (con't)

- An OLTP database is designed around the need to quickly perform four actions: *Create, Retrieve, Update* and *Delete (CRUD)*
- Well-designed OLTP databases must also conform to Third Normal Form (3NF) and are said to contain *Normalized Data*
- A normalized database reduces the risk of data corruption during transactional processing
- OLTP databases can contain one-to-many and many-to-many relationships

Relational Database

- Tables, keys, and relationships
  - The OrderDetails table is an “intersection” table used to break the many-to-many relationship between Product and Orders into two one-to-many relations

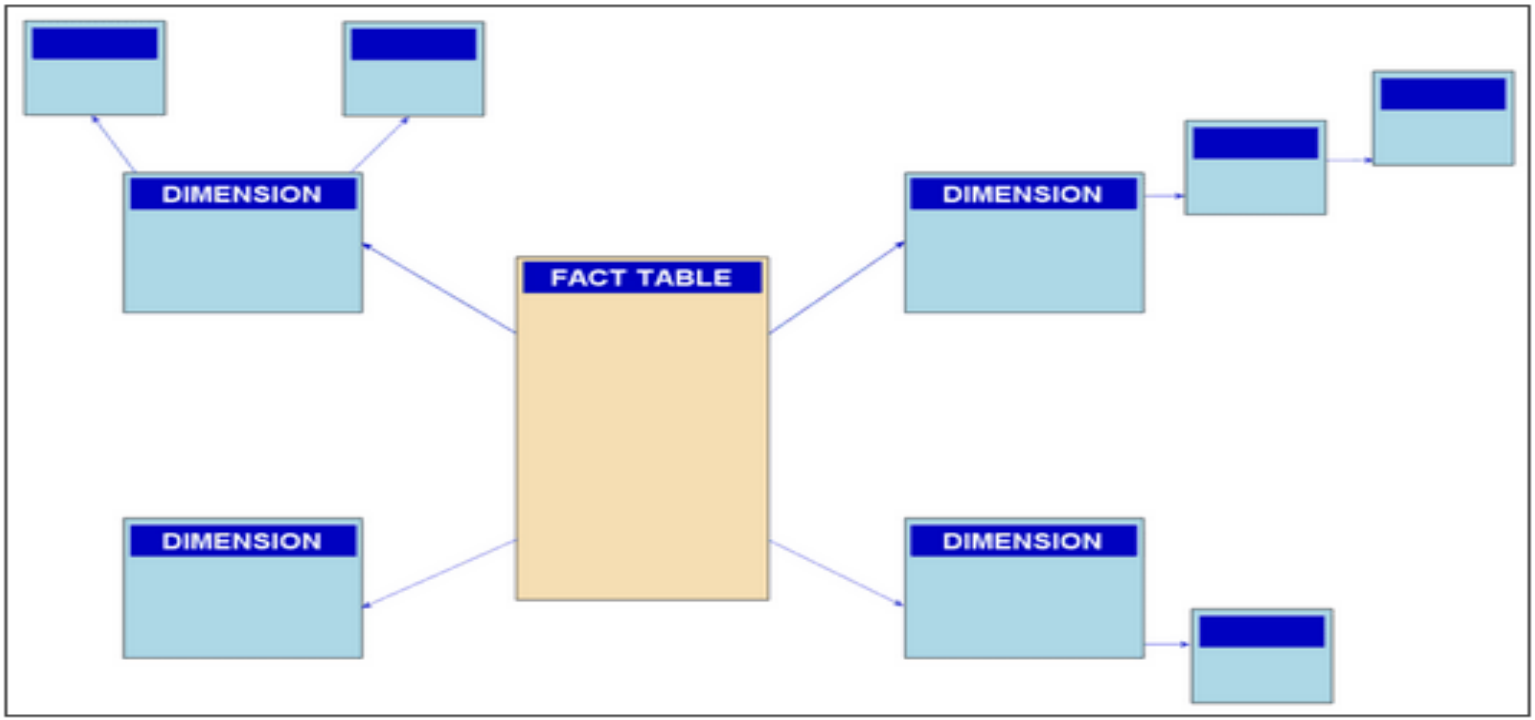


OLAP Database (Excel Data Model)

