Building the Analysis Model 1

Suradet Jitprapaikulsarn

Requirement Analysis

- Focus on "What" not "How"
- Model information, function and behavior
- User's point of view → Scenario-based models
- How data are transformed → Flow-oriented models
- Objects, attributes, and their relationships → Class-based models
- States of the system and its classes and the impact of events → Behavioral models

---

Objectives of Analysis models

- Describe the requirements
- Establish the foundation for design and implementation
- Define the verification and validation suites

Rules of Thumb

- Focus on visible requirements
- Increase the understanding
- Delay infrastructure considerations
- Minimize coupling, maximize cohesion
- Value of all stakeholders?
- KISS (Keep It Simple, Sweetie)

Derived from Roger S. Pressman, *Software Engineering: A Practitioner's Approach*, 6th Edition, McGraw-Hill, 2005

Domain Analysis

- Understand the background information so that we can understand the problem

Domain Analysis

- Acquiring the general information about the domain
- Mastering terminology
- Knowing the players and their attributes
- Getting to know the environment
- Examining the current practice
- Understanding generic versus specific
- Familiarizing with the competitors

Analysis Approaches

- Structured approach
  - Separate data from process

- Object-oriented approach
  - Data and process are considered together

Structured Analysis

- Data Dictionary

- Data model ⇨ ERD, Semantic object diagram (SOD)

- Flow-oriented model ⇨ Data flow diagram (DFD), Control-flow diagram

- Behavioral model ⇨ State diagram

- Scenario-based model ⇨ Process narrative

Data Modeling

- Analyzing data objects independently from processes
- Focusing on the data domain
- Be at the same abstraction level as stakeholder
- Pointing out the relationship among data objects

Data Modeling

- Data Objects
- Data Attributes
- Relationships

Data Objects

Data Object = something that is described by a set of attributes (data item) and that will be manipulated in the system (or software)

- **External entities** (printer, user, sensor)
- **Things** (report, display, signal)
- **Occurrences or events** (alarm, telephone call)
- **Roles** (manager, clerk)
- **Organization units** (Accounting Dept, R & D)
- **Places** (building, manufacturing floor)
- **Structures** (employee records)

Data Attribute

Aspect, quality, characteristic, or description of the object

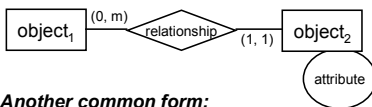| object: automobile |
| --- |
| attributes: |
| make |
| model |
| body type |
| price |
| options code |

Relationship

A "**fact**" that must be "**remembered**" by the system and cannot or is not computed or derived mechanically

- multiple relationships between two data objects are possible
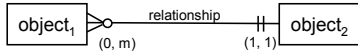- objects can be related in many different ways

---

ERD Notation

One common form:

Another common form:

Building an ERD

- Level 1—model all data objects (entities) and their "connections" to one another
- Level 2—model all entities and relationships
- Level 3—model all entities, relationships, and the attributes that provide further depth

Data Model: Example

Customer —(1,1)— places —(1,m)— request for service

request for service —(1,1)— generates —(1,n)— work order

standard task table —(1,1)— selected from —(1,w)— work tasks —(1,w)— consists of —(1,1)— work order
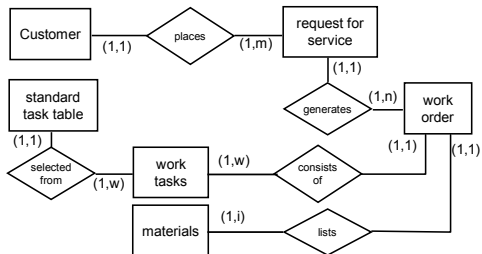
materials —(1,i)— lists —(1,1)— work order

Derived from Roger S. Pressman, *Software Engineering: A Practitioner's Approach*, 6th Edition, McGraw-Hill, 2005