NATIONAL ENGINEERING CENTER
University of the Philippines
Diliman, Quezon City

2.0 Introduction to Data Warehousing

Eugene Rex L. Jalao, Ph.D.

Associate Professor

Department Industrial Engineering and Operations Research

University of the Philippines Diliman

@thephdataminer

Module 1 of the Business Intelligence and Analytics Track of UP NEC and the UP Center of Business Intelligence

Module 1 Outline

1. Intro to Business Intelligence
   - Case Study on Selecting BI Projects
2. **Data Warehousing**
   - **Case Study on Data Extraction and Report Generation**
3. Descriptive Analytics
   - Case Study on Data Analysis
4. Visualization
   - Case Study on Dashboard Design
5. Classification Analysis
   - Case Study on Classification Analysis
6. Regression and Time Series Analysis
   - Case Study on Regression and Time Series Analysis
7. Unsupervised Learning and Modern Data Mining
   - Case Study on Text Mining
8. Optimization for BI

eljalao@up.edu.ph

Outline for This Session

- Intro to Data Warehousing
- Kimball DW Lifecycle
- Dimensional Model vs Normalized Models
- ETL Overview
- Case Study

Recall Our Basic Framework

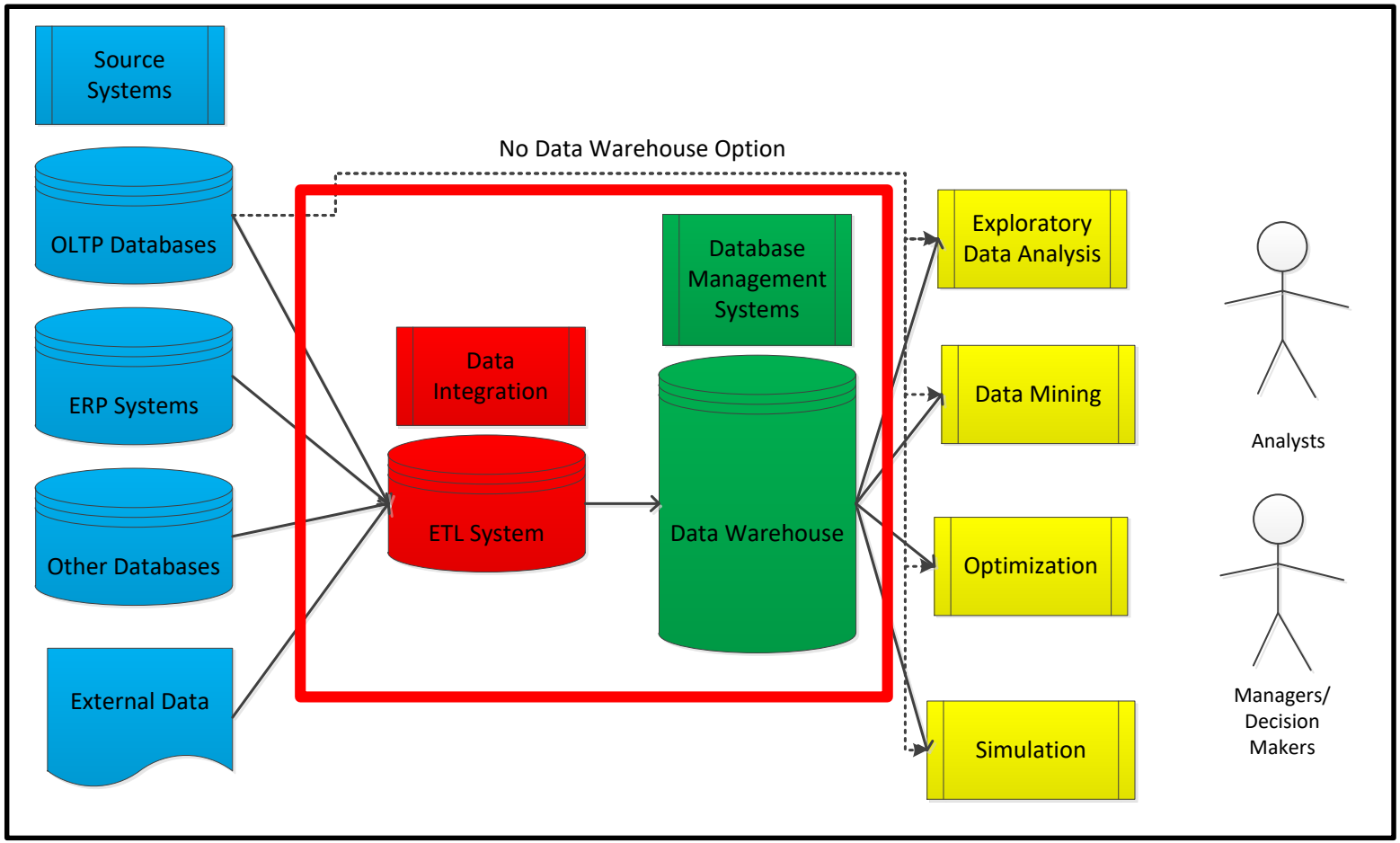

Figure 2.1: BA Framework

eljalao@up.edu.ph

Intro to Data Warehousing

Definition 2.1: Data Warehouse

- A **physical repository** where relational data are specially organized to provide enterprise-wide, cleansed data in a standardized format
- "The data warehouse is a collection of **integrated**, **subject-oriented databases** designed to support DSS functions, where each unit of data is **non-volatile** and **relevant** to some moment in time"

Intro to Data Warehousing

- Some Characteristics of a DW
  - Subject oriented
  - Integrated
  - Time-variant (time series)
  - Nonvolatile
  - Summarized
  - Not normalized (usually)
  - Metadata
  - Web based, relational/multi-dimensional
  - Real-time and/or right-time (sometimes)

Intro to Data Warehousing

Definition 2.2: Data Mart

- A departmental data warehouse that stores only relevant data
  - **Dependent data mart**
    - A subset that is created directly from a data warehouse
  - **Independent data mart**
    - A small data warehouse designed for a strategic business unit or a department

