Introduction

School of Information Studies
Syracuse University

Agenda

- Data warehouse development methodologies
- The Kimball Lifecycle for building a data warehouse
- Requirements gathering
- Creating dimensional models from functional requirements
- Walk through the high-level dimensional modeling worksheet.

Development Methodologies

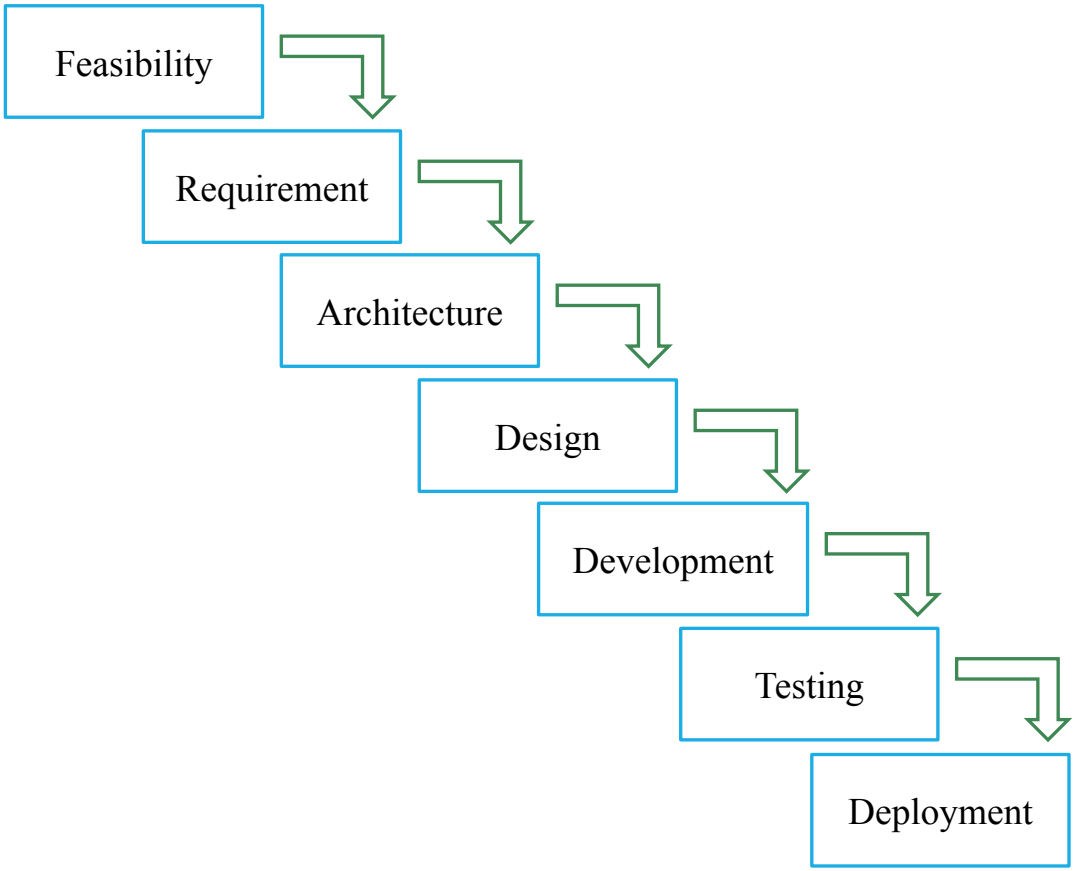
School of Information Studies
Syracuse University

User-Centric or Data-Centric?

