Introduction to Modeling

No single view of the requirements provides a complete understanding of them. You need a combination of textual and graphical requirements representations to paint a full picture of the intended system and to help you detect inconsistencies, ambiguities, and omissions.

Types of Models

- Process
  - Business Function
  - Decomposition
  - Dependency Diagram
  - Process Maps/Flow Charts
- Event
  - Context Diagrams
  - Event Lists
  - State Transition Diagrams
- User Interface
  - Story Boards
  - Prototypes
  - Dialog Maps
  - Web Site Maps
- Data
  - Entity Relationships
  - Logical and Physical Data Models
- Combined
  - Data dictionary
  - Data Flow Diagrams
- Object Oriented
  - Class Diagrams
  - Functional Object Model
  - Component Object Model
  - Message Sequence Diagram

Source: Karl Wiegers

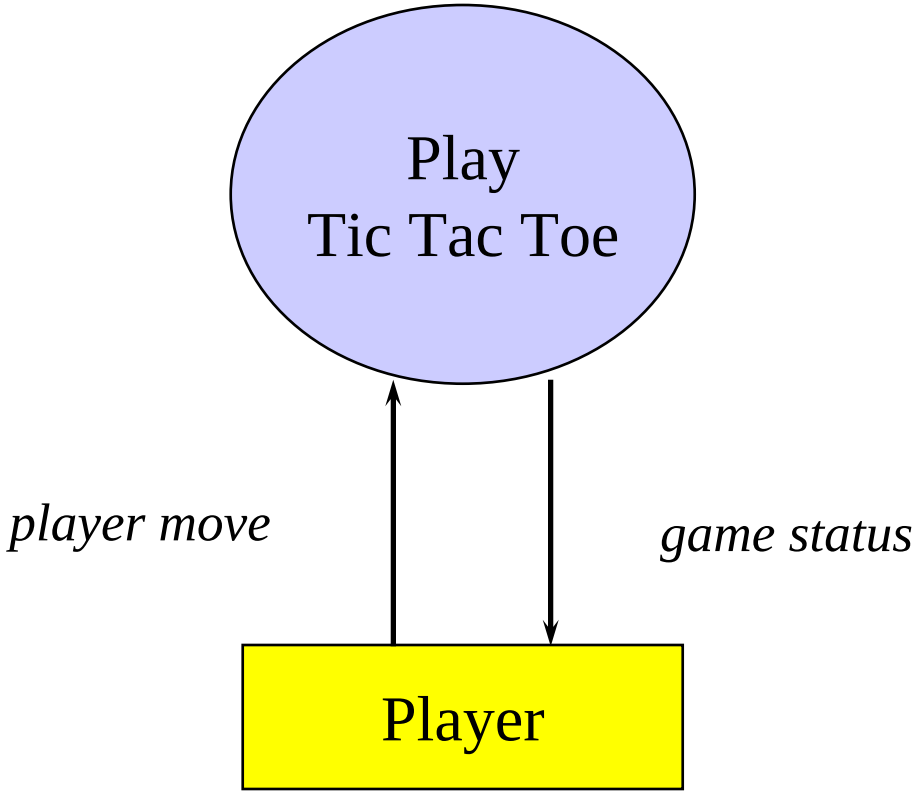
Which Model?

- The factors that influence the choice of the model:
  - The nature of the problem
  - The expertise of the requirements engineering
  - The process requirements of the customer
  - The availability of methods and tools
- In almost all cases, it is useful to start by building a model of the system boundary.

Context Diagram

- Establishes the boundaries of the system.
- Shows the system as part of the bigger picture:
  - Defines what/who interacts with the system
  - Defines the high-level flow of data/materials
- System is a black-box, no detail about what goes on inside the system is shown.
- Is built and enhanced during brainstorming, interviews, and/or facilitated sessions.
- Is broken down into other models

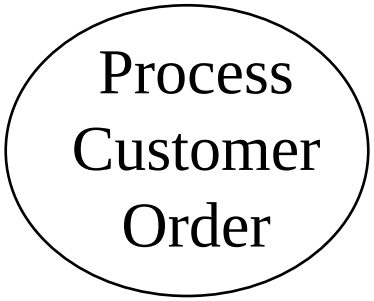
A Simple Context Diagram



Source: Karl Wiegers

Context Diagramming Steps

- Draw a circle in the middle of the page.
- Label the circle with a short name for the system.