Intro to Data Warehousing

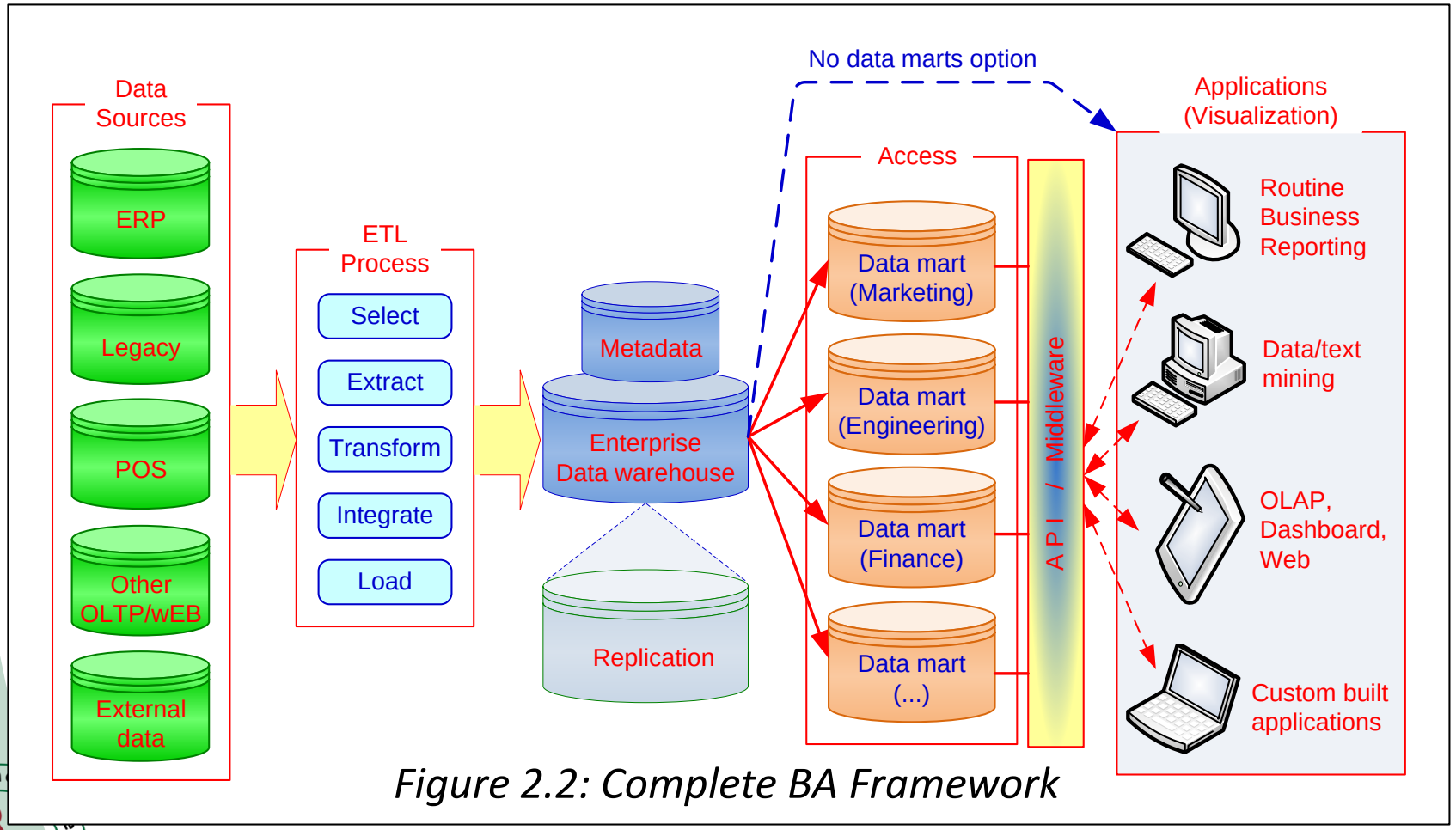

Figure 2.2: Complete BA Framework

Intro to Data Warehousing

- Alternative DW Architectures
  - Independent Data Marts
  - Data Mart Bus Architecture
  - Hub-and-Spoke Architecture
  - Centralized Data Warehouse
  - Federated Data Warehouse