Two Basic Models

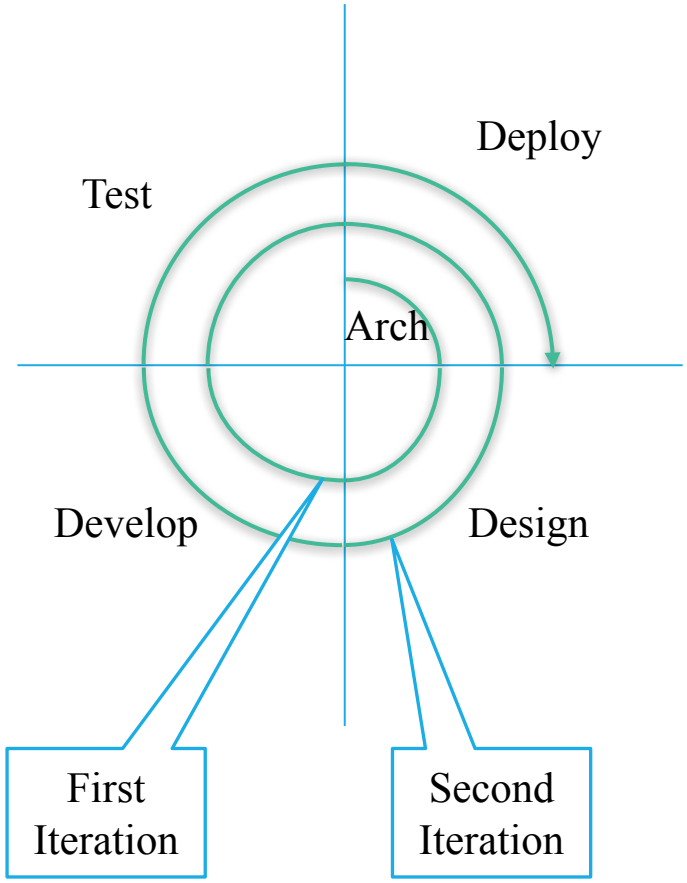
- Waterfall model
- Spiral model

Waterfall Model



- User-centric
- Starts with the user's needs and finds the data within the organization to satisfy them
- Linear in nature
- Top-down approach
- Starts with feasibility and requirements

Spiral Model



- Data-centric
- Starts with the data in the organization; ends with the user's needs
- Iterative in nature
- Bottom-up approach
- Starts with architecture

Comparison of Inmon and Kimball Approaches

School of Information Studies
Syracuse University

Kimball vs. Inmon Opposing Approaches

Kimball

- User-centric, "what we want"
- Enterprise bus TA
  - DDS is user-facing.
- Waterfall model
- Top-down (user first)

Inmon

- Data-centric, "what we have"
- Hub and spoke TA
  - NDS is internal.
- Spiral model
- Bottom-up (data first)

Which one is better? Neither!