Source: Karl Wiegers

Context Diagramming Steps

- List all terminators that interact with the system:
  - A terminator is a source or gather of data
  - Pick roles at the “end of the line”
  - Give them clear names

Terminators

Customer
Credit Card Company
Warehouse

Source: Karl Wiegers

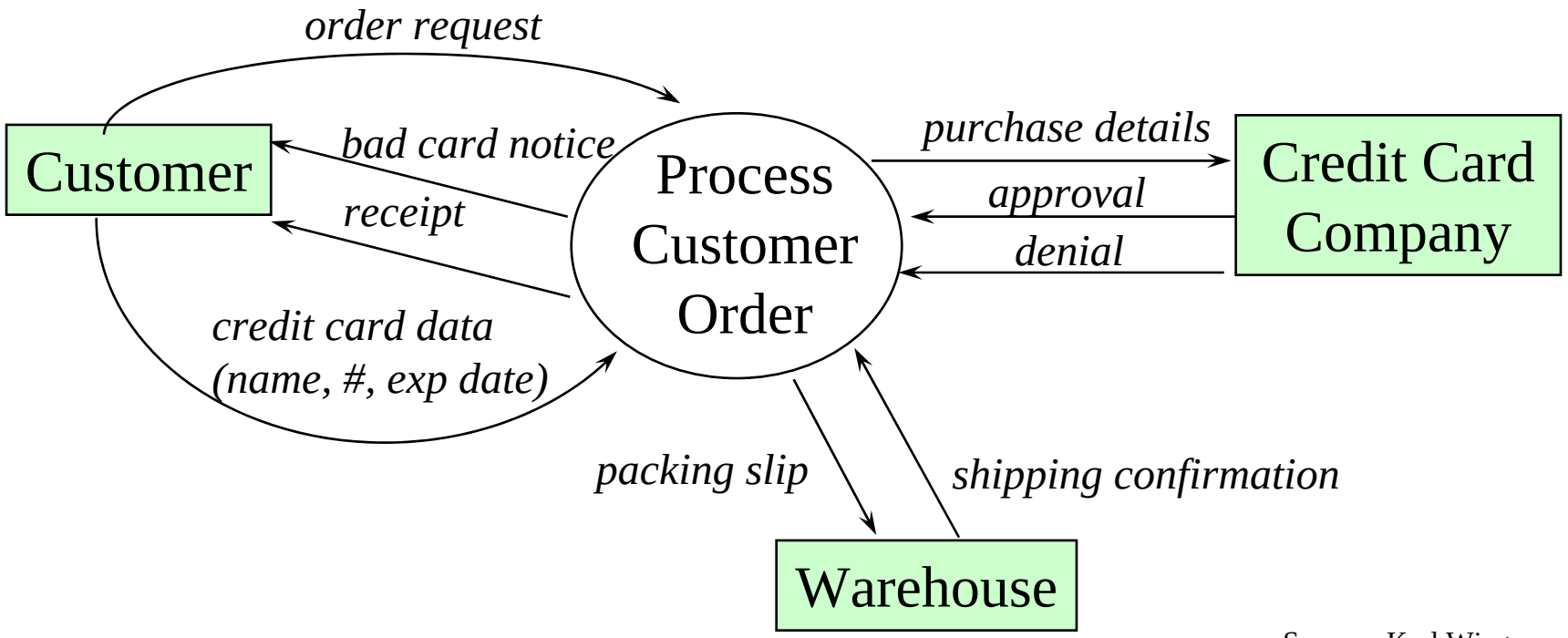
Context Diagramming Steps

- Surround the circle with the terminators in labeled rectangular boxes.

