# Data Warehousing

Week 1

Data Warehousing

- Introduction to Basic DW Concepts

# Agenda

- Data Warehousing
  - What is a data warehouse (DW)?
  - How are DWs different from other data stores?
  - Why do we need DWs?
    - OLTP versus OLAP
  - Implementation Overview
    - Some *Very* General Guidelines

- Data Warehousing
  - What is a data warehouse (DW)?
  - How is it different?
  - Why do we need DWs?
    - OLTP versus OLAP
  - Implementation Overview

# DW Definitions …

- As originally proposed by **Bill Inmon** in 1990:

"A warehouse is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision making process"

-- Bill Inmon, 1995

  - Subject-oriented = particular segment of company business
  - Integrated = multiple data sources are integrated
  - Time-variant = all data is for a particular, identified point in time
  - Non-volatile = data is stable; added but not removed

# DW Definitions

- **Ralph Kimball** has a much simpler – but somewhat *less* informative – definition of a data warehouse:

A Data Warehouse is "a copy of transaction data specifically structured for query and analysis."

-- Ralph Kimball, 1996

A Data Warehouse is "nothing more than the union of all the constituent data marts."

-- Ralph Kimball, 1998

# DW Vendors

- Computer Associates International (Ingres)
- IBM Corporation (DB2, IDS, UDB)
- Hitachi Data Systems (Nucleus Exploration Systems)
- Informatica Corporation (PowerCenter, PowerMart)
- Microsoft Corporation (SQL Server)
- Oracle Corporation (Oracle Database)
- SAS Institute (SAS ETL Studio)
- Sybase (Adaptive Server Anywhere, Sybase IQ)
- Torrent Systems (Orchestrate)
- Vanguard Solutions Group (Graphical Performance Series)
- etc.

- Data Warehousing
  - What is a data warehouse (DW)?
  - How is it different?
  - Why do we need DWs?
    - OLTP versus OLAP
  - Implementation Overview

# A *Different* Design …

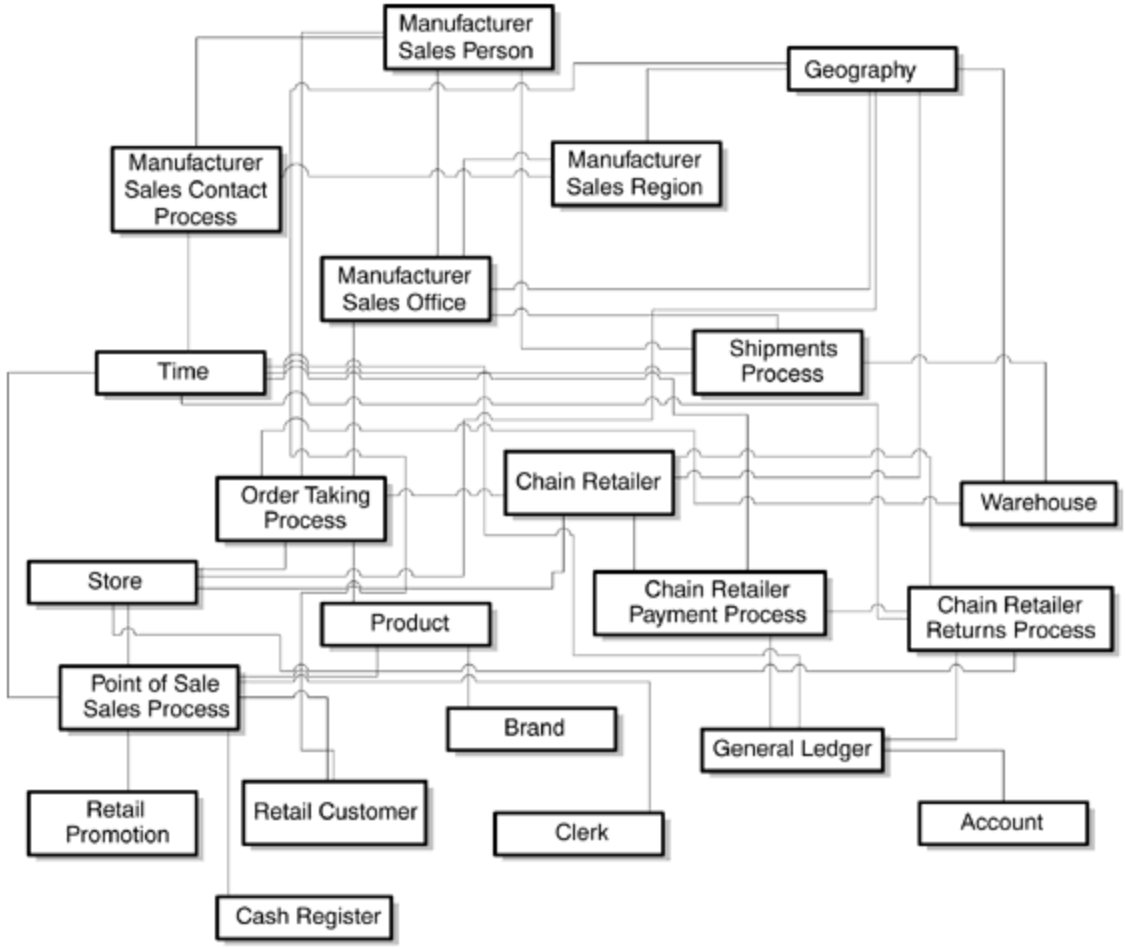
- A data warehouse is a *computing environment* that is built from a *set* of technologies – not just one specific software or hardware product. It is:
  - **User-centric and user-driven**
  - **Designed for fast and flexible interactive access**
  - Supports "ask-answer-ask-again" data analysis patterns
  - Conducive to answering complex questions
  - Facilitates discovering unanticipated results

    and provides strategic information on an *enterprise-wide* scale.
  - Data analysis for decision support

# A *Different* Design …

- Operational System
  - Day-to-day operational systems (OLTP)
    - Detail-level data
  - Insert, update, delete, query
  - Update daily; dynamic
  - Large amounts of data
  - ER modeling (normalized)
    - Complex; understood by DB professionals
  - Implement via DBMS with front-end applications
- Data Warehouse
  - Decision-making and data investigation (OLAP)
    - Detail-level & summary data
  - Insert, query
  - Update periodically; ~static
  - Massive amounts of data
  - ER/Dimensional modeling
    - Straight forward; understood by the end users
  - Implement via DBMS with front-end analysis SW