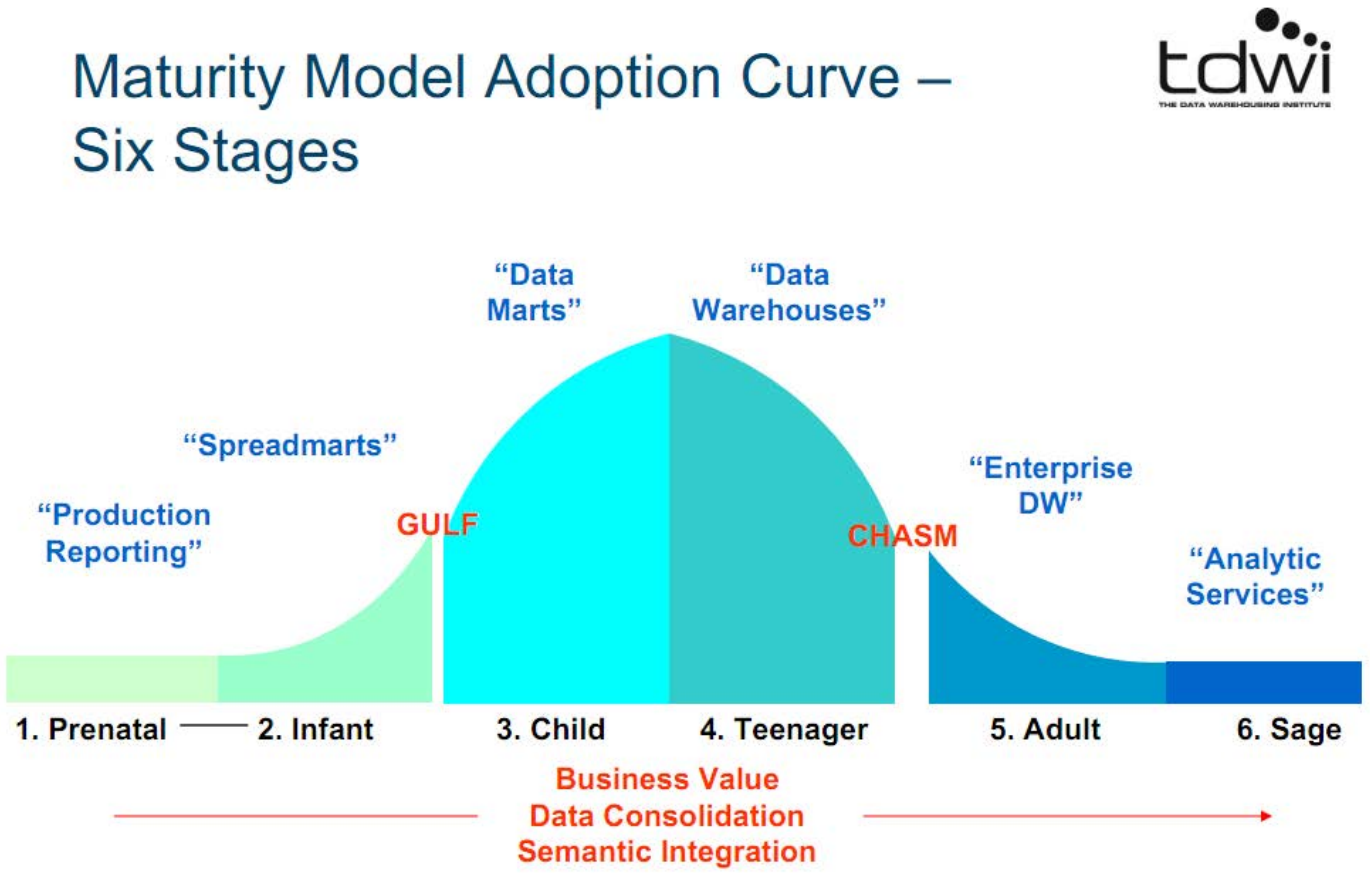
The Data Warehouse Maturity Model

School of Information Studies
Syracuse University

Realities of DW Development

- Keep your goals in mind. They drive your development.
- You will make mistakes. Consider them lessons learned.
- Business users and management will want to see results. Pick your low-hanging fruit and start there.
- You will not get it right, so you will need to iterate regardless of spiral or waterfall models.
- Keep business users and stakeholders invested in the process. It is their data! This holds true for both user- and data-centric approaches!

Data Warehouse Maturity Model



Kimball Lifecycle

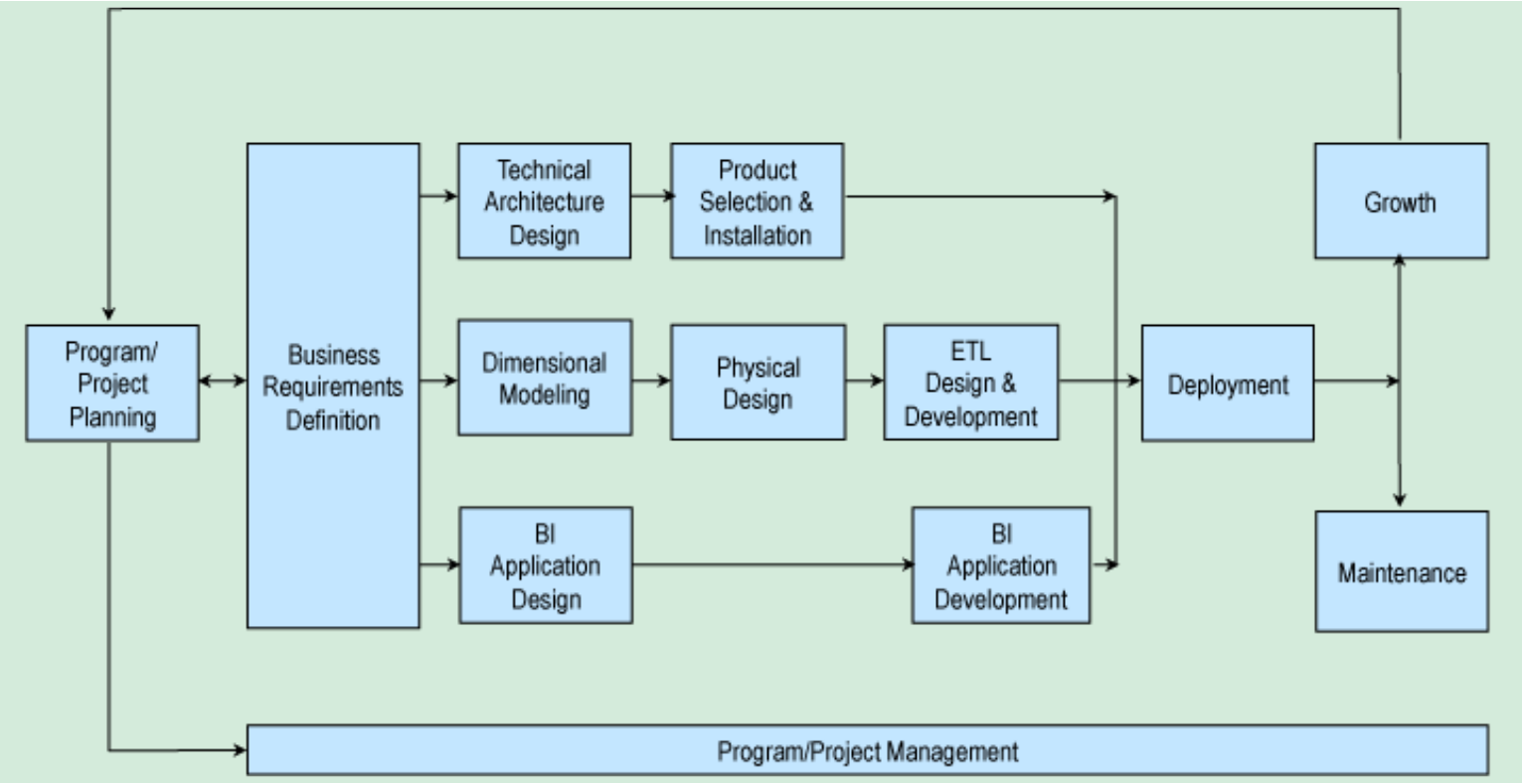
School of Information Studies
Syracuse University