Source: Karl Wiegers

Context Diagramming Steps

- Add a line with an arrow for each data flow.
- Label each data/material flow.



Source: Karl Wiegers

Exercise

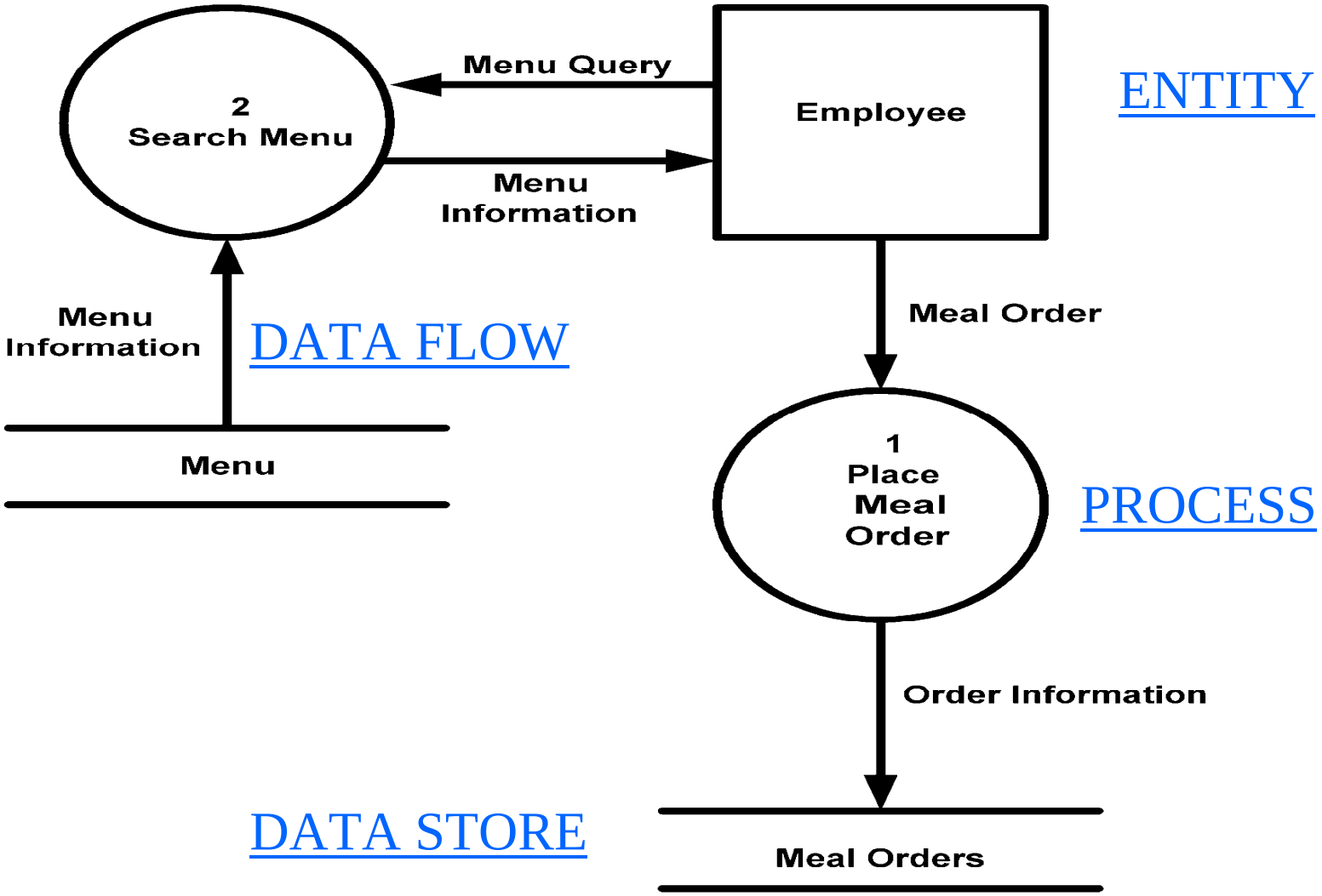
Context Diagram

Create a Context Diagram that reflects the requirements defined by the previous Brainstorming session