# ER Model – High Level Entities

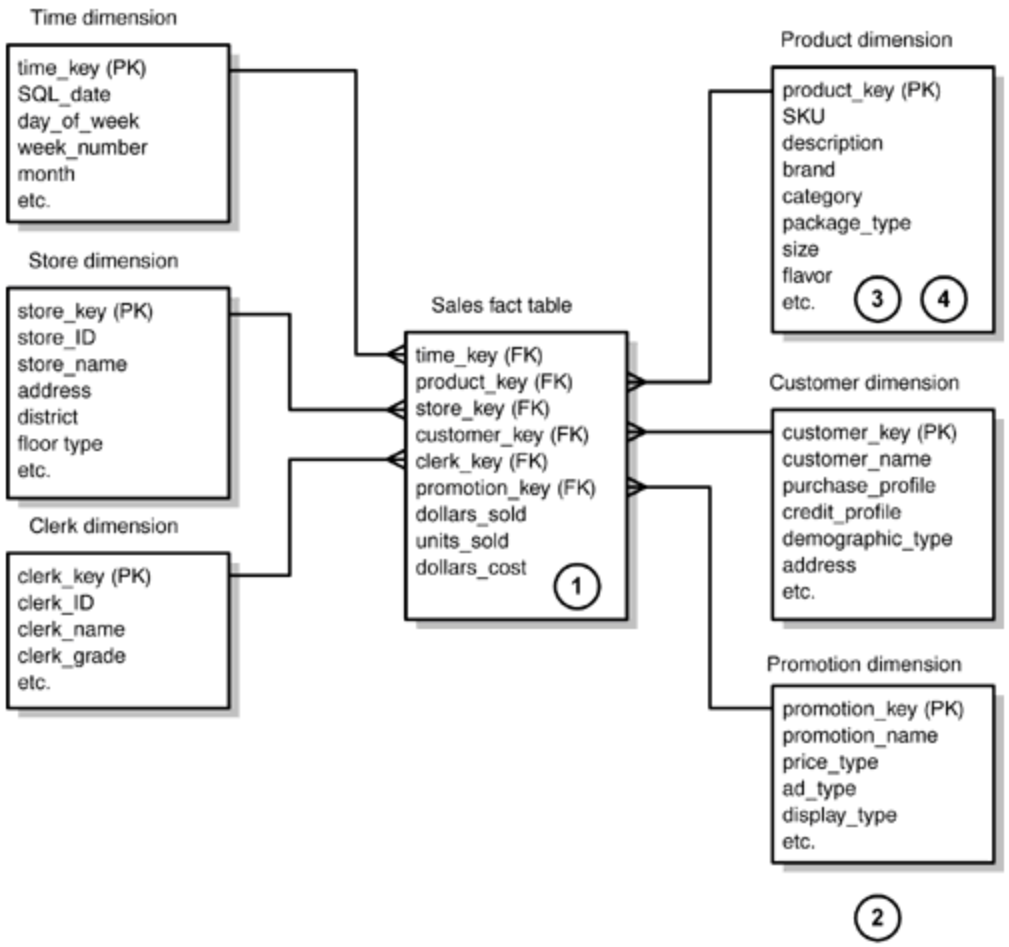



An entity-relationship model of an enterprise that manufactures products, sells products to chain retailers, and measures the retailers' sales.

# What can you remember?

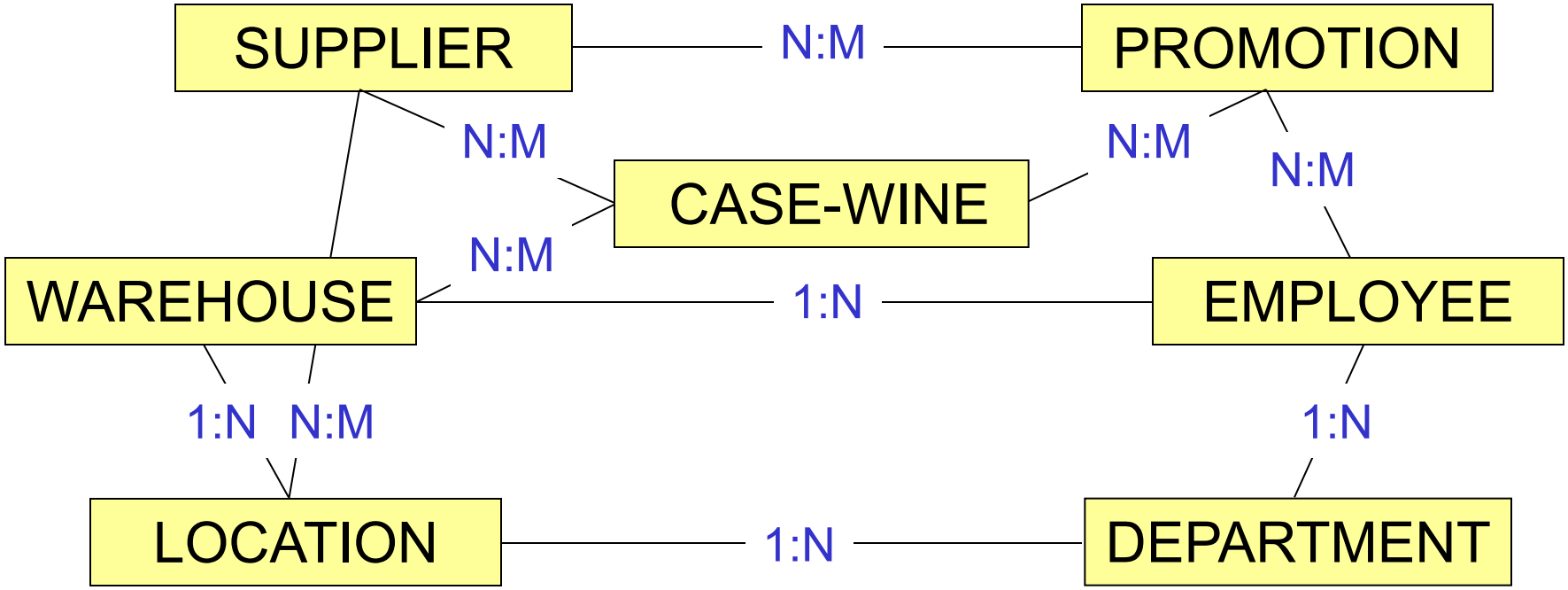
- Once the ER model (previous slide) is hidden, what do you remember about the entities?
- What about the next slide?

