Analysis Modeling

Instructor: Dr. Jerry Gao
Analysis Modeling

- Elements of the analysis model
- Data modeling
- Functional modeling and information flow
- Behavioral modeling
- Mechanics of structure analysis
  - Creating an ER diagram
  - Creating a data-flow model
  - Creating a control flow model
  - Control specification
- The data dictionary
- An overview of other classical analysis methods
  - Data structured system development
  - Jackson system development
  - SADT
**Analysis modeling**

Analysis models (a set of models) is the first set of technical representation.

Two types of analysis models: structured analysis, object-oriented analysis

Structured analysis is classical modeling method.
Descriptions about structured analysis by Tom Demarco[DEM79].
- The products of analysis must be highly maintainable.
- Use effective way to partition problems
- Use graphic notation whenever possible
- Differentiate logical and physical consideration
- Use some means of keeping track …
- New tools to describe logic and policy...

History of analysis modeling:
- In the late 1960s and early 1970s, early work in analysis modeling began
- “structured analysis” was coined by Douglas Ross in [DEM79].
- By the mid-1980s, deficiencies of structured analysis were criticized.
- Ward and Mellor, and others introduced the extensions for real-time in 1985.
- Later consistent