Data & Process Flow Diagrams

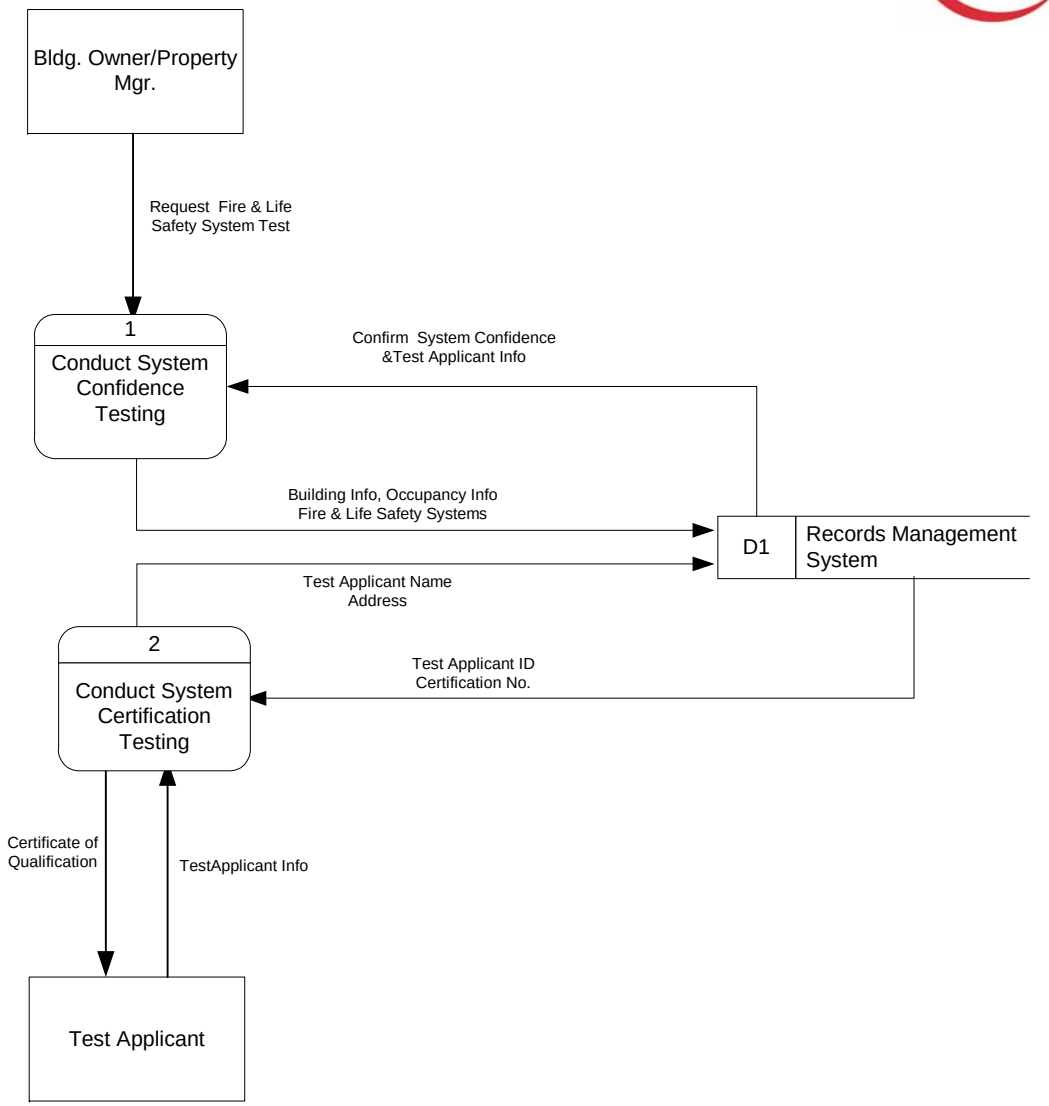
- Basic tool of structured systems analyst.
- Identifies:
  - The data stores
  - The transformation processes
  - Flow of data or material between the processes, stores, and outside world
- Works well for transaction-processing and other function-intensive applications.
- Represents systems over a wide-range of abstraction.