# Dimensional Model



# ER vs. Dimensional Modeling …

- Example: **NoW*h*ine Distributors, Inc.**, a wine distributor



# ER vs. Dimensional Modeling …

- Let's look at a subset: the SUPPLIER-WINE relationship:


- supplierID
- supplierName
- city, etc.

- wineID
- wineName
- year, etc

  - SUPPLIER represents the companies or organizations that sell or broker wine from individual vineyards.
  - CASE-WINE represents case lots of wine of various types: Zinfandel, Chablis, Merlot, etc that have been purchased.

# ER vs. Dimensional Modeling …

- Suppose we're interested in analyzing who our top suppliers within various categories of wine have been over the last five years so that we can improve our business relationships with them …
  - Will the day-to-day operational database for NoW$_h$ine support this?

    Why or why not?

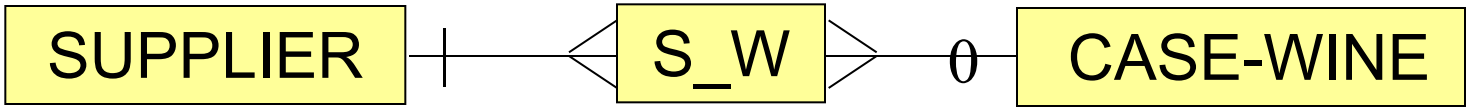
    *How is "top" defined?*

    *What time period?*

    *Wine "categories?"*

# ER vs. Dimensional Modeling …

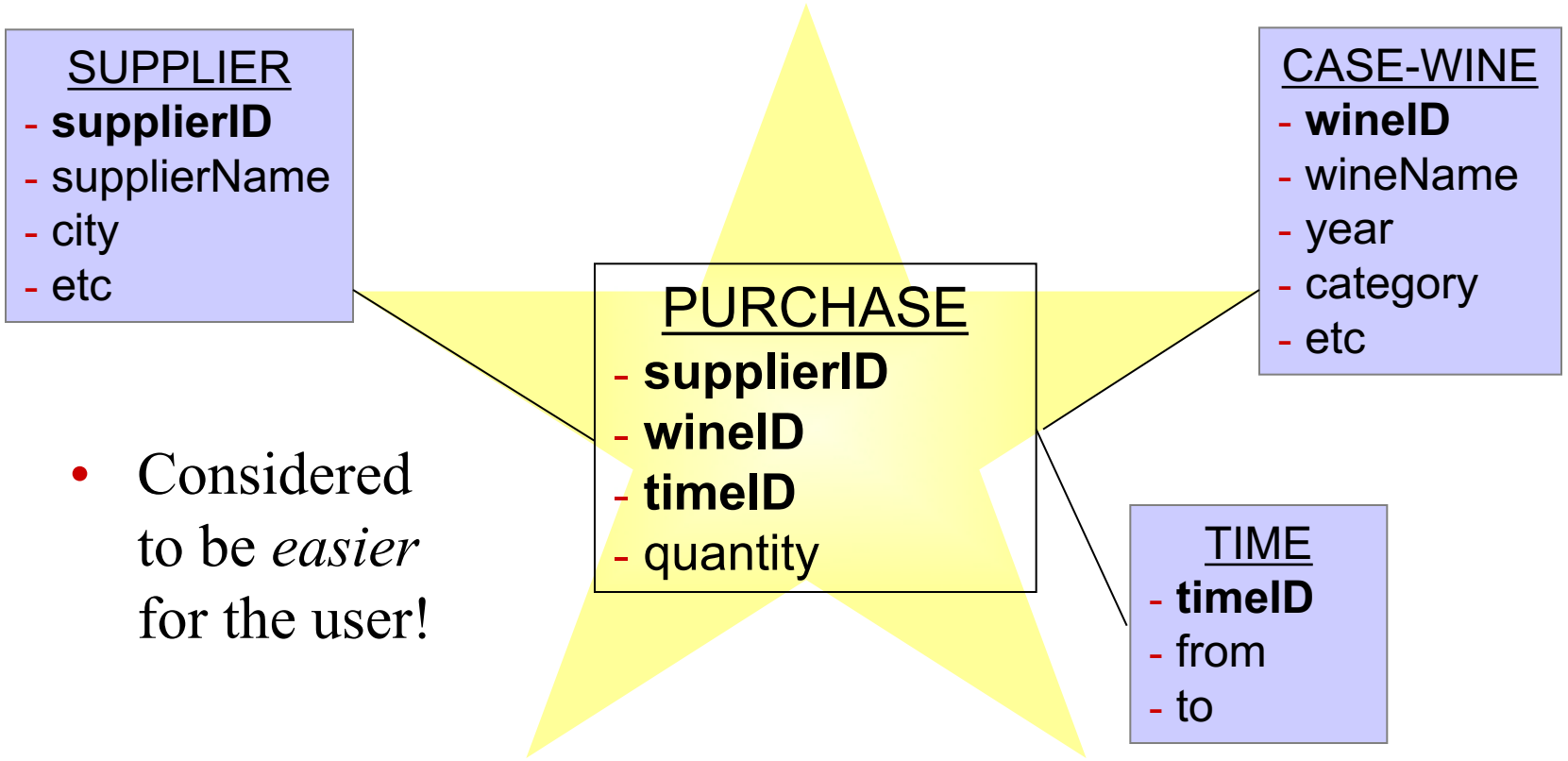
- Traditional (relational) physical design for an N:M relationship:




| SupplierID | WineID | Quantity | Time |
|---|---|---|---|
| S1 | W1 | 200 | T1 |
| S1 | W2 | 150 | T2 |
| S2 | W2 | 500 | T3 |
| … | … | … | … |

# Dimensional Modeling

- Star schema for the SUPPLIER_WINE relationship:



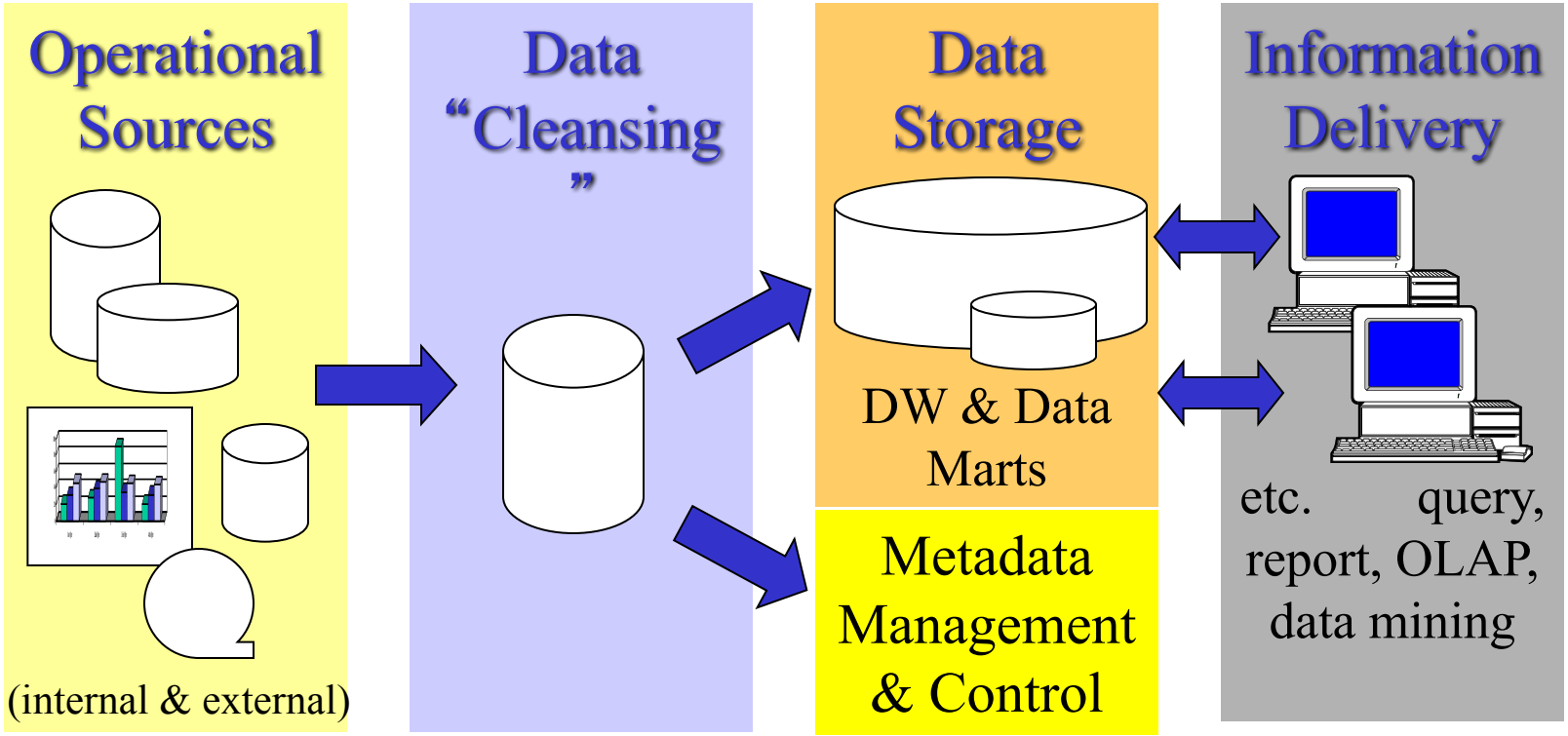

- Considered to be *easier* for the user!

# A *Different* Design

## ~ System Component Comparison ~

- Operational System
  - **Front-end**
    - GUI
  - **Connectivity**
    - Data interfaces
    - Network software
  - **Data Storage**
    - DBMS
  - **Display**
    - Screens, reports
- Data warehouse
  - **Source Data**
    - Production, internal, archived, external
  - **Data Staging**
    - Extract, transform, load
  - **Data Storage**
    - DBMS
  - **Information Delivery**
    - Reports, queries, statistical analyses

# DW Components




- Overview of the components of a data warehousing system.

# DW versus Data Mart

- Def. data mart = a specialized, subject-oriented, integrated, non-volatile, time-variant data store designed to support a specific subset of management decisions. *-- C. J. Date, 2004*
- Why? In actual practice, extensive analysis and reporting is typically only done on a subset of the data in a DW and at specific times.
  - Used for more "specific" analysis such as forecasting, simulation, and "what-if" modeling.
- Created:
  - Extract data from an existing data warehouse (dependent)
  - Create independently from an existing data warehouse (independent)
  - Create first or instead of a data warehouse (independent)

- Data Warehousing
  - What is a data warehouse (DW)?
  - How is it different?
  - Why do we need DWs?
    - OLTP versus OLAP
  - Implementation Overview

# Why do we need DW?

- "We have mountains of data in this company, but we can't access it."
- "We need to slice and dice the data every which way."
- "You've got to make it easy for business people to get at the data directly."
- "Just show me what is important."
- "It drives me crazy to have two people present the same business metrics at a meeting, but with different numbers."
- "We want people to use information to support more fact-based decision making."

# Examples of heterogeneous data

STUDENT_DATA

| Student_No | Last_Name | MI | First_Name | Telephone | Status | • • • |
|---|---|---|---|---|---|---|
| 123-45-6789 | Enright | T | Mark | 483-1967 | Soph | |
| 389-21-4062 | Smith | R | Elaine | 283-4195 | Jr | |

STUDENT_EMPLOYEE

| Student_ID | Address | Dept | Hours | • • • |
|---|---|---|---|---|
| 123-45-6789 | 1218 Elk Drive, Phoenix, AZ 91304 | Soc | 8 | |
| 389-21-4062 | 134 Mesa Road, Tempe, AZ 90142 | Math | 10 | |

STUDENT_HEALTH

| Name | Telephone | Insurance | ID | • • • |
|---|---|---|---|---|
| Mark T. Enright | 483-1967 | Blue Cross | 123-45-6789 | |
| Elaine R. Smith | 555-7828 | ? | 389-21-4062 | |