- The Excel data model design that you create in Power Pivot is an OLAP database
- OLAP database is read-only and has no need for CRUD
- Data analysis is performed using a special table and relationship arrangement called a *Star* or *Snowflake* schema that has a central *Fact* table surrounded by several *Dimension* tables
- OLAP databases used with Power Pivot cannot contain many-to-many relationships
- Because OLAP databases do not process transactions they have no need to conform to third normal form; thus reason several dimension tables are often de-normalized into a single dimension table

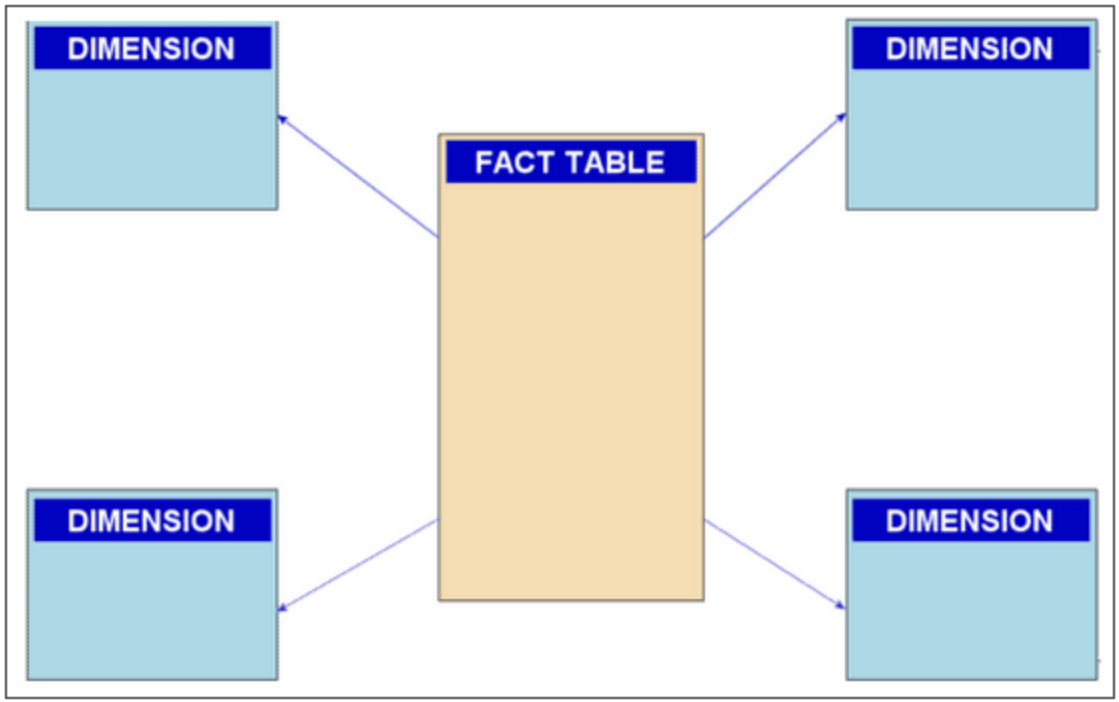
Snowflake Scheme

- Notice that some of the dimension tables have sub-dimensions giving the schema the appearance of a snowflake



OLAP Star Schema Database (best schema for Excel OLAP)

- The fact table might be the order details, and the rows might be customer, product, date