eljalao@up.edu.ph

Intro to Data Warehousing

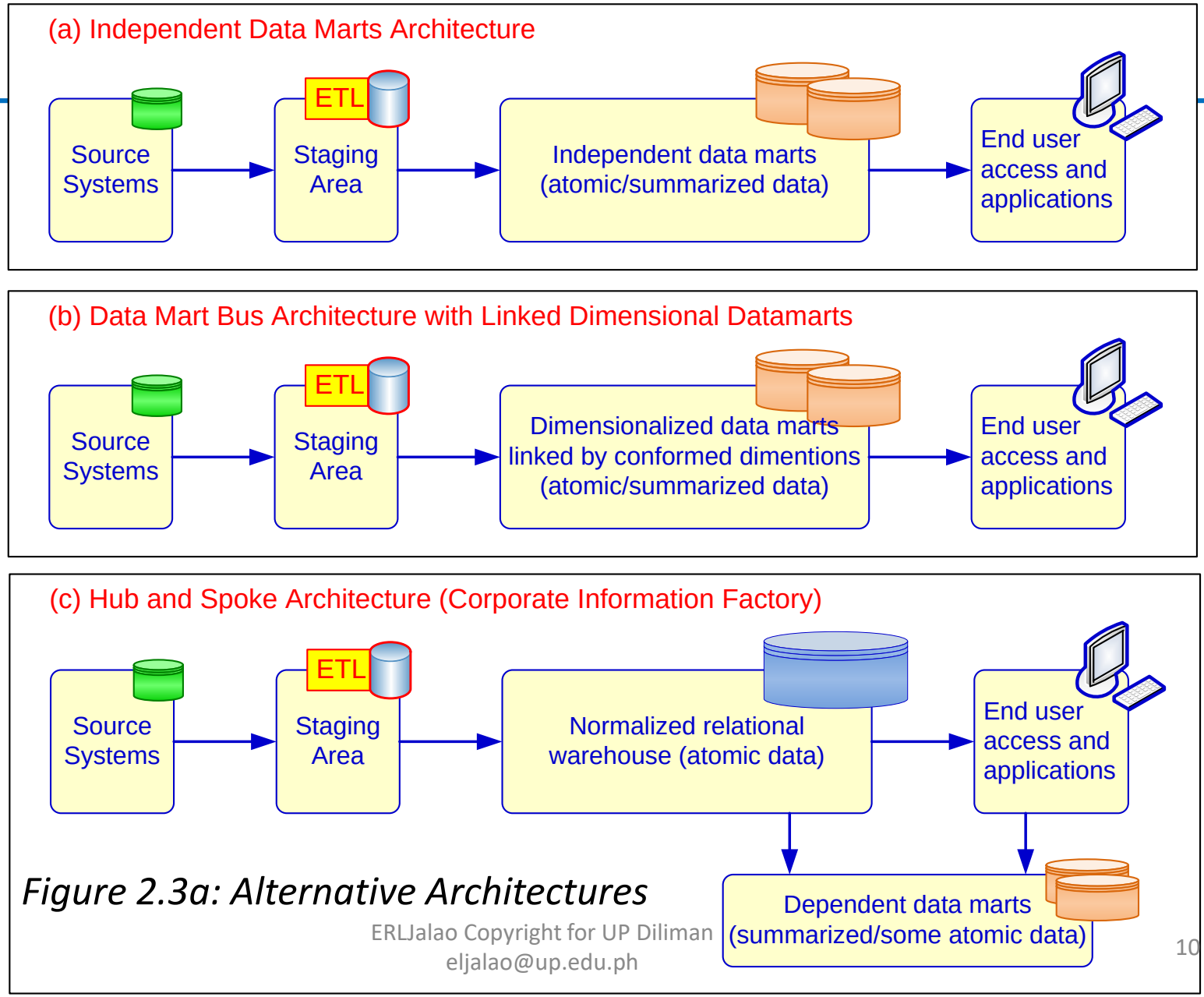

Figure 2.3a: Alternative Architectures

Intro to Data Warehousing

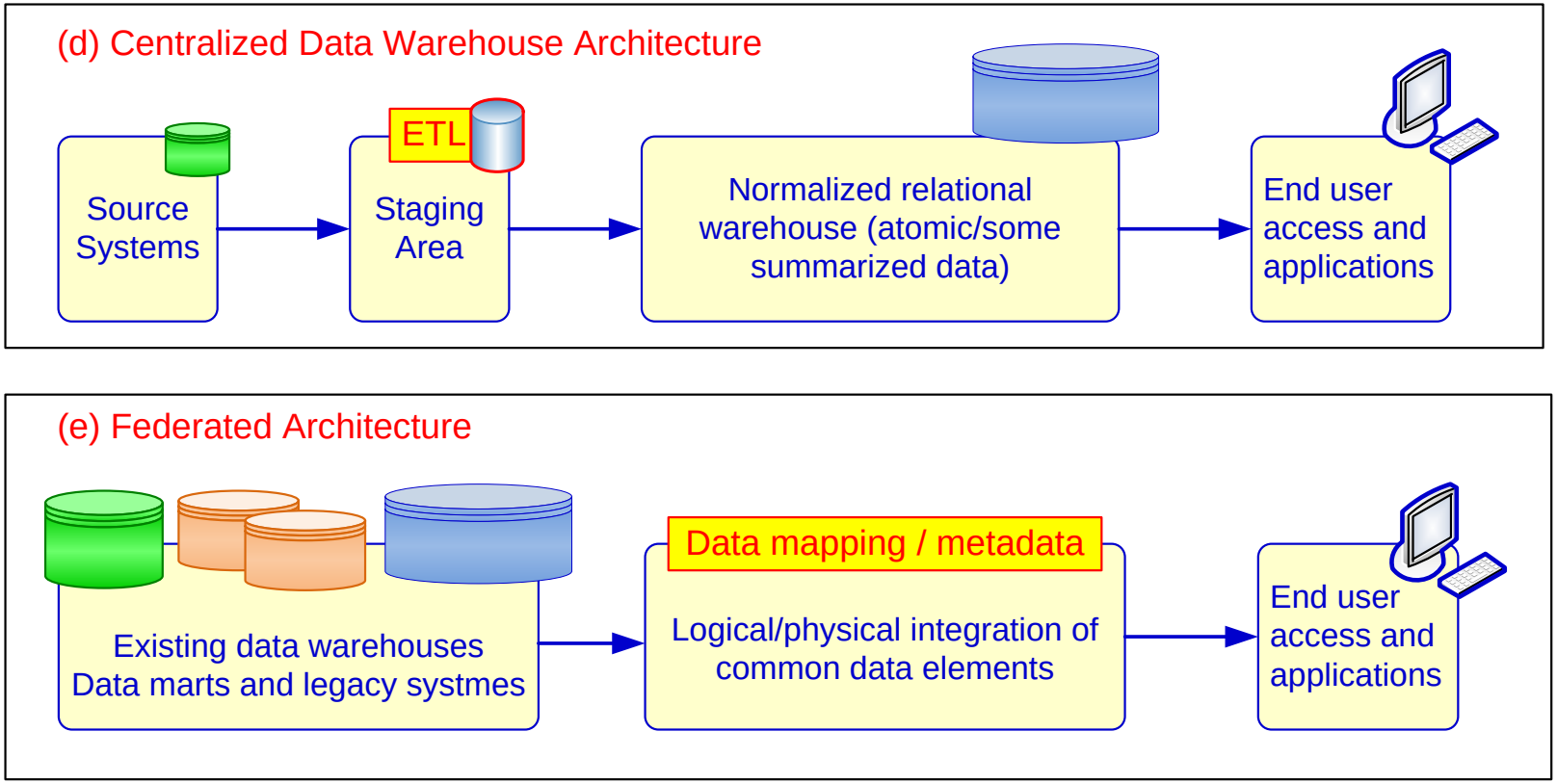

Figure 2.3b: Alternative Architectures

Intro to Data Warehousing

- Ten factors that potentially affect the **architecture** selection decision:
  1. Information interdependence between organizational units
  2. Upper management's information needs
  3. Urgency of need for a data warehouse
  4. Nature of end-user tasks
  5. Constraints on resources

Intro to Data Warehousing

- Ten factors that potentially affect the **architecture** selection decision:
  6. Strategic view of the data warehouse prior to implementation
  7. Compatibility with existing systems
  8. Perceived ability of the in-house IT staff
  9. Technical issues
  10. Social/political factors

Outline for This Session

- Intro to Data Warehousing
- **Kimball DW Lifecycle**
- Dimensional Model vs Normalized Models
- Star Schema Models
- ETL Overview
- DW Implementation Guidelines
- Case Study

Kimball DW Lifecycle

Definition 2.3: Kimball DW/BI Lifecycle

- Began at a company called **Metaphor** in the **mid-1980s**
- Originally named Business Dimensional Lifecycle
- Renamed **Kimball Lifecycle** in 2008
- Three fundamental concepts
  - Focus on business
  - End-User Easy Interpretation
  - Iterative development of enterprise data warehouse rather than big bang
  - Performance

Kimball DW Lifecycle

- Inmon vs. Kimball
  - **Bill Inmon (EDW/DM)**
    - The EDW should be in at least 3rd normal form.
    - But the data marts should be in dimensional form.
    - Big Bang Approach
  - **Ralph Kimball (Architected EDW)**
    - The EDW is based on dimensional model design
    - Focus on User-Friendliness and Easy to Use
    - Develop EDW on a departmental basis piece by piece
  - **Difference?**
    - Kimball's approach is more practical, more interpretable, easier to implement and less costly based on industry best practices.