Kimball Methodology

- User-centric
- Focuses on requirements, so top-down
- Enterprise bus technical architecture
- Waterfall approach, but you can build out each data mart iteratively

Kimball Lifecycle

- A framework for building a data warehouse in enterprise bus technical architecture.



Kimball Terminology

School of Information Studies
Syracuse University

Some Kimball Terminology

Program: Collection of coordinated projects.

Several data marts will be created with conformed dimensions.

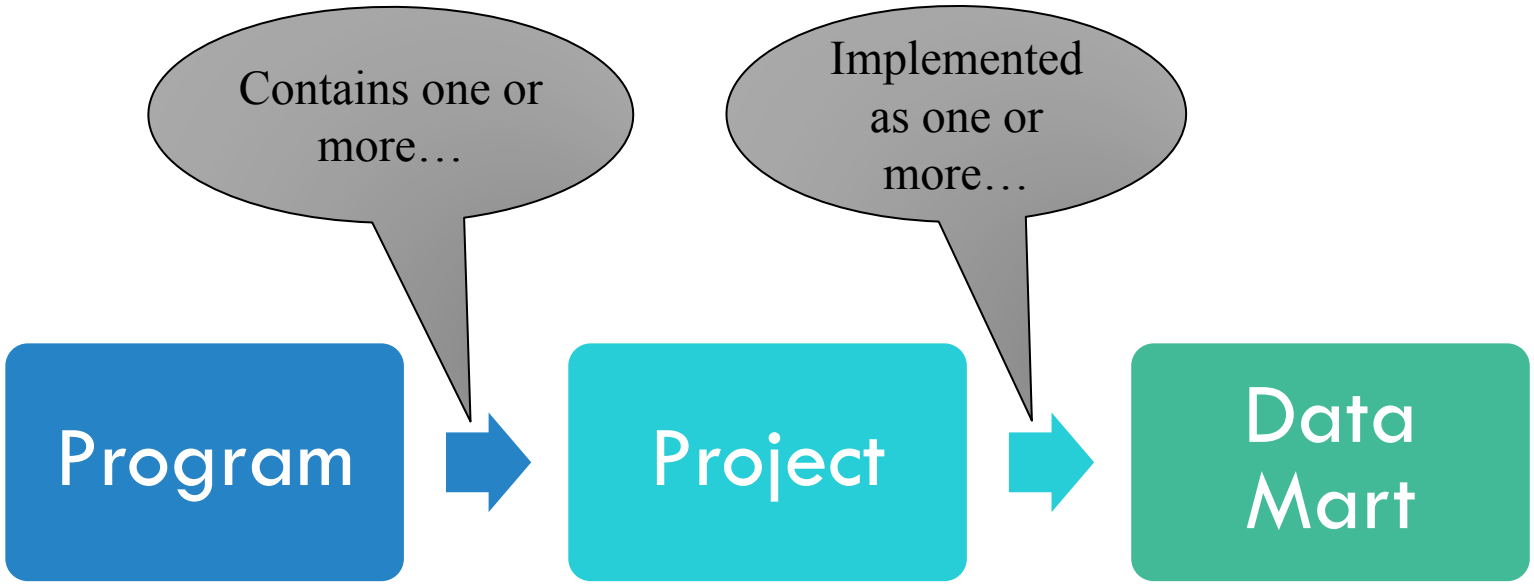
E.g., sales BI program: Build data marts for Internet sales, store sales, and partner sales.