DFD Symbology

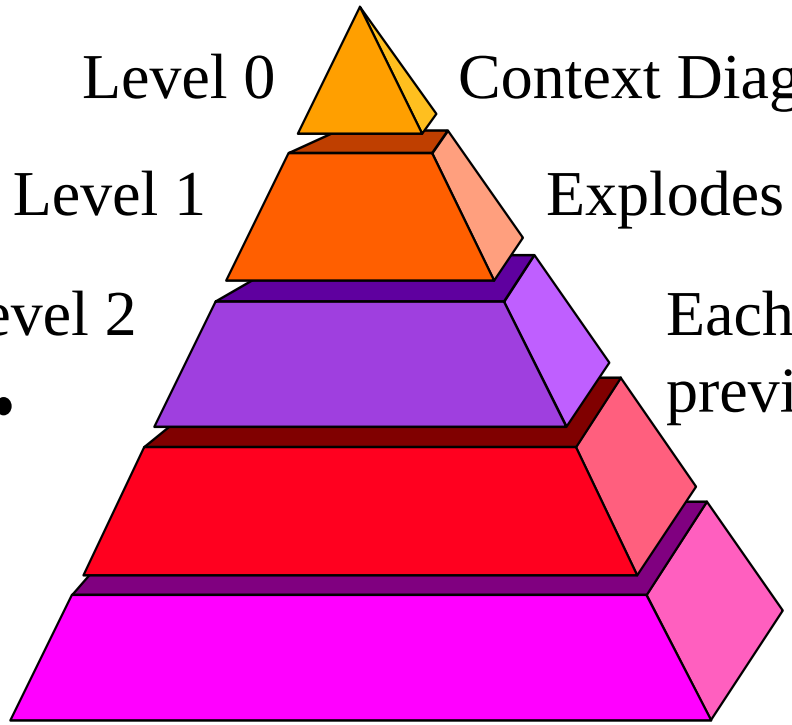


Source: Karl Wiegers

Sample Data & Process Flow Diagram



Data Flow Diagramming



Level 0 Context Diagram - Only 1 Process

Level 1 Explodes Level 0 into DFDs

Level 2 Each function/process from the previous level gets a DFD

•
•
•

Level n Have as many levels as you need to fully understand the process or the software

Source: Karl Wiegers

Ground Rules for Creating DFDs

- Place data stores only on the Tier 0 DFD and lower, not on the Context Diagram
- Processes cannot communicate directly; only through a data store
- Data cannot flow directly from one store to another; it must pass through a process
- DFDs do not imply processing sequence
- Name each process as a specific action (Verb + Action). Use names that are meaningful to the user
- Number each process uniquely and hierarchically. If a process is 2 on the Tier 0 DFD its children on the Tier I DFD would be 2.1, 2.2, 2.3, etc.
- Limit the number of processes on a single diagram (our aim is to clarify, not confuse)
- Any process that has only input, or only output, is suspect

Process Narratives

- Each Dataflow Level usually has an associated process flow narrative
  - Description of the process and business rules
  - May also contain data element descriptions

Dataflow & Process Flow Diagram

Create a Tier 0 Dataflow Diagram for the Meal Ordering System

Introduction to Data Modeling

Data Modeling is …

- The process of analyzing and representing:
  - The things a business captures information about.
  - Facts about the things.
  - The relationships between the things.
- For the purposes of:
  - Understanding and communicating business knowledge.
  - Developing stable, flexible, and reusable databases to improve business performance.

Source: Karl Wiegers

Data Modeling Objectives

- Document and communicate business information requirements.
- Enable databases with:
  - Minimum redundancy
  - Maximum integrity
  - Reusability
  - Consistency
  - Timely access
  - Usability
  - Stability
  - Flexibility
- Decrease application development and maintenance costs.
- Increase productivity within the company.