The Kimball DW Lifecycle

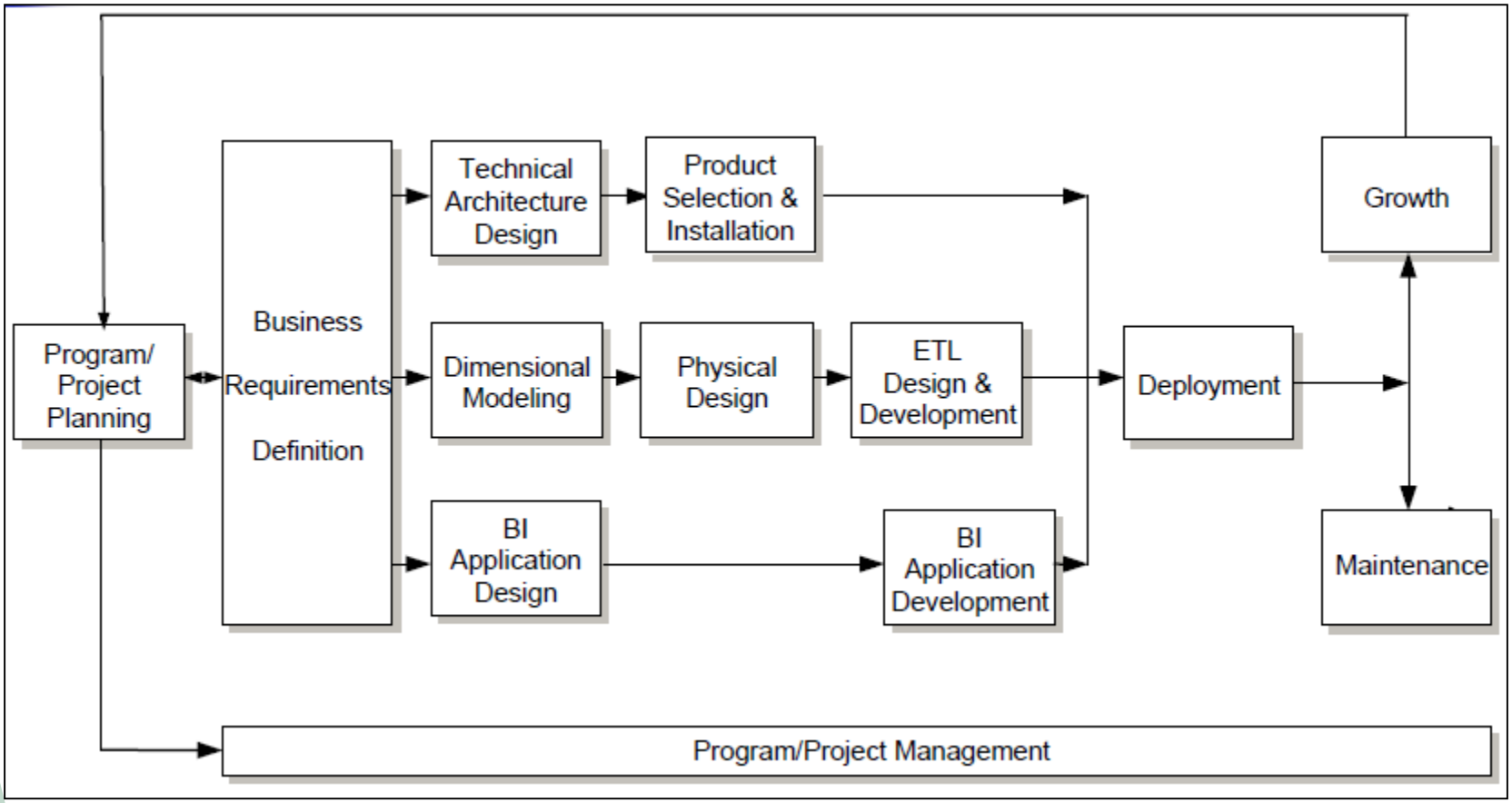

Figure 2.4: Kimball DW Lifecycle

Program/Project Planning

- **Define and scope** the DW
- **Readiness** assessment
- **Resource planning** including hardware, software and staffing requirements
- Define and sequence **tasks** for entire DW lifecycle
- Estimate **tasks, durations**
- Assign **staff to tasks**, balance resources
- Communicate the **Project Plan**

eljalao@up.edu.ph

Program/Project Management

- Keep project on track; avoid **scope creep**
- Track and resolve **issues and bugs**
- Maintain **continuous communications**
- Manage **expectations**
- Enable **creeping commitment**
- Establish and maintain a **DW Executive Steering Committee**

Business Requirements Definition

- Understand the **business**
- Understand business **user requirements**
- Business requirements establish foundation for **three parallel tracks**
  - Data track
  - Technology track
  - Application track
- Develop Business **case and justification**

The Kimball Lifecycle

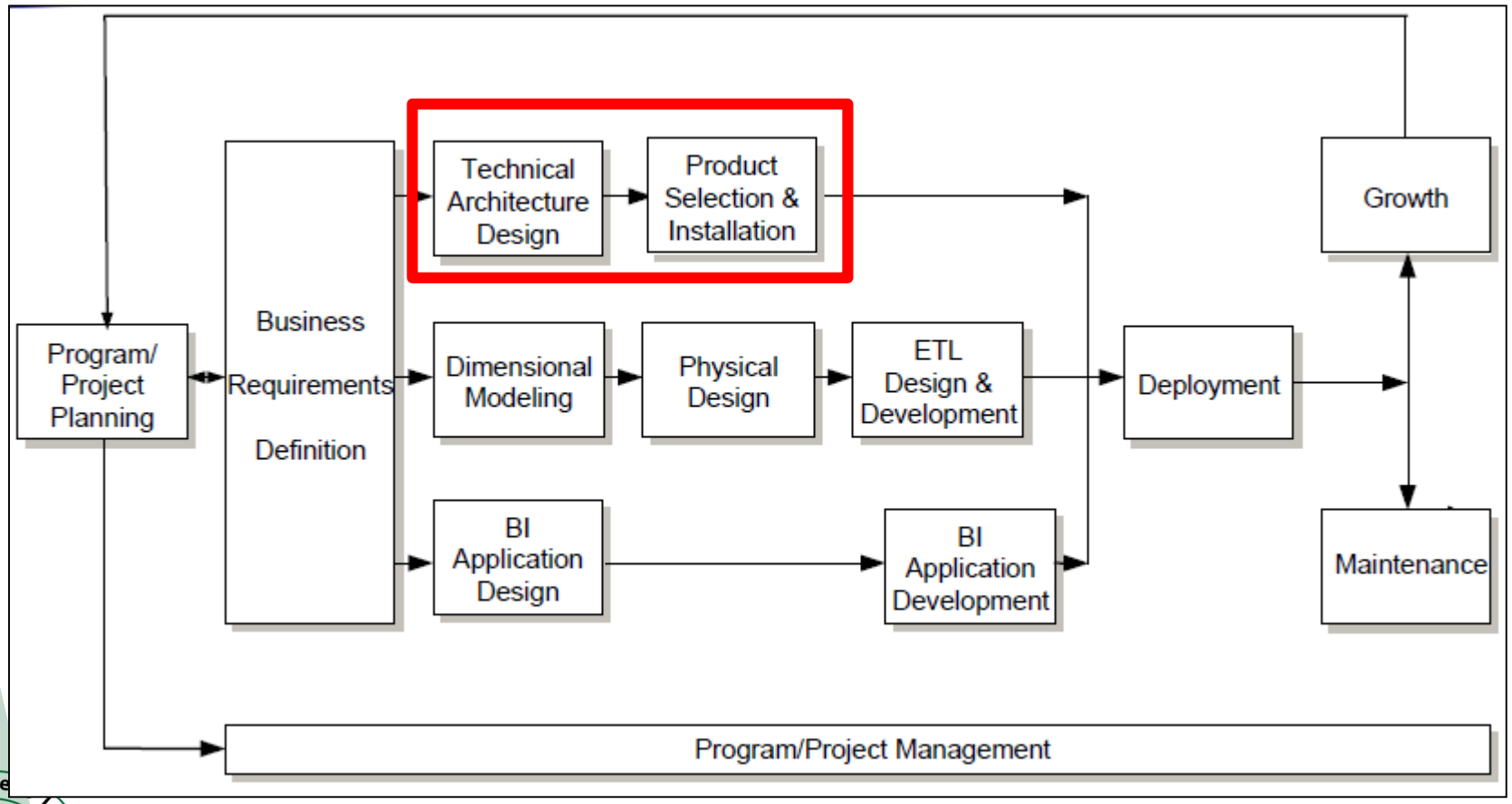

Figure 2.4: Kimball DW Lifecycle

Technology Track: Technical Architectural Design

- Consider **three factors** simultaneously:
  - Business requirements, Current technical environment and Planned strategic technical directions
- Design **back room architecture**
  - Design ETL (data staging) environment
  - Identify DBMS operating system and hardware environment
- Design **front room architecture**
- Design the Infrastructure and **metadata**
- Manage **security** requirements