Project: Single iteration of the entire cycle.

Encompasses a business process that results in a data mart.

E.g., build a data mart for Internet sales.

The Relationships Between: Programs, Projects, and Data Marts



Each **project team** will work on one **project** within the same **program**.

Is Your Organization Prepared to Take This On?

First you must assess your ***organizational readiness*:**

- ✓ Do you have **strong support** from **upper management**?
- ✓ Is there a **compelling business motivation** behind the initiative ?
- ✓ Is it **technically feasible** with the resources and data you're given?

Planning Activities

The **charter:**

- **Define** the project background.
- Set project **scope** and boundaries (what's excluded).
- Identify **success criteria** for the project.
- State the **business justification.**

Assemble the **project team** (minimum roles):

- **Business lead**: in charge of initiative
- **Project manager**: manages project
- **Business analyst**: collects requirements
- **Data architect**: dimensional modeling/implementation
- **ETL architect**: ETL design/implementation
- **BI architect**: BI design/implementation

Planning Activities

Establish a **communication plan.**

- How will you keep stakeholders informed?
- How often and in what form will you meet?
- Who needs to be present at which meetings?

Create your **project** plan and **task list.**

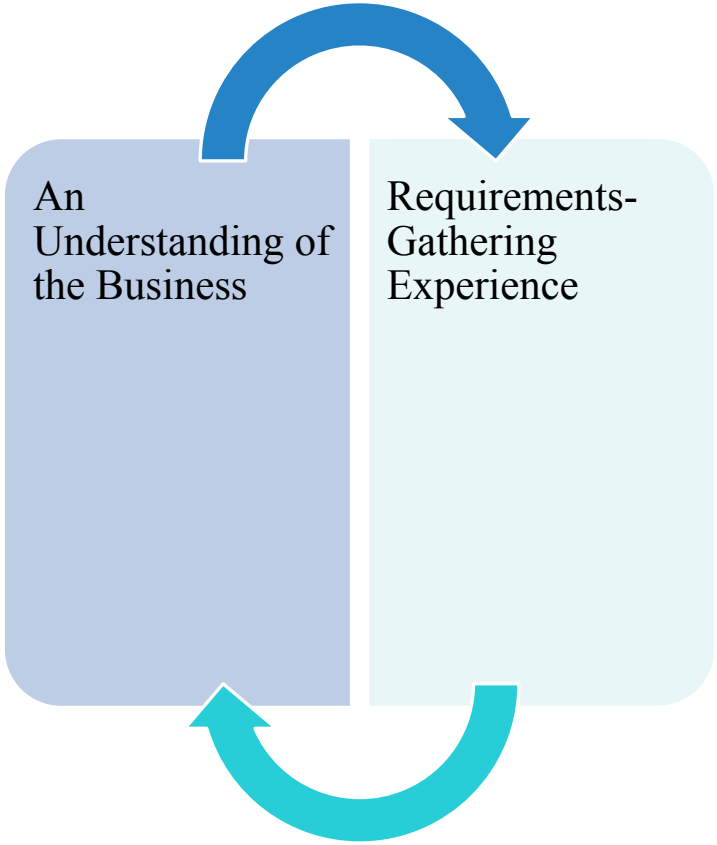
Track issues using a **change log** or **issue-tracking system**.

Hold a **kickoff meeting** to get everyone on the same page.

Understanding the Business

School of Information Studies
Syracuse University

Requisite Skills



Understanding the Business

- **Event:** a frequent activity within the business. Also known as a business process. Helps you to identify business transactions.
  - Examples: Include a purchase order; student registers for a course.
- **Status:** the condition of an object at a point in time. Helps you to identify workflows.
  - Examples: Order was packaged but now has shipped; student has registered and now receives a grade.
- **Level:** is a quantitative measurement of an object at a point in time. Helps you to identify periodic snapshots.
  - Examples: credit card balance; student GPA
- **Roles:** the who, what, and when of the event, status, or level. Helps you to identify dimensions.

Functional vs. Non-Functional Requirements

School of Information Studies
Syracuse University

Functional Requirements

- Define what the system does or should do.
- These requirements address needs of business users.
- Examples:
  - Business users must be able to analyze sales of product over time and by geographic region, customer segment, or sales territory.
  - Business users must be able to view their finance data within their own department such as revenues and expenses by fiscal period broken down into account codes.

Nonfunctional Requirements

- Guide and constrain the system architecture.
- These requirements outline procedures, rules, or regulations.
- Examples:
  - The maximum query response time should be no longer than 20 seconds.
  - Extracts from the accounting system can take place only on Sundays between 5 and 6 AM.