# History: Support for Decision-Making

- Businesses and organizations need strategic information upon which to base decisions.
- Timeline:
  - 1950's & 1960's – Electronic Computing Systems = store transactional data and generate pre-defined "activity" reports.
    - Ad hoc reports (special, one-time programs), special extracts (periodic data with special programs), small applications (user-controlled parameters against extracts; early online capability)
  - 1970's – early Decision Support Systems/Information Centers = pre-defined customizable applications and reports (printed & online) available with support from a computing professional.
    - Often mainframe-based; typically used for marketing

# History (continued)

- Timeline (continued):
  - 1980's – Executive Information Systems = key strategic information and simple reports available on the executive desktop in a easy to use format.
    - Distributed computing, early expert systems (subject-focused)

    → Finally achieve first "true" decision-support systems
  - 1990's – Strategic Information Systems = business decision-making support includes the ability to explore "what if" scenarios.
    - Commercialization of the internet; growth of intranets
    - Data warehouses

# Operational Data Systems

- Def. operational systems = computing systems that provide the information necessary to run day-to-day business activities.
  - Data records individual event-driven, "business" transactions.
    - Snapshots of events – i.e. a sale at a given point in time
  - Large quantities of data.
  - Data resides on different systems, across multiple platforms, and in diverse and incompatible data structures.
- These are OLTP (online transaction processing) systems intended for basic, day-to-day business processes.
  - Put transaction data *into* operational databases.

# Decision-Support Systems …

- Def. decision-support systems (DSS) = computing systems that provide vital strategic information for effective decision-making.
  - In-depth, up-to-date data that encompass the *entire* organization.
    - Data is *integrated* from all operational information systems
  - Used to:
    - Watch business processes, establish goals, set objectives, define business strategies, and monitor results
    - Keep a business/organization focused on-task and competitive.

# Decision-Support Systems

- Characteristics of *good* strategic information:
  - Integrated - a single, enterprise-wide viewpoint
  - Data integrity - data is correct and accurately represents business rules
  - Credible - single-source values
  - Accessible - formatted for ease of access; flexible for intuitive and investigational analysis
  - Timely - up-to-date information is available when needed

→ These characteristics are not directly obtainable from daily operational data.

# DSS & OLAP

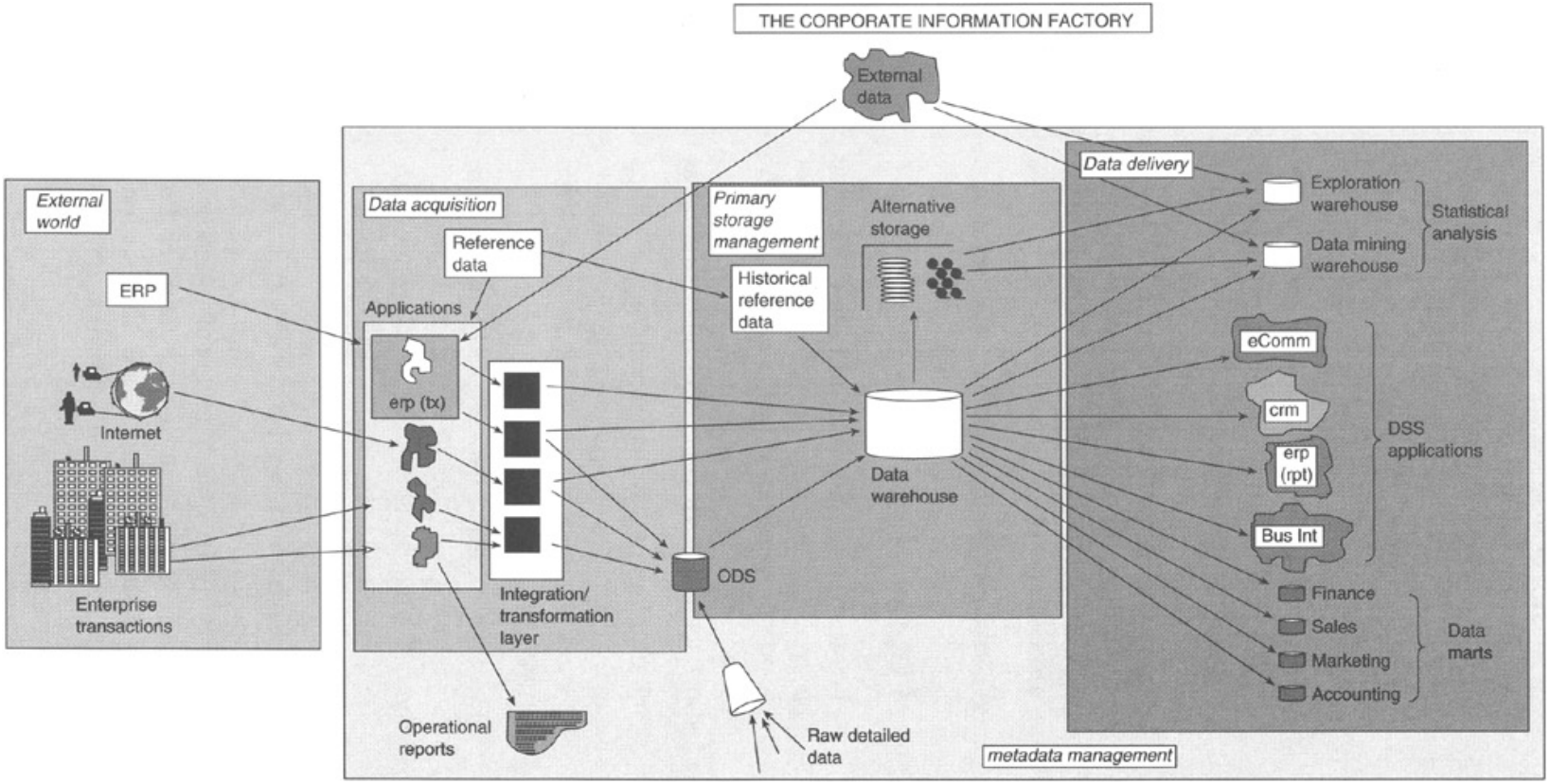
- DSS support OLAP (online analytical processing) systems.
  - Originally proposed by E. F. Codd (1993)
- Per the, now defunct, **OLAP Council**:
  - Designed for decision-makers: analysts, managers, and other executives.
  - Software that is designed to support fast, consistent, and interactive multidimensional analysis and investigation of strategic information.
  - Get insight into the data from the viewpoint of the user.
- Per the **OLAP Report** (founded 1994), OLAP is fast analysis of shared multi-dimensional information (FASMI).