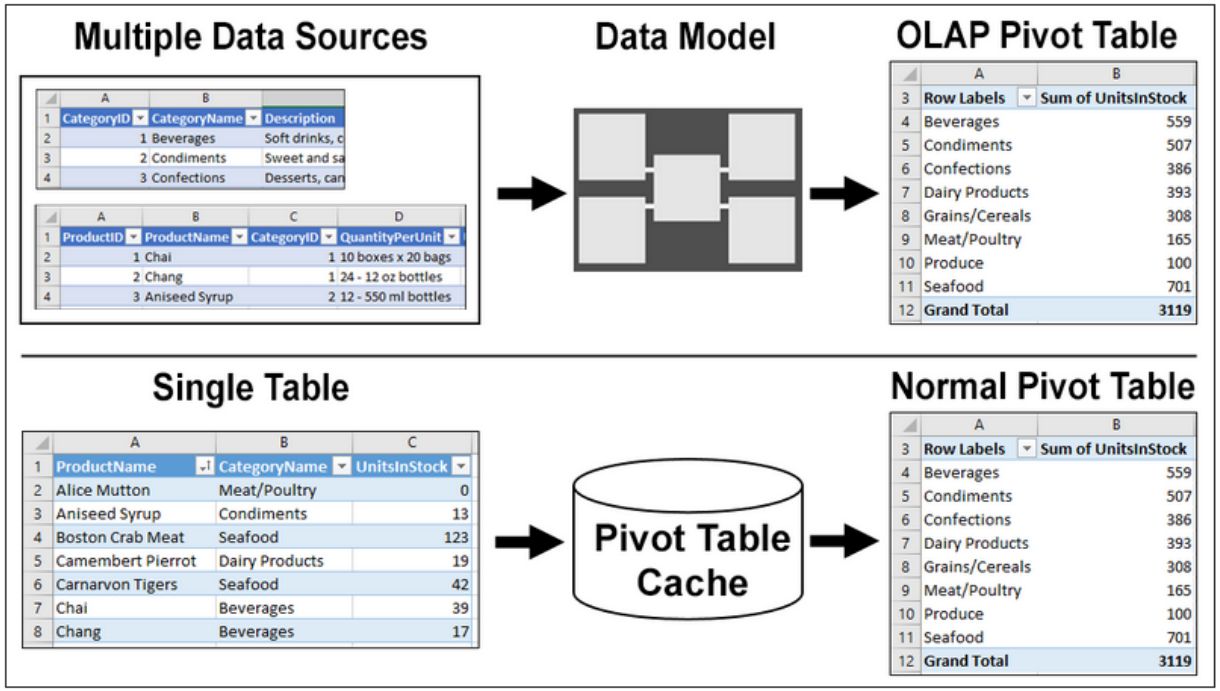


Conversion to Star Schema

- To create an OLAP star schema database from an OLTP database it is first necessary to create a snowflake schema database by converting any many-to-many relationships into one-to-many relationships
- The snowflake schema can then be converted into a star schema by de-normalizing any chains of dimension tables into single dimension tables

Normal Pivot Table vs OLAP Pivot Table

- The OLAP database (data model) is a lot simpler than the OLTP database from which it was created; it is easy to analyze data once it has been converted into a properly-constructed data model



Normal Pivot Table vs OLAP Pivot Table (con't)

- The *OLAP pivot table* obtains its data from a *Data Model* created from one or more related tables
  - The ability to select fields from more than one related table is one of the biggest advantages of working with OLAP pivot tables
- The Excel *regular pivot table* obtains its data from a pivot table data cache created from a single table residing in an Excel worksheet or external data source

MDX

- In contrast, when data resides in a data model it can be retrieved directly from the data model using an industry-standard query language called MDX (Multi-Dimensional eXpressions)
- MDX is also used by many non-Microsoft products such as *Crystal Reports*
- An OLAP pivot table automatically generates the MDX queries it needs to return values to display in the OLAP pivot table