Requirements- Gathering Techniques

School of Information Studies
Syracuse University

How Do You Discover Requirements?

Key Activities of Requirements Gathering

Interviews with and **observations** of business users

Data audits: data profiling to assess capabilities of data sources

Documentation, including

- ✓ Interview write-ups
- ✓ Identify business processes
- ✓ Enterprise bus matrix
- ✓ Prioritization grid
- ✓ Issues list

Sample Interview Questions

- What type of routine analysis do you perform? What data are used and where do you get them? What do you do with the data once you get them?
- Which reports do you use? Which data on the report are important? If the report were dynamic, what would it do differently?
- Describe your products. How do you distinguish different products? How are they categorized? Do categories change over time?

Data Profiling

One important activity is to **explore** your existing data to get a sense of:

- **Technical feasibility of the project**
- **Structure and condition of data**
- **Availability of data sources**

We call this **data profiling.**

If the source data are in a relational database, you can use the SQL **SELECT** statement to profile data.

For other sources, **Microsoft Excel** makes for a good data-profiling tool.

Functional Requirements to Dimensional Models

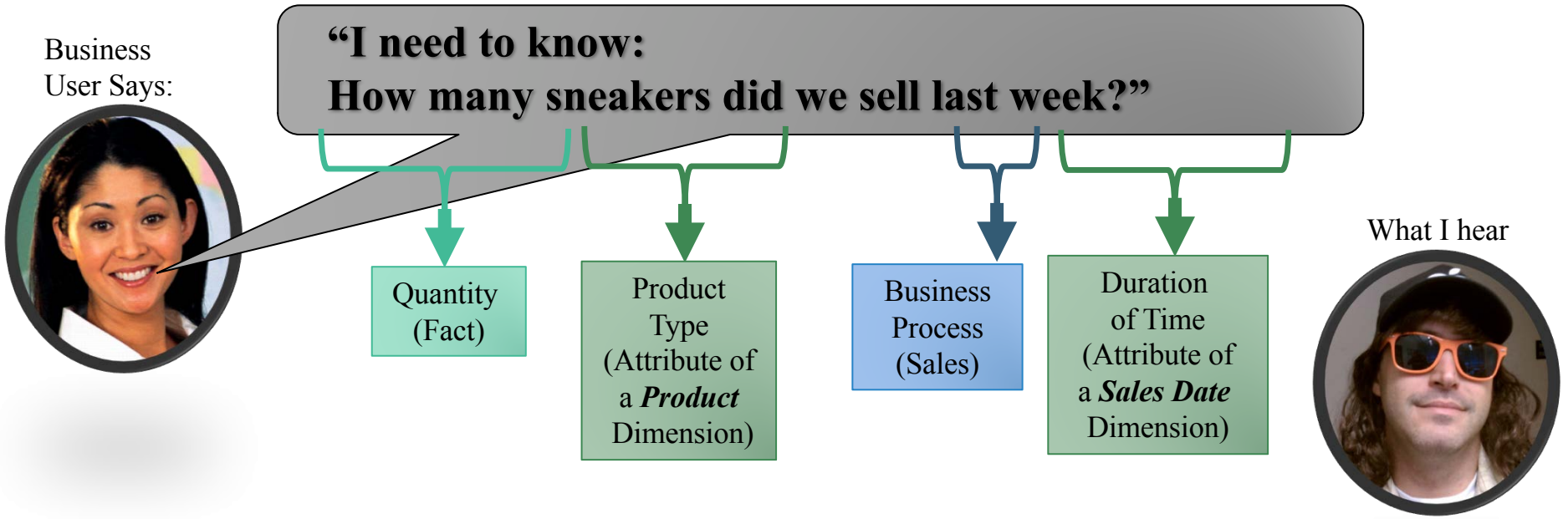
School of Information Studies
Syracuse University

Critical Skill: Turn Functional Requirements into Dimensional Models!

Here's the process for building a dimensional model from a business process. This dimensional model will eventually become a star schema in your enterprise data warehouse.

1. Identify the **business process** and **business process type**.
   These are **events**, **status,** or **levels.**
2. Identify the **facts** of the business processes.
   Quantifiable values we measure.
3. Identify the **attributes in dimensions** of the business process.
   The roles by which we measure the business process.

Demo of My Amazing Dimensional Modeling Skills!



Facts are the ***business process measurement events***.