# Operational Systems versus DSS

- Operational Systems
  - Store current data
  - optimized for transactions
  - Has high access frequency
  - Handles reads, inserts, updates, and deletes
  - Predictable and repetitive usage patterns
  - Sub-second response times
  - Support large numbers of users
- Decision Support Systems
  - Data is archived, derived, and summarized
  - Optimized for complex queries
  - Medium to low access frequency
  - Read only
  - Ad hoc, random or heuristic usage
  - Responds in seconds to minutes
  - Small number of users
    - Often highly skilled

- Data Warehousing
  - What is a data warehouse (DW)?
  - How is it different?
  - Why do we need DWs?
    - OLTP versus OLAP
  - Implementation Overview

# Figure 2.1: The basic structure of the corporate information factory.
# http://www.inmoncif.com/




Corporate Information Factory, Second Edition
by W. H. Inmon, Claudia Imhoff and Ryan Sousa
John Wiley & Sons © 2001

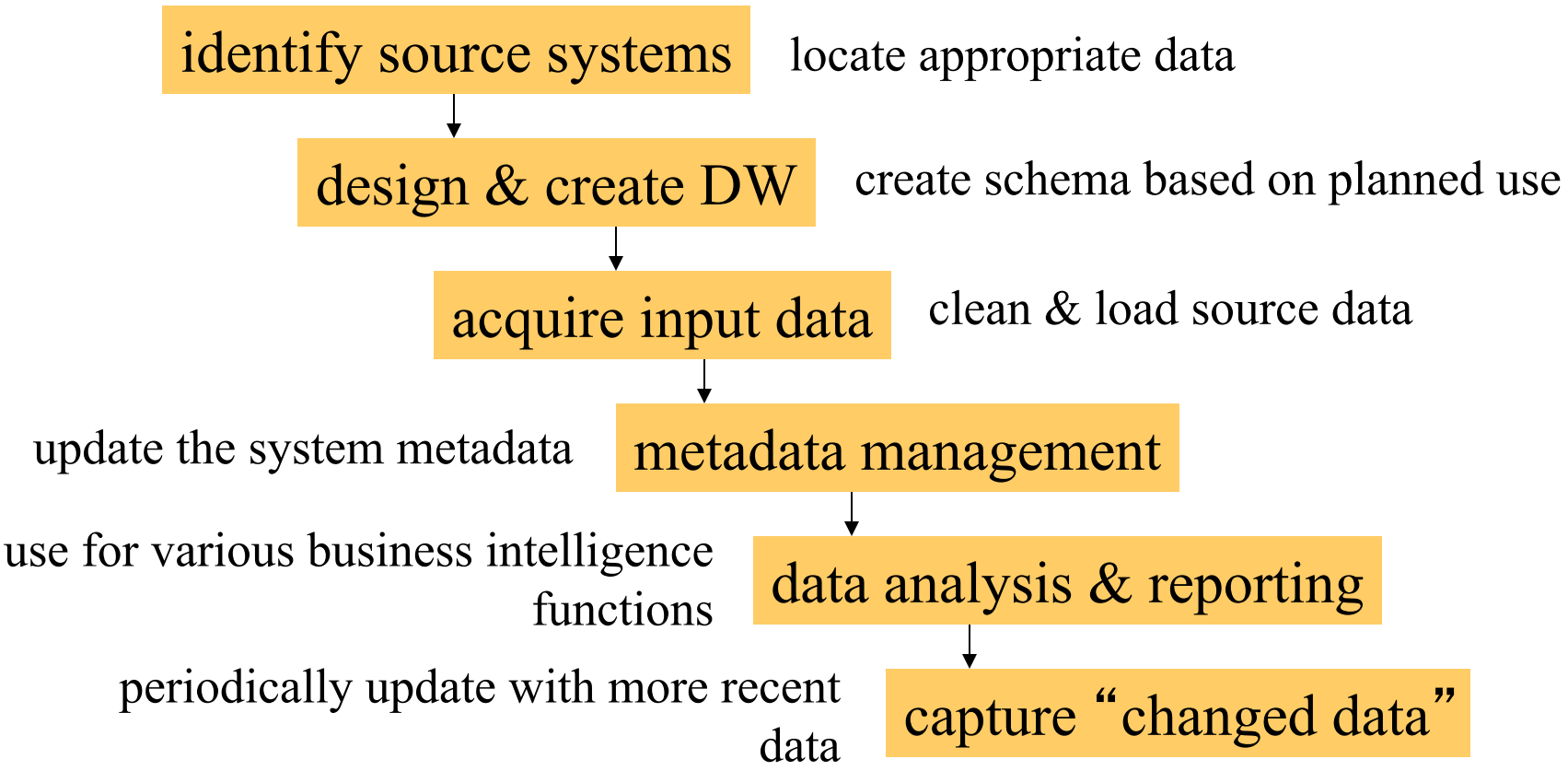
# Data Warehousing

- Def. **Data Warehousing** = the activities needed to create, maintain, and utilize a data warehouse or data mart.