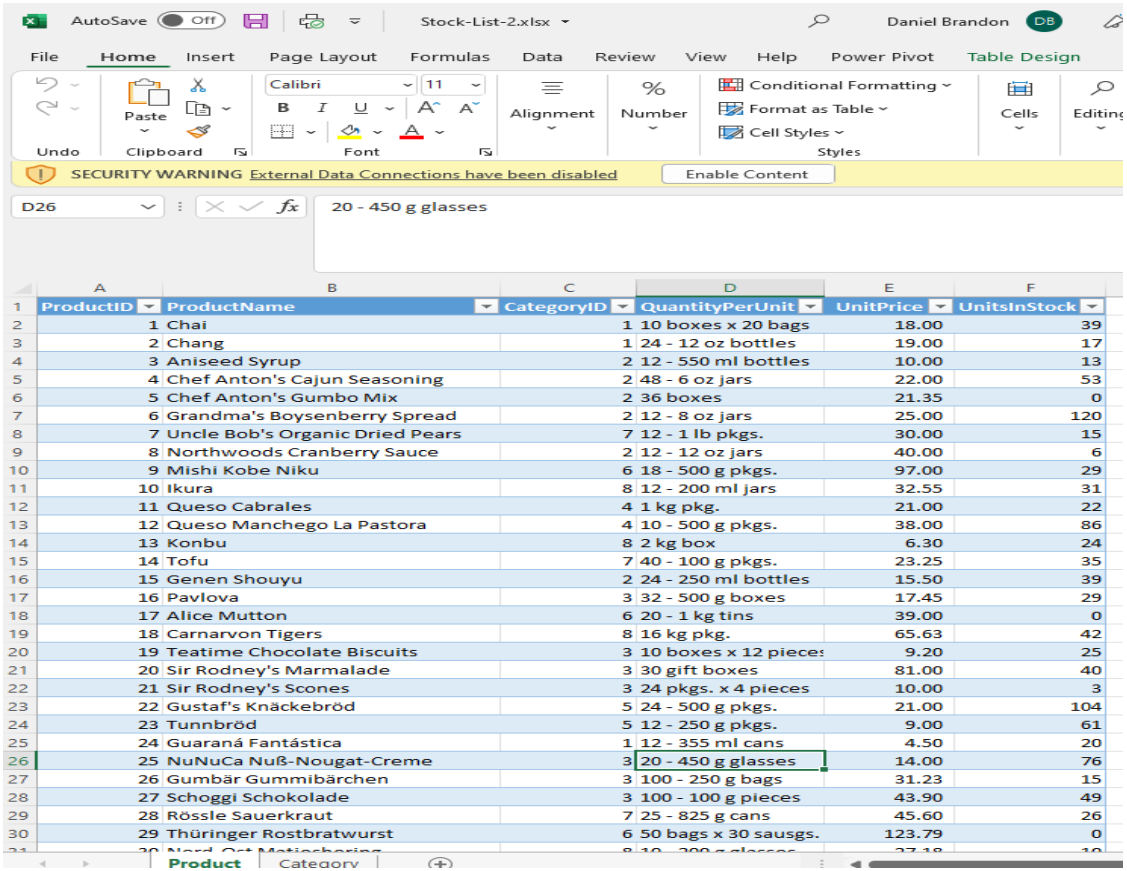
Big Data and Excel OLAP

- Excel worksheets are limited to approximately one million rows. OLAP pivot tables obtain their data from a Data Model. Data Models can contain about two thousand million data rows.
- A regular pivot table can work with big data by using Get & Transform to load data from an external non-Excel data source) directly into the *PivotTable Data Cache*
- While a regular pivot table can also overcome the million-row limitation in this way, the *PivotTable Data Cache* is still restricted to 2.1 thousand million data items (rather than rows) and is unable to work with more than one table

DAX

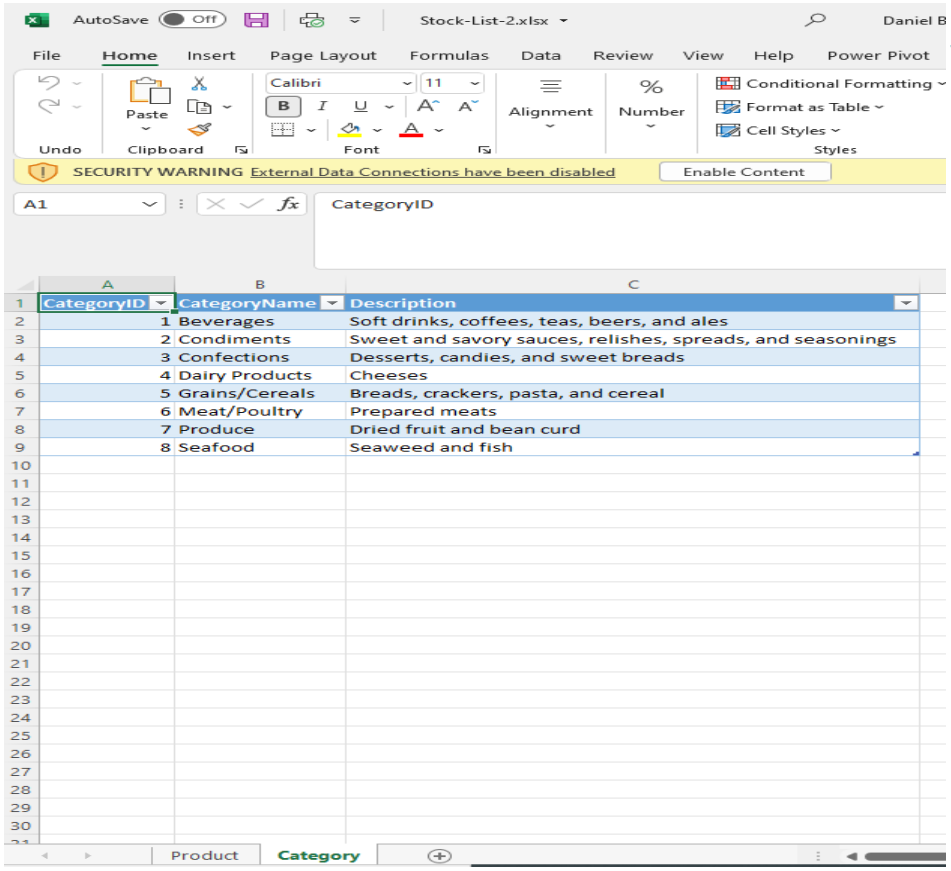
- DAX (*Data Analysis Expressions)* is a collection of over 200 functions that can be used when creating a data model
- Data models often need to have aggregate fields that are calculated from the information in the source data; this type of field (called a calculated measure) can be easily added via DAX
- While calculated measures are simple, there are many DAX-related functions that ar complex (such as *row and filter context* and *implicit and explicit measures*)