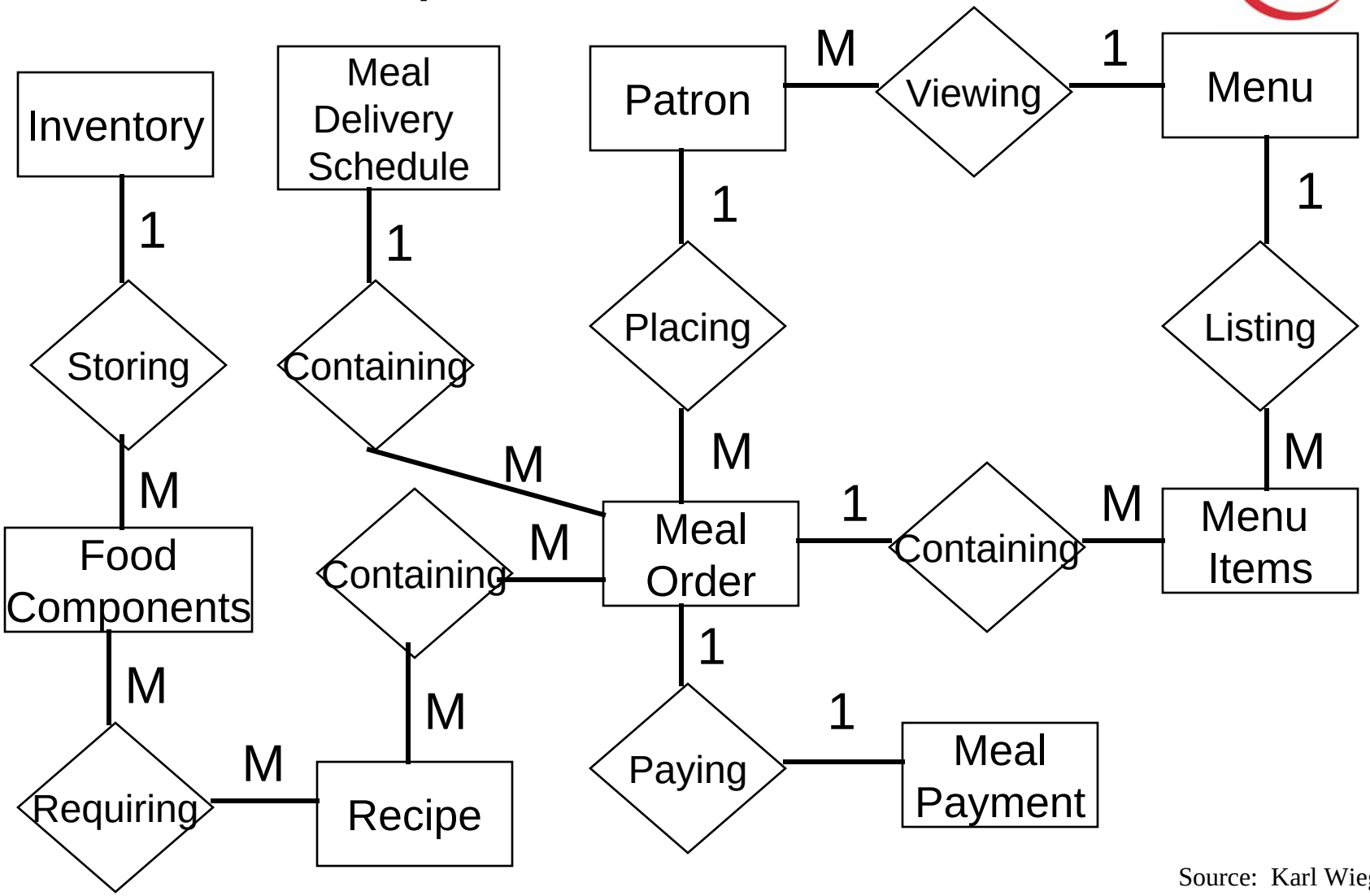
Source: Karl Wiegers

Entity Relationship Diagram (ERD)

An analysis model that identifies the logical relationships between pairs of entities.