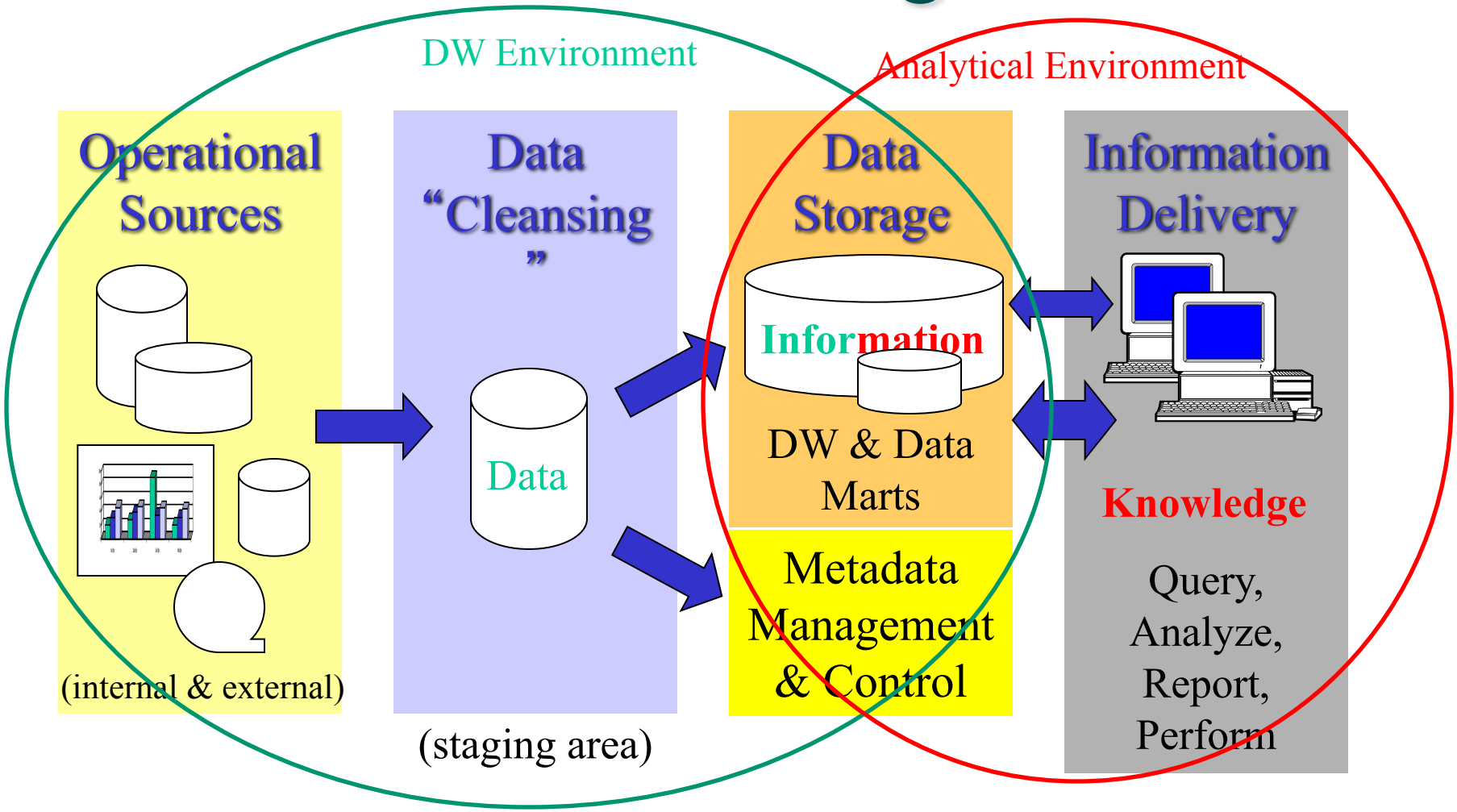
- Specifically, the processes of:
  - Creating
  - Populating
  - Querying

  a data warehouse.
- Can involve multiple technologies and intermediary steps.