Spreadsheet Example (Product Sheet)



	A	B	C	D	E	F
1	ProductID	ProductName	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock
2	1	Chai	1	10 boxes x 20 bags	18.00	39
3	2	Chang	1	24 - 12 oz bottles	19.00	17
4	3	Aniseed Syrup	2	12 - 550 ml bottles	10.00	13
5	4	Chef Anton's Cajun Seasoning	2	48 - 6 oz jars	22.00	53
6	5	Chef Anton's Gumbo Mix	2	36 boxes	21.35	0
7	6	Grandma's Boysenberry Spread	2	12 - 8 oz jars	25.00	120
8	7	Uncle Bob's Organic Dried Pears	7	12 - 1 lb pkgs.	30.00	15
9	8	Northwoods Cranberry Sauce	2	12 - 12 oz jars	40.00	6
10	9	Mishi Kobe Niku	6	18 - 500 g pkgs.	97.00	29
11	10	Ikura	8	12 - 200 ml jars	32.55	31
12	11	Queso Cabrales	4	1 kg pkg.	21.00	22
13	12	Queso Manchego La Pastora	4	10 - 500 g pkgs.	38.00	86
14	13	Konbu	8	2 kg box	6.30	24
15	14	Tofu	7	40 - 100 g pkgs.	23.25	35
16	15	Genen Shouyu	2	24 - 250 ml bottles	15.50	39
17	16	Pavlova	3	32 - 500 g boxes	17.45	29
18	17	Alice Mutton	6	20 - 1 kg tins	39.00	0
19	18	Carnarvon Tigers	8	16 kg pkg.	65.63	42
20	19	Teatime Chocolate Biscuits	3	10 boxes x 12 pieces	9.20	25
21	20	Sir Rodney's Marmalade	3	30 gift boxes	81.00	40
22	21	Sir Rodney's Scones	3	24 pkgs. x 4 pieces	10.00	3
23	22	Gustaf's Knäckebröd	5	24 - 500 g pkgs.	21.00	104
24	23	Tunnbröd	5	12 - 250 g pkgs.	9.00	61
25	24	Guaraná Fantástica	1	12 - 355 ml cans	4.50	20
26	25	NuNuCa Nuß-Nougat-Creme	3	20 - 450 g glasses	14.00	76
27	26	Gumbär Gummibärchen	3	100 - 250 g bags	31.23	15
28	27	Schoggi Schokolade	3	100 - 100 g pieces	43.90	49
29	28	Rössle Sauerkraut	7	25 - 825 g cans	45.60	26
30	29	Thüringer Rostbratwurst	6	50 bags x 30 sausgs.	123.79	0