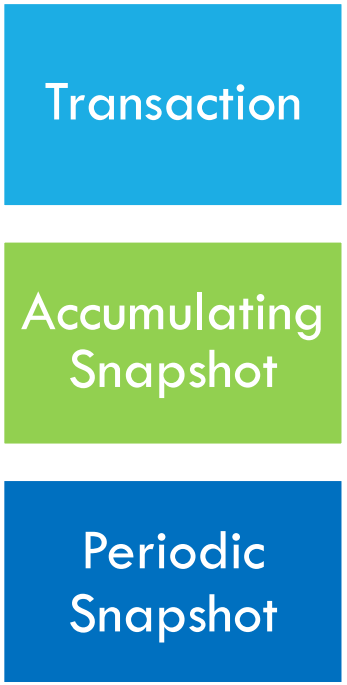
Dimensions provide the ***context*** for that event.

Identifying Business Process Types

School of Information Studies
Syracuse University

1: Identifying Business Processes Three Types

1. Events or transactions.
2. Workflows based on object status are called accumulating snapshots.
3. Quantitative measurements of points in time based on levels periodic snapshots.



These business processes end up being ***fact tables in a ROLAP star schema***. In the design phase they are called ***dimensional models.***

Transaction Fact

- Based on a business event.
- The most **basic** fact grain.
- **One row per line** in a transaction.
- Corresponds to a point in space and time.
- Once inserted, it is not revisited for update.
- Rows inserted into fact table when transaction or event occurs.

Examples:

- Sales, returns, telemarketing, registration events

Accumulating Snapshot Fact

- Based on a status.
- Used to capture a business process workflow.
- Fact row is initially inserted, then **updated** as milestones occur.
- Fact table has **multiple dates** that correspond to each milestone and records a change in status.
- **Special facts**: milestone counters and lag facts for length of time between milestones.

Examples:

- Order fulfillment, job applicant tracking, rental cars

Periodic Snapshot Fact

- At **predetermined intervals**, snapshots of quantitative measurements are taken and stacked consecutively in the fact table.
- Snapshots can be taken daily, weekly, monthly, hourly, etc.
- Complements detailed transaction facts but does not replace them.
- Sometimes required when source system does not store the underlying transactions.

Examples:

- Financial reports, bank account values, semester class schedules, daily classroom lab logins, student GPAs

Identifying the Facts of the Business Process

School of Information Studies
Syracuse University

2: Identify the Facts of the Business Process

- **Facts** are quantifiable numerical values associated with the business process.
  - How much?
  - How many?
  - How long?
  - How often?
- If it's not tied to the business process, it's not a fact.

For example:

- Points scored in a game → Fact
- Player height → Not a fact

3: Types of Facts

Additive: Fact can be summed across all dimensions

- The most useful kind of fact
- Quantity sold, hours billed

Semi-additive: Cannot be summed across all dimensions, such as time periods

- Sometime these are *averaged* across the time dimension.
- Quantity on hand, time logged on to computer.

Non-additive: Cannot be summed across any dimension

- These belong not in the fact table but with the dimension.
- Basketball player height, retail price.

Identify Dimensions and Attributes

School of Information Studies
Syracuse University

3: Identify the Dimensions and Attributes

- Dimensions provide **context** for our facts.
- We can easily identify dimensions because of the "***by***" and/or "***for***" words.
  - **E.g.,** total accounts receivables ***for*** the IT department ***by*** month
- Dimensions have **attributes** that describe and categorize their values.
  - **E.g.,** student: major, year, dormitory, gender
- The attributes help constrain and summarize facts.