Technology Track: Product Selection and Installation

- Evaluate and select the following **tools**:
  - Hardware platform
  - DBMS
  - ETL tool (data staging tool)
  - BI tool (end user data access tool)
- Install and test to assure **end-to-end integration**
- Train **team**

The Kimball Lifecycle

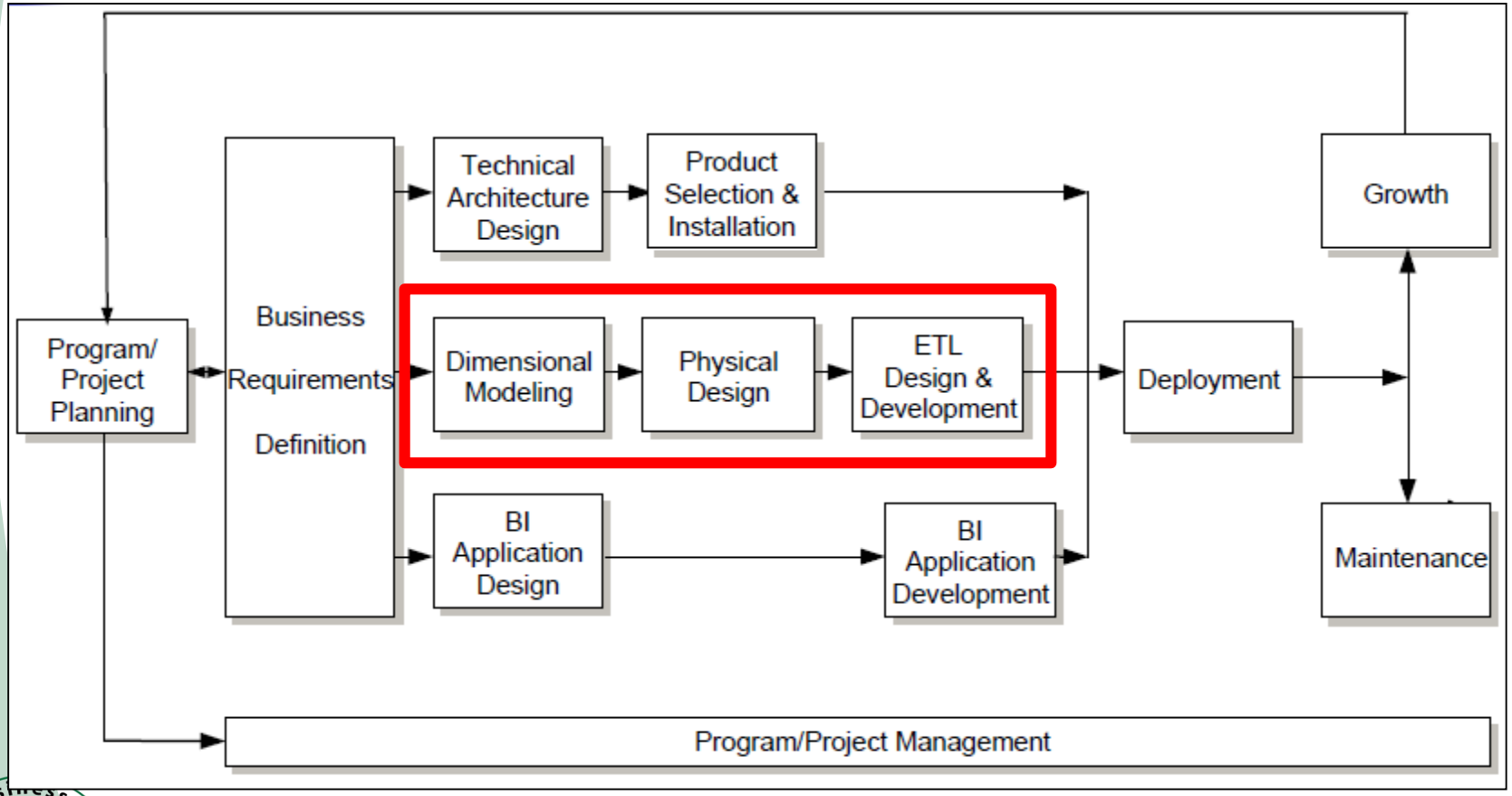

Figure 2.4: Kimball DW Lifecycle

Data Track: Dimensional Modeling

- Identify business **processes/events** and the associated fact tables and dimensions
- Analyze relevant operational **source systems**
- Develop dimensional model using a **standard methodology**
- Develop preliminary **aggregation plan**

Data Track:
Physical Design

- Define **data naming** standards
- Set up **database environment**
- Determine indexing and **partitioning strategies**

eljalao@up.edu.ph

Data Track:
ETL Design and Development

- Three major steps: **Extract, Transform, Load (ETL)**
- Develop **source-to-target** data mappings
- **Extract data** from source operational systems
  - Expose data quality issues buried in source systems
- **Transform** to move and clean/correct data
- **Load** - two staging processes
  - Initial load, including available historical data
  - Incremental loads, often daily
- Typically **underestimated**