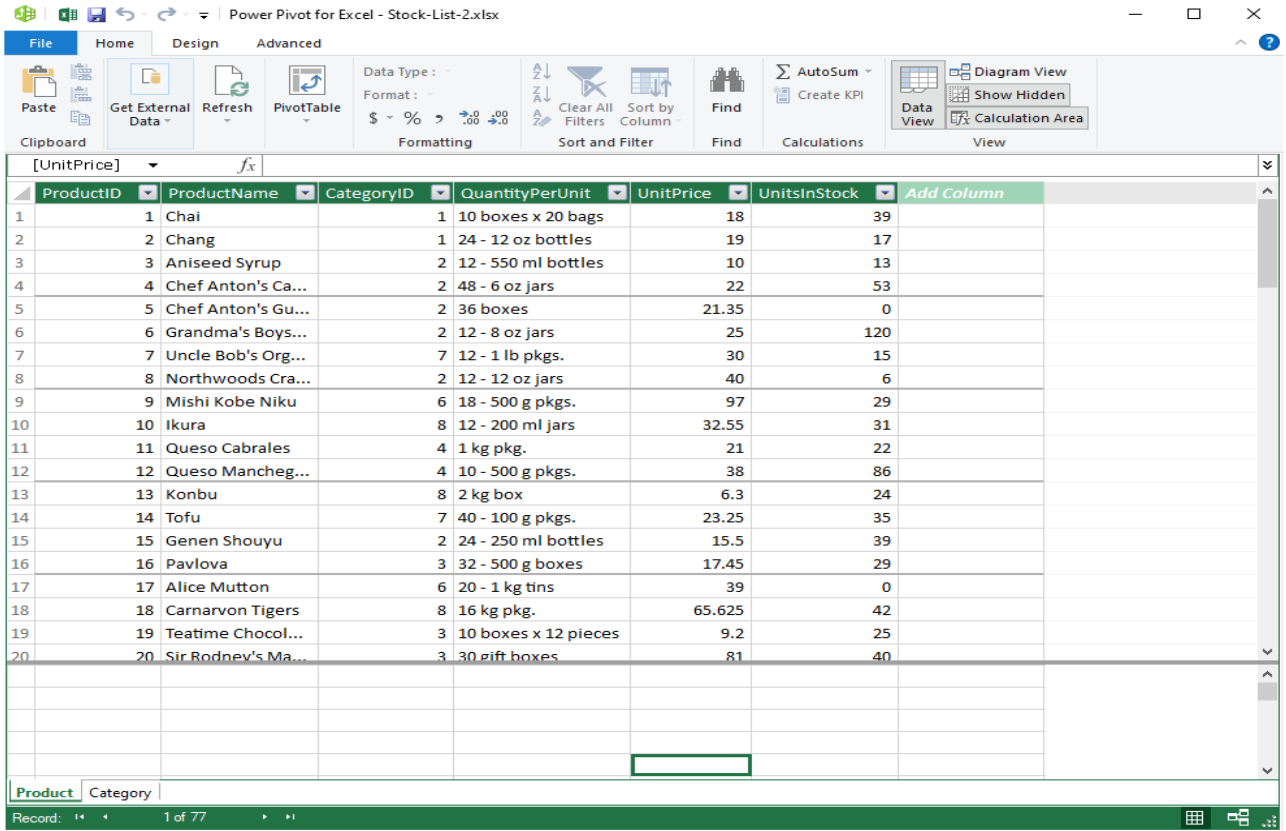
Spreadsheet Example (Category Sheet)



Power Pivot Tab

Power Pivot Window (opens in a new window)