Wiegers, p. 485

ERD Example



Source: Karl Wiegers

Entity Attributes

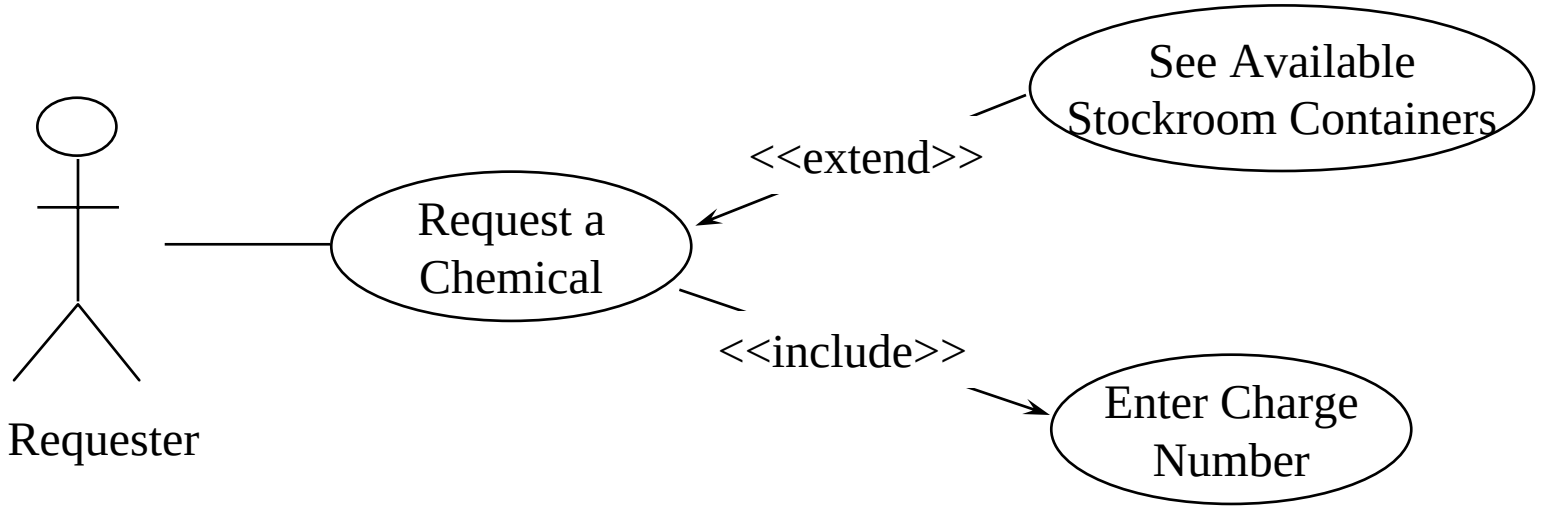
Meal Order
Meal Order Number Order Date Meal Date 1:M {Ordered food item} Delivery Instruction Meal Order Status

Meal Date =
The date the meal is to be delivered or picked up; format MM/DD/YYYY; default = current date if the current time is before the order cutoff time, else the next day; may not be prior to the current date.

Source: Karl Wiegers

What are Use Cases?

- Use case describes a sequence of interactions between a system and external actor that results in the actor accomplishing a task that provides benefit to someone.



How to Identify Use Cases

- Identify actors and their roles.
- Identify the external events to which the system must perform.
- Express business processes in terms of specific scenarios.
- Derive likely use cases from existing functional statements.

(See example in Student Manual)

Traps to Avoid

- Too many use cases.
- Duplication across use cases.
- User interface design included in the use case.
- Including data definitions in use case.
- Attempting to associate every requirement with a use case.

Summary

- Sources of Requirements
- Requirements Elicitation Techniques
  - One on One Interviews
  - JRP Sessions
  - Brainstorming Techniques
- Different types of Modeling Techniques