eljalao@up.edu.ph

The Kimball Lifecycle

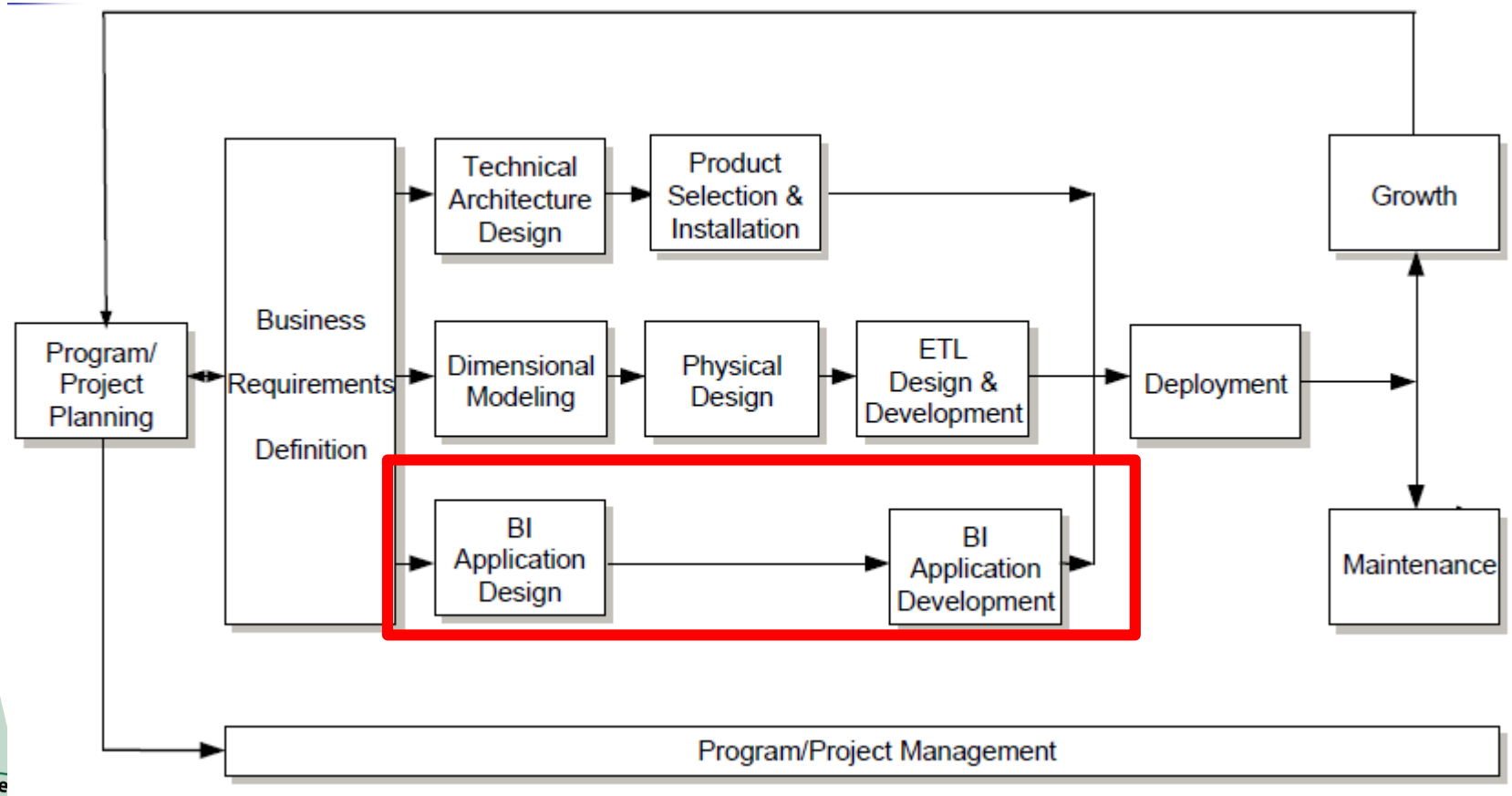

Figure 2.4: Kimball DW Lifecycle

Application Track:
BI Application Design

- Identify standard **analytic and report requirements** to meet 80% – 90% of user needs
- Plan and assure **ad hoc query and reporting capability**
- Develop **report templates** for report families
- Get **user signoff** on report templates and commit to them
- Identify **metrics and metric calculations**, Key Performance Indicators (KPIs)

eljalao@up.edu.ph

Application Track:
BI Application Development

- Ideally, use a **single advanced BI** tool that meets all user needs
- Advanced tools provide significant **productivity gains** for the application development team
- Good BI design enables end users to modify existing reports and develop ad hoc reports **quickly** without going to IT
- The best tools provide powerful **Web-enabled capability**

Deployment

- Develop and implement **user testing plan**
- Develop **test protocols** to provide thorough, explicit, reusable documents for testing and training
- Obtain user signoff via **User Acceptance Test (UAT)**
- Develop and implement **user training plan**
  - Classes
  - Online manual
- Develop and implement **user support plan**
  - Help desk
  - Problem reporting, tracking, resolution

Maintenance

- Adapt to **business changes**
- Ongoing **user training and support**
- Maintain and monitor DW **usage statistics**
- **Purge and archive** data

eljalao@up.edu.ph

Growth

- Add **new** business dimensional projects
- **Leverage** existing dimensions
- Repeat the Lifecycle iteratively for each project

eljalao@up.edu.ph

Outline for This Session

- Intro to Data Warehousing
- Kimball DW Lifecycle
- **Dimensional Models vs Normalized Models**
- ETL Overview
- Case Study

Dimensional Models

Definition 2.4: Dimensional Modelling

- Dimensional modeling is a **logical design** technique for structuring data so that it is **intuitive** for business users and delivers fast query performance.
- **Widely accepted** as the preferred approach for DW/BI presentation.
- **Simplicity** is fundamental to usefulness.
- Allows software to easily navigate databases.
- Divides world into **measurements** and **context**.

Dimensional Models

- Dimensional models are the **front room deliverable**
- They provide the business users ease of use and **fast BI query** performance
- Same content as normalized relational models (or more) but **denormalized** for understanding and performance

Dimensional Models

Definition 2.5: Facts

- Measurements are **numeric values** called facts
  - Example: Sales Amount, Count of Attendance

Definition 2.6: Dimensions

- Context intuitively divided into clumps called **dimensions**. Dimensions describe the "who, what, where, when, why, and how" of the facts.
  - Example: Sales by Quarter, Sales by Product, Count of Attendance by Course

Dimensional Models

- A dimensional model consists of a **fact table** containing measurements surrounded by a halo of **dimension tables** containing textual context.
- Known as a **star join**.
- Known as a **star schema** when stored in a relational database

Dimensional Models

Definition 2.7: Star Schema

- The most commonly used and the simplest style of dimensional modeling
  - Contain a **fact table** surrounded by and connected to several **dimension tables**
  - Fact table contains the **descriptive attributes** (numerical values) needed to perform decision analysis and query reporting
  - Dimension tables contain **classification and aggregation** information about the values in the fact table