# Basic DW'ing Life Cycle

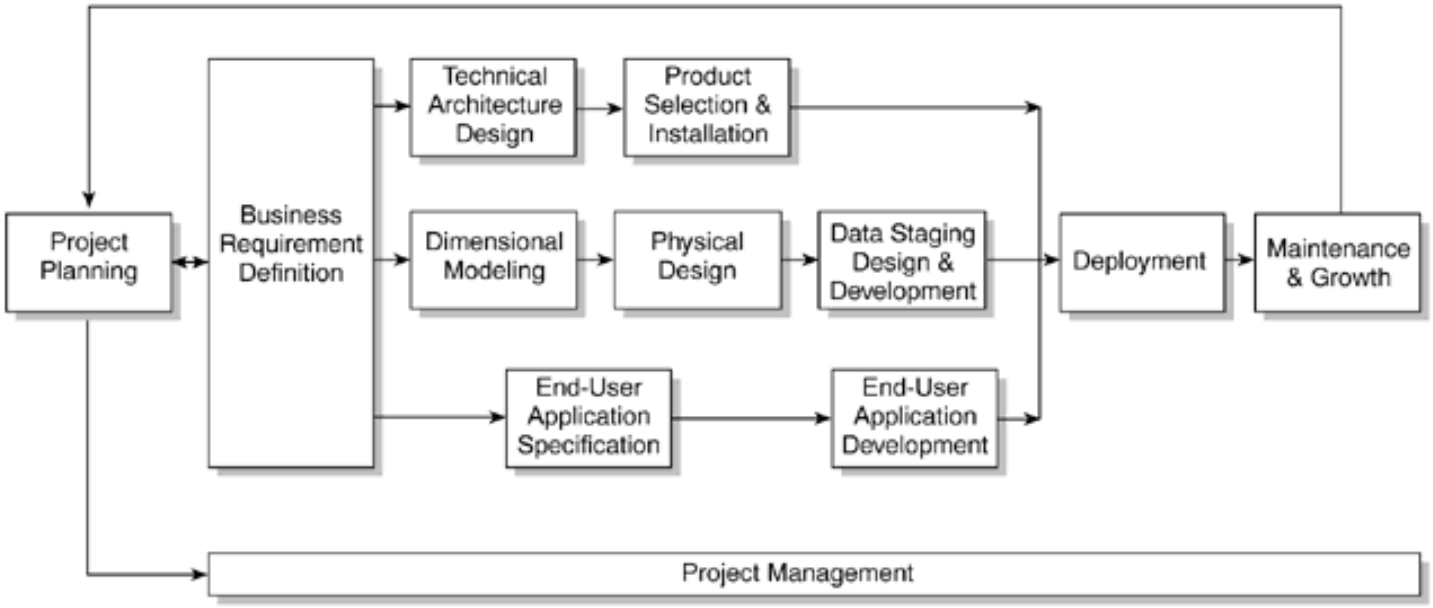


# Business Intelligence



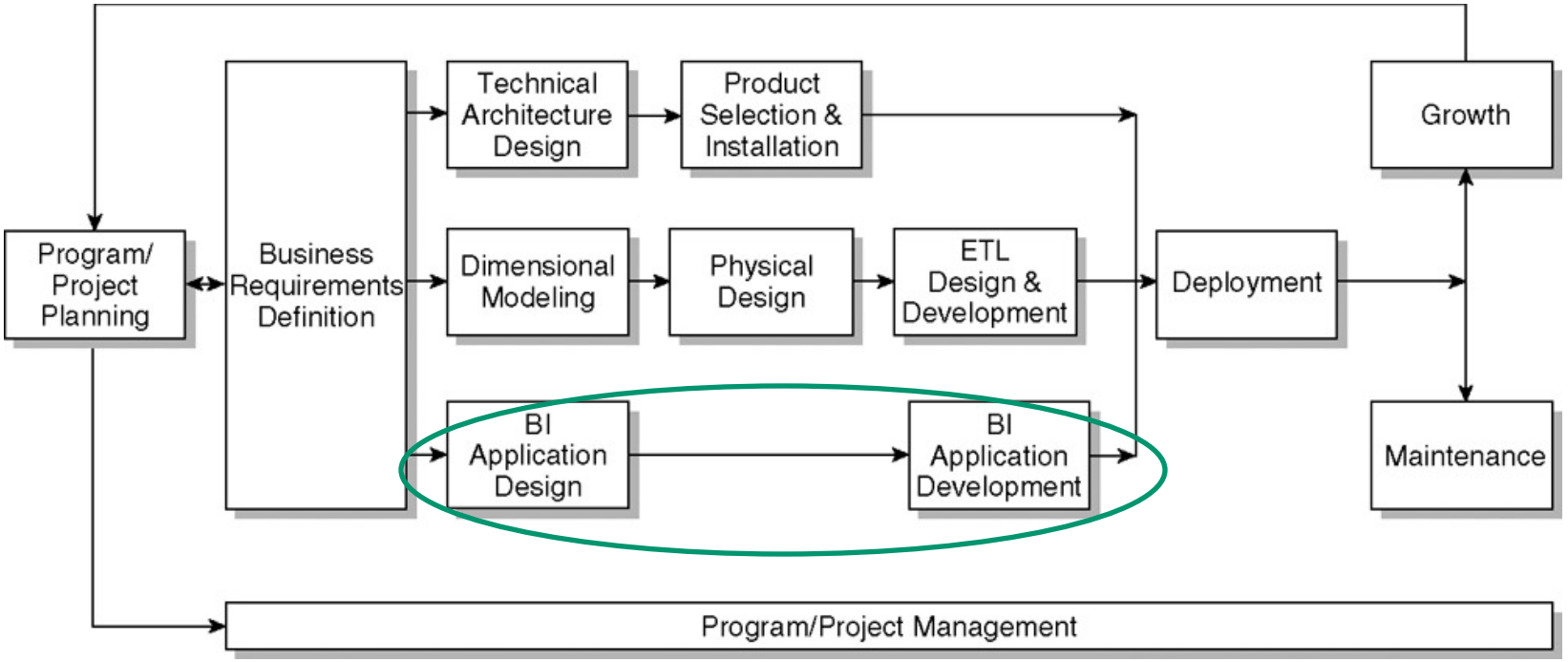
getcompa

# Figure 2.1 The Business Dimensional Lifecycle diagram.




The Data Warehouse Lifecycle Toolkit:
by Ralph Kimball, Laura Reeves, Margy Ross and Warren Thornthwaite
John Wiley & Sons © 1998

# Figure 1-1: The Kimball Lifecycle diagram (http://www.kimballgroup.com/ )




The Data Warehouse Lifecycle Toolkit, Second Edition by Ralph Kimball et al. John Wiley & Sons © 2008

# Some Very General Guidelines

- **#1: Build it; they *will* come.**
  - Don't just create a DW in the hopes that it will be useful. You need buy-in from the target users and management from the beginning. A DW must be built from a clear set of business objectives and a documented set of user requirements.
- **#2: Define the Architectural Framework *in Advance*.**
  - Define a comprehensive framework, or architecture, for a DW *before* beginning implementation. A DW project can only be successful if all user needs and business realities are addressed from the beginning.

# General Guidelines …

- **#3: Document *All* Assumptions.**
  - Define all user assumptions and possible data issues early in the requirements gathering phase to make sure that they are reflected in the final implementation. A DW project can only be successful if all key issues are addressed from the beginning.
- **#4: Use the *Right* Tools for the Job.**
  - DW technologies are typically different from those used for most other standard application development projects. You will need tools specifically for data analysis, development, implementation, and delivery in a DW environment.

# General Guidelines …

- **#5: *Understand* the DW Life Cycle.**
  - The life cycle for a DW (DWLC) is different from the traditional application development life cycle (SDLC). Data warehousing projects tend to continue as long as they remain a viable source of decision-making information.
- **#6: *Expect* Data Problems.**
  - Allow enough time and resources to deal with the issues that typically arise when collecting and cleansing data for a DW. You will need to work with multiple data owners and resolve data conflicts. Expect the data analysis necessary to identify the best data sources and the data preparation process can both be tedious and time consuming.

# General Guidelines …

- **#7: *Learn* from Mistakes.**
  - Once initiated, DW projects are ongoing and tend to expand over time. Therefore, be sure to carefully document any mistakes that occur. Learning from past mistakes will mean better quality DW implementations in future.

# Conclusion …

- Data warehousing is a *process* that utilizes multiple technologies.
- Data warehousing is *different* from traditional database utilization.
  - Currently, data warehouses are built separately from traditional databases – i.e. are separate integrated, analytical systems.
- Data warehouses are built and tuned to process many *different* queries as efficiently as possible.
  - Quick retrieval when the access path is not known in advance.

# Conclusion

- Data warehouses are integrated systems designed to facilitate user analysis and decision-making.
- They provide a single, clean, consistent source of data for analysis and decision-making.
  - Separate from operational systems.
- Data warehouses produce the strategic information that is *vital* for successful businesses!

**So … Let's investigate more!**

# References


- "Modern Data Warehousing, Mining, and Visualization – Core Concepts", George M. Marakas, Prentice Hall, 2002.
- "Data Warehousing Fundamentals, a Comprehensive Guide for IT Professionals", Paulraj Ponniah, John Wiley & Sons, 2nd Edition 2010.
- C. J. Date, "An Introduction to Database Systems," Pearson Education/Addison Wesley, 8th Edition, 2004, Chapter 22.
- "The Data Warehouse Lifecycle Toolkit", Kimball, et al, John Wiley & Sons, 1998 & 2008.
- "Corporate Information Factory", 2nd Edition, Inmon, et al, John Wiley & Sons, 2001.
- http://www.dwinfocenter.org
- http://www.intranetjournal.com/features/datawarehousing.html
- http://www.inmoncif.com/
- http://www.ralphkimball.com/