- Power Pivot->Manage Data Model



	ProductID	ProductName	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock
1	1	Chai	1	10 boxes x 20 bags	18	39
2	2	Chang	1	24 - 12 oz bottles	19	17
3	3	Aniseed Syrup	2	12 - 550 ml bottles	10	13
4	4	Chef Anton's Ca...	2	48 - 6 oz jars	22	53
5	5	Chef Anton's Gu...	2	36 boxes	21.35	0
6	6	Grandma's Boys...	2	12 - 8 oz jars	25	120
7	7	Uncle Bob's Org...	7	12 - 1 lb pkgs.	30	15
8	8	Northwoods Cra...	2	12 - 12 oz jars	40	6
9	9	Mishi Kobe Niku	6	18 - 500 g pkgs.	97	29
10	10	Ikura	8	12 - 200 ml jars	32.55	31
11	11	Queso Cabrales	4	1 kg pkg.	21	22
12	12	Queso Mancheg...	4	10 - 500 g pkgs.	38	86
13	13	Konbu	8	2 kg box	6.3	24
14	14	Tofu	7	40 - 100 g pkgs.	23.25	35
15	15	Genen Shouyu	2	24 - 250 ml bottles	15.5	39
16	16	Pavlova	3	32 - 500 g boxes	17.45	29
17	17	Alice Mutton	6	20 - 1 kg tins	39	0
18	18	Carnarvon Tigers	8	16 kg pkg.	65.625	42
19	19	Teatime Chocol...	3	10 boxes x 12 pieces	9.2	25
20	20	Sir Rodney's Ma...	3	30 gift boxes	81	40

Power Pivot Window (con't)

- The Power Pivot window seems very similar to Excel but you are really looking at a completely different application that has been engineered to "look and feel" like Excel which is intended to make you feel instantly at home with Power Pivot
- There is a data grid that is similar to Excel with rows and columns
- You can also see tabs showing that there are two tables in the above data model (very similar to two worksheets in a workbook)

Power Pivot Window vs Excel

- **Just like Excel, you can:**
  - Rename and delete columns
  - Change column widths
  - Insert new columns
  - Click and drag columns to new locations
  - Rename and delete sheet tabs
  - Click and drag sheet tabs to re-arrange them
  - Use AutoSum to aggregate all of the values in a column; this creates a calculated measure
  - Format cells
  - Sort columns

Power Pivot Window vs Excel (con't)

- **Unlike Excel you can't:**
- Type a value into a cell or change any of the information you see displayed (although you can refresh the data from the original data source)
- Have more than one table in a sheet; a Power Pivot sheet contains a single table rather than a worksheet
- Refer to a cell using A1 notation (A1 notation means cell references such as B12 or A1:C14); Power Pivot does not recognize letters to identify columns - columns must be referred to by name
- Assign different data types within a column ; one cannot, for example, have text in some cells and numbers in other cells in the same column
- Have more than one formula in a column. When a formula is added to a Power Pivot column it always applies to every cell in that column

Power Pivot Window vs Excel (con't)

- **Some things Power Pivot can do that Excel cannot**
  - Create relationships between tables
  - Assign a larger range of data types to columns
  - Use DAX functions
  - Create calculated measures
  - Work with *Big Data.* An Excel worksheet can contain a maximum of just over a million rows. A Power Pivot table can contain a maximum of just over a thousand million rows.
  - Produce extremely fast (often perceived as instant) results even when analyzing data sets containing many millions of rows
- Power Pivot uses Microsoft's *xVelocity in-memory analysis engine*
  - This engine can scan billions of data rows per second and can produce reports in a tiny fraction of the time needed by Excel

Excel Get & Transform

- Excel Get & Transform is a re-branding of Power Query one of the three “power” tools now included in all current Excel versions
- Get & Transform’s main role is to prepare data tables for Power Pivot, enabling Power Pivot to create a data model from them
- Even if you do not use Power Pivot, Get & Transform is also very useful when importing external data into regular Excel workbooks

Get & Transform (con't)

- Excel Get & Transform is an advanced ETL tool (Extract, Transform and Load)
- Before Get & Transform was added to Excel, users had to import data into a worksheet and then transform the data inside Excel
- The Get & Transform tool enables one to import data (from one or more external sources) and then transform it before it is loaded into Excel (or a data model)
- Get & Transform can also load data into a special construct called a Data Model

Get & Transform (con't)

- **Extract:** Most business data that is analyzed with Excel doesn't start its life in an Excel workbook but is imported from an external data source (often from a database); e*xtract* simply means moving this data from the external data source into the Get & Transform tool
- **Transform:** Extracted data often isn't in a form that can be easily analyzed by Excel as there may be unwanted columns, badly named fields, badly formatted fields or corrupted data
  - The Get & Transform tool includes a vast array of features that enable one to clean data before loading it into an Excel table or Pivot Table
- **Load:** This simply means exporting the transformed data from the Get & Transform tool to its destination - a Get & Transform query can export (load) transformed data into an Excel table or into an Excel Pivot Table

Get & Transform (con't)

- Advantages in using Get & Transform to implement an ETL methodology:
  - **Re-usable and sharable queries:** The Get & Transform tool generates a re-usable *Query; t*his means that all of the actions that you define can be repeated to refresh the data in a table on your worksheet with a single click
  - **Automatically refreshed data:** You can configure a query to automatically refresh an Excel table at a timed interval
  - **The ability to transform big data:** Get & Transform does not share Excel's limits (of approximately a million rows of data) so it can be used to transform big data (sending the result directly to the data model, 3D Map, or to an Excel worksheet after aggregation)
  - **Better tools:** Get & Transform has some advanced transformation tools that are not found in the standard Excel product
  - **The ability to combine data:** Get & Transform allows one to merge queries; combine relational data from disparate data sources to create a de-normalized data extract

Data Source

Get & Transform

1. Extract

US Labor Force - by occupation:
farming, forestry, and fishing: 0.7%
manufacturing, extraction, transportation, and crafts: 20.3%
managerial, professional, and technical: 37.3%
sales and office: 24.2%
other services: 17.6%
note: figures exclude the unemployed (2009)

2. Transform

Occupation	% Workforce
farming, forestry, and fishing	0.70%
manufacturing, extraction, transportation, and crafts	20.30%
managerial, professional, and technical	37.30%
sales and office	24.20%
other services	17.60%

3. Load to one of the items below

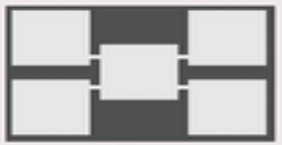
Table in Excel worksheet

	A	B
2		
3	farming, forestry, ar	0.70%
4	manufacturing, extract	20.30%
5	managerial, profession	37.30%
6	sales and office	24.20%
7	other services	17.60%

Normal Pivot Table

	A	B
3	Occupation	% of Workforce
4	farming, forestry, and fishing	1%
5	managerial, professional, and technical	37%
6	manufacturing, extraction, transportatio	20%
7	other services	18%
8	sales and office	24%
9	Grand Total	100%

Data Model



OLAP Pivot Table

	A	B
3	Occupation	% Workforce
4	farming, forestry, and fishing	1%
5	managerial, professional, and technical	37%
6	manufacturing, extraction, transportatio	20%
7	other services	18%
8	sales and office	24%
9	Grand Total	100%

Excel Modern Data Analysis

- The construct that Excel describes as an Excel Data Model is actually an OLAP database; users can use the Power Pivot tool to create an Excel Data Model design
- Excel modern data analysis has these steps:
  - Get the data
  - Transform the data
  - Relate the data
  - Summarize the data
  - Visualize the data

Get The Data

- This is done using the Get and Transform tool
- Data can be imported from a huge range of data sources including relational databases, Excel worksheets, CSV files and web pages
- Get & Transform can work with very large data sets (often called Big Data) as it is not restricted to Excel's million row limitation

Transform The Data

- This is also done using the Get and Transform tool
- Unlike Excel, the transform actions are stored in PQFL (Power Query Formula Language) expressions
- Both the connection details and the PQFL steps are stored in a query
- This means that the query can be re-run, avoiding repetitive work in the future if the source data changes

Relate The Data

- Power Pivot enables tables to be related in a data model using primary key/foreign key relationships
- This provides huge flexibility and avoids the use of any Excel functions (such as VLOOKUP) to relate data

Summarize The Data

- While a traditional pivot table can only access data residing in a single table, data models can be analyzed with a new type of pivot table (called an OLAP pivot table)
- The OLAP pivot table can do just about anything that a regular pivot table can do but also can access data residing in multiple related tables
- Power Pivot also enables DAX (Data Analysis Expressions) to be added to the data model
- DAX enables calculated columns and calculated measures (aggregations) to be simply defined

Visualize The Data

- OLAP Pivot Tables and OLAP Pivot Charts provide the primary method of visualizing data residing in data models
- It is also possible to use any of Excel's classic analysis and visualization features by using CUBEVALUE functions to extract data directly from a data model into Excel cells

Data Specialist vs Data Users