Dimensional Models

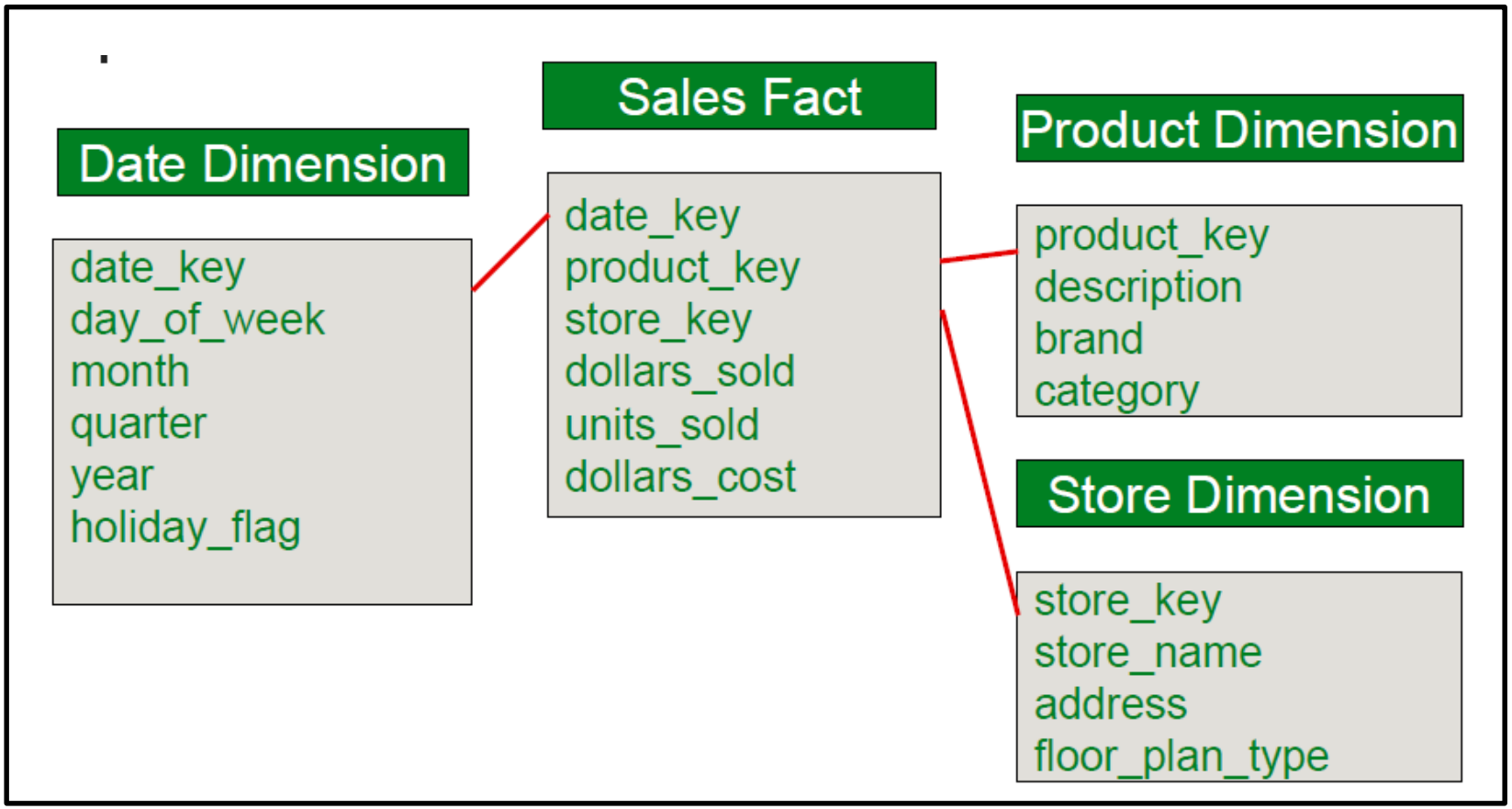

Figure 2.5: Star Schema Example

Normalized Modeling

Definition 2.8: Normalized Modelling

- A Normalized Model is a **logical design** technique for structuring data which consists of several tables designed to **minimize redundancy and dependency**.
- Tables are joined using **keys**
- Other than keys, each attribute may appear in only **one table.**
- Currently used as the

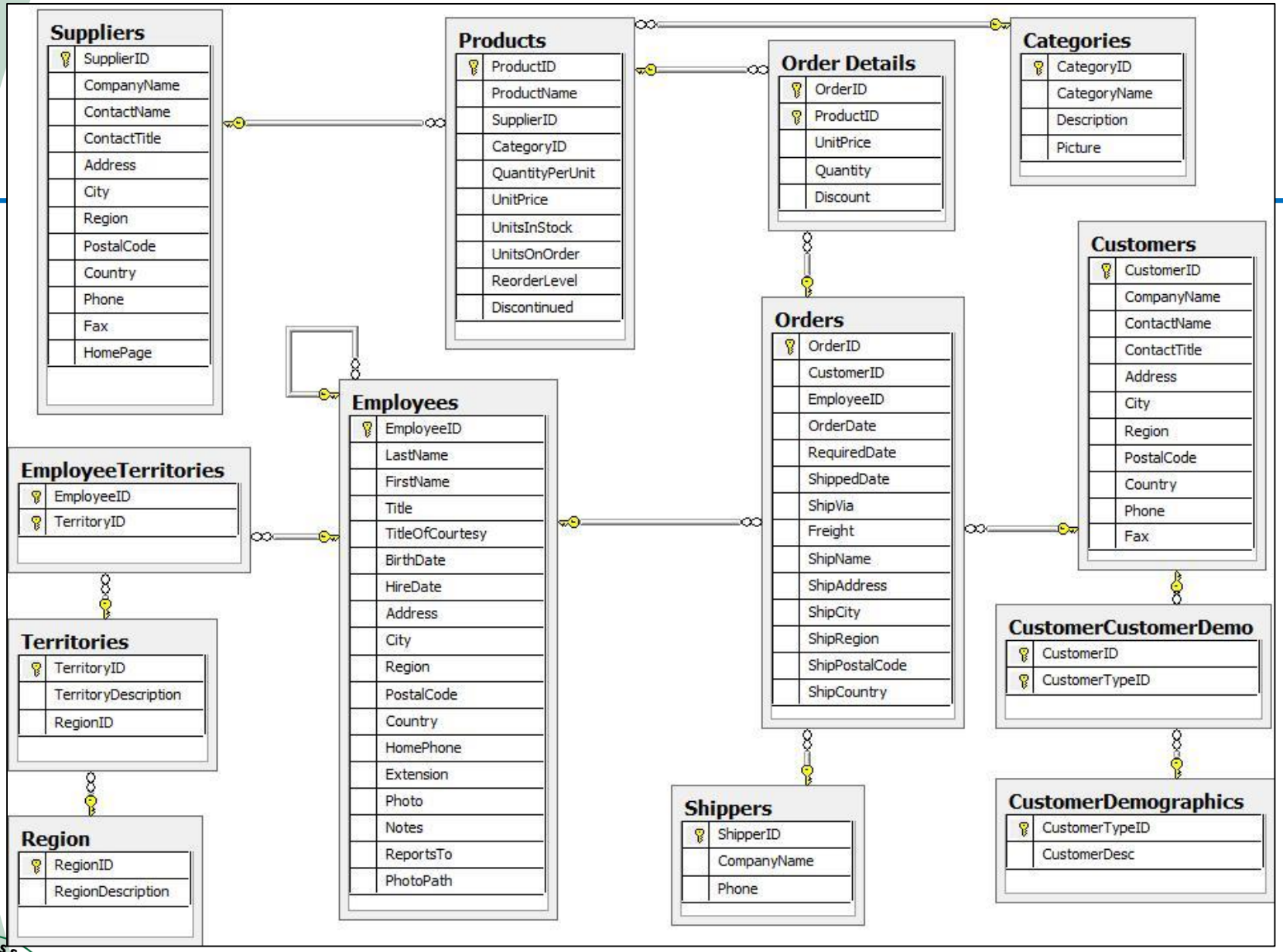

Figure 2.6: Normalized Model

Normalized Modeling

- Design objective: a **Third Normal Form (3NF) model**.
- Modeling business processes results in numerous data entities/tables and a **spaghetti-like interweaving** of relationships among them.
  - Some ERP systems have **tens of thousands of tables**.
  - Even a small model can be challenging.

Normalized Modeling versus Dimensional Models

- Normalized models look very **different** from dimensional models
  - Normalized models **confuse** business users
  - Business users see their business in dimensional models
- Dimensional models may contain **more content** than normalized models
  - **History**
  - Enhanced with content from **external sources**

Normalized Modeling versus Dimensional Models

- Advantages of Normalized Models
  - Normalized models essential to good **operational** systems
  - Excellent for **capturing and understanding** the business (rules)
  - Great for speed when processing **individual transactions**
  - When properly designed and implemented, they assure **referential integrity**

eljalao@up.edu.ph

Normalized Modeling versus Dimensional Models

- Disadvantages of Normalized Models
  - Not usable by end-users – **too complicated and confusing**
  - Not usable for DW queries – performance too slow (**many joins**)
  - But make **excellent source** if available in operational system

Outline for This Session

- Intro to Data Warehousing
- Kimball DW Lifecycle
- Dimensional Model vs Normalized Models
- **ETL Overview**
- Case Study

eljalao@up.edu.ph

ETL

Definition 2.9: ETL

- Stands for Extraction, Transformation and Loading
  - Objective: To get **data out** of the source and **load it** into the data warehouse – simply a process of copying data from one database to other
  - Data is **extracted** from a database, **transformed** to match the data warehouse schema and **loaded** into the data warehouse database
  - When defining ETL for a data warehouse, it is important to think of **ETL as a process**, not a physical implementation
  - Usually handled using **Structured Query Language (SQL)** scripts
    - SQL: A special-purpose programming language designed for managing data held in a **relational database**

ETL

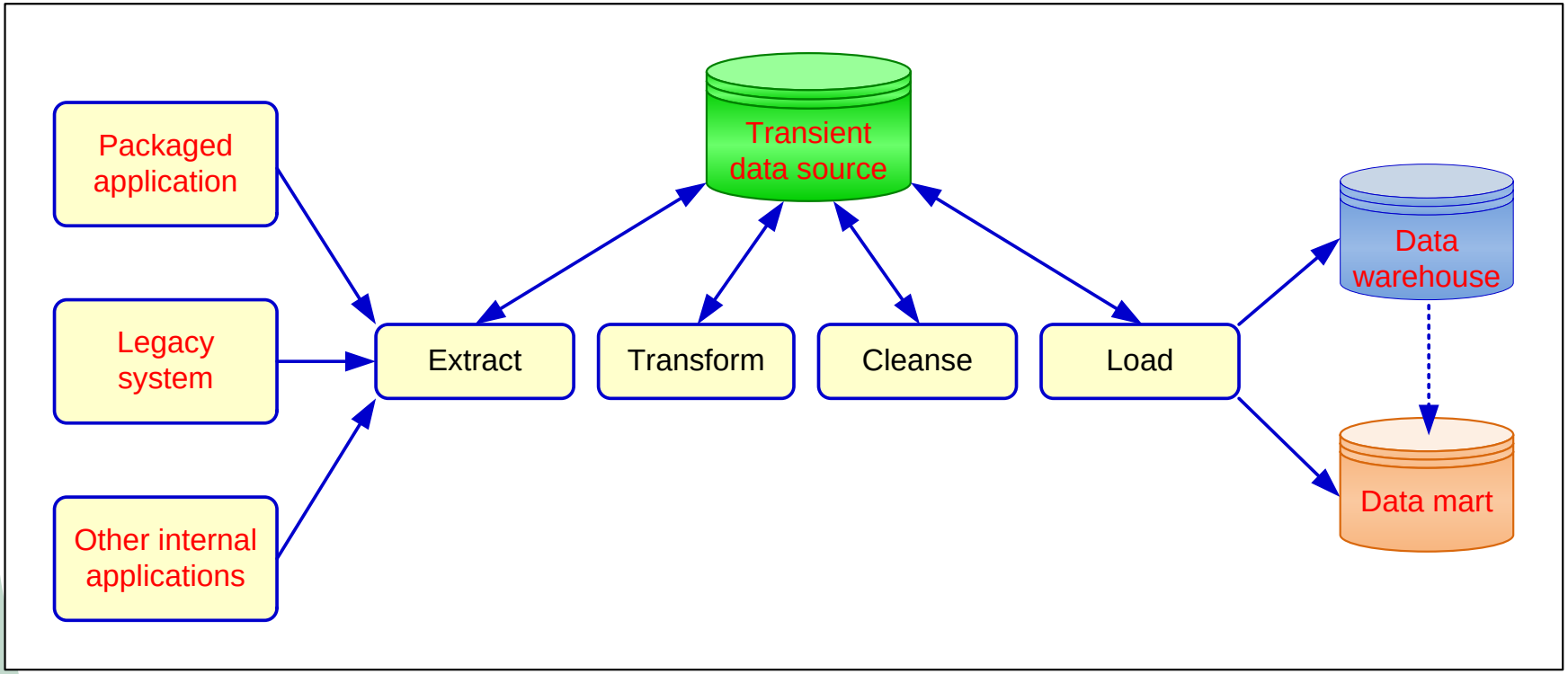

Figure 2.7: ETL Framework

ETL

- ETL is often a **complex combination** of process and technology that consumes a significant portion of the data warehouse development efforts
- It is **not a one time event** as new data is added to the Data Warehouse periodically **– monthly, daily, hourly**
- Because ETL is an integral, ongoing, and recurring part of a data warehouse
  - **Automated**
  - **Well documented**
  - **Easily changeable**

ETL

Definition 2.10: Extraction

- Data is **extracted** from heterogeneous data sources
- Each data source has its **distinct set** of characteristics that need to be managed and integrated into the ETL system in order to **effectively extract data**.
- Usually done using **SQL Select Statements**

ETL

Definition 2.11: Transformation

- Main step where the **ETL adds value**
- Actually **changes data** and provides guidance whether data can be used for its intended purposes
- Performed in a **staging area**
- Sample Transformations
  - **M for Male**
  - **1 for Yes**

ETL

Definition 2.12: Loading

- Data **is loaded** into data warehouse tables
- Creating and assigning the **surrogate keys** occur in this module.
- Usually done using **Insert SQL Statements**

ETL

- Tool Selection
  - Important criteria in selecting an ETL tool
    - Ability to **read from** and **write to an** unlimited number of data sources/architectures
    - Automatic **capturing and delivery** of metadata
    - A history of conforming to **open standards**
    - An **easy-to-use interface** for the developer and the functional user
  - Some Commercial ETL Tools
    - SQL Server Integration Services (**Microsoft**)
    - Cognos Data Manager (**IBM**)
    - BusinessObjects Data Integrator (**SAP**)

Outline for This Session

- Intro to Data Warehousing
- Kimball DW Lifecycle
- Dimensional Model vs Normalized Models
- ETL Overview
- **Case Study**

eljalao@up.edu.ph

Case Study 2

- Extracting Art

Outline for This Session

- Intro to Data Warehousing
- Kimball DW Lifecycle
- Dimensional Model vs Normalized Models
- ETL Overview
- Case Study

eljalao@up.edu.ph

References

- Simon, Alan. CIS 391 PPT Slides
- Tan et al. Intro to Data Mining Notes
- Runger, G. IEE 520 notes
- UCI Irvine Data Warehousing Notes