Documenting Functional Requirements

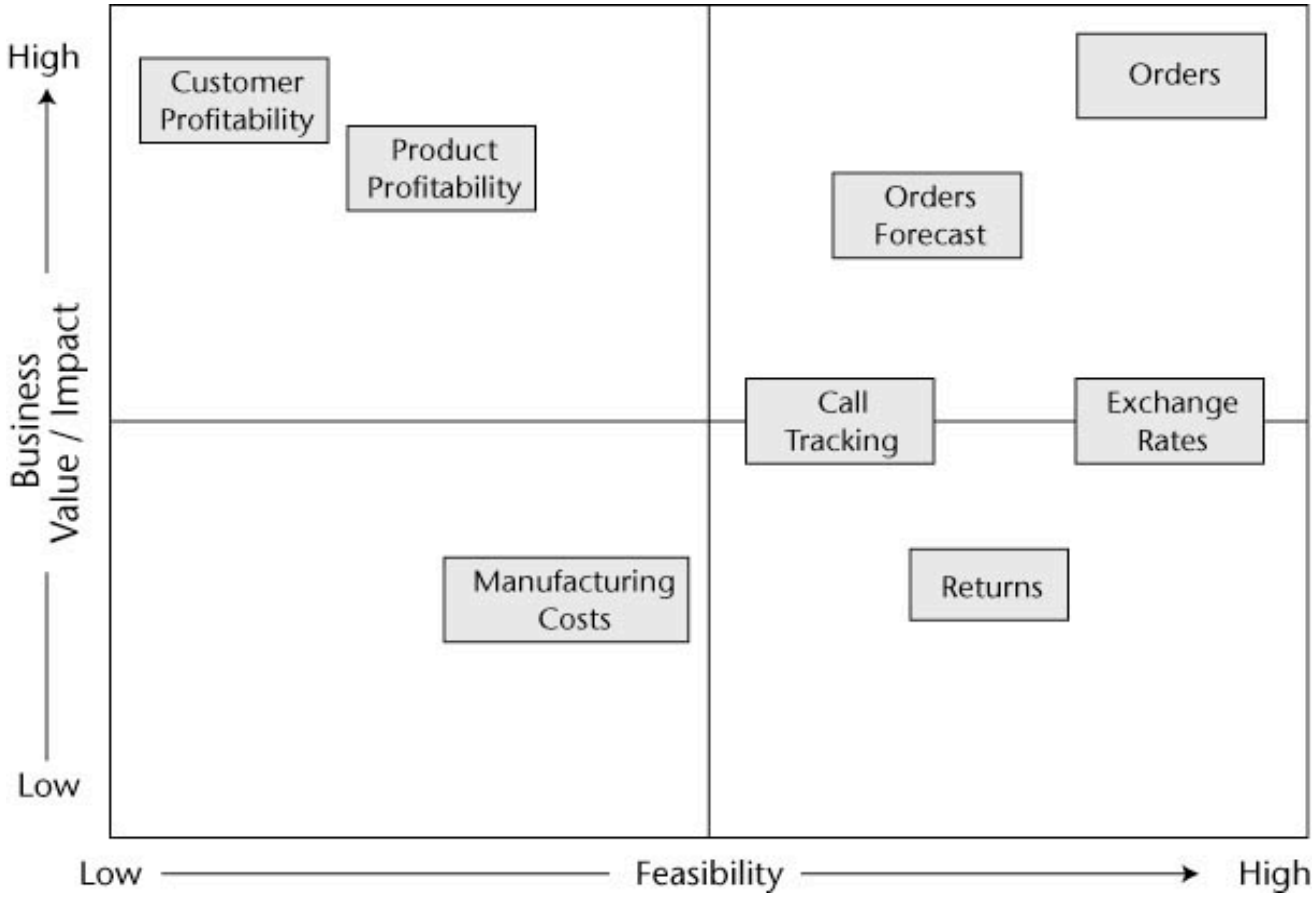
School of Information Studies
Syracuse University

Q:How do you document this?
A: Enterprise Bus Matrix

A key deliverable from requirements gathering, the **bus matrix** documents your **business processes**, **facts** and **dimensions** across all projects in your program.

Business Process Name	Fact Grain Type	Granularity	Facts	Queue Date	Ship Date	Return Date	Title	Customer
Netflix Rentals	Transaction	One row per item in rental queue	Rental, cost of title at time of rental	X			X	X
Netflix Fulfillment	Accumulating Snapshot	One row per item qeued thu returned	Queue to ship time, ship to return time	X	X	X	X	X

Prioritization Grid



Requirements Analysis Checklist

Identify each **business process** with fact grain type.

- Transaction
- Periodic snapshot
- Accumulating snapshot

Identify the **facts** and **dimensions** of each business process.

Create an **enterprise bus matrix.**

- Outline which dimensions go with which facts.
- Should be based on the data you have (profiling).

Create a **prioritization grid.**

- Establish priority for each business process. Which ones must be first?

The High-Level Modeling Worksheet Explained

School of Information Studies
Syracuse University

Demo: High-Level Modeling Sheet

Exercise: High-Level Sheet

Fill out the high-level dimensional modeling worksheet for the following functional requirements for the remote lab:

1. The manager should be able to track user logins to the remote lab. She should be able to track those logins by user, date of logon, or computer.
2. The manager should be able to determine how long a user is logged in to remote lab (aka a session) by measuring the time in minutes between login and logoff. This should be trackable by user or computer.

Demo: Solution

Debrief: How Did You Do?

- How did you do?
- Do you have any misconceptions about how to use the tool?
- Where those cleared up after you saw my solution?
- What do you understand/not understand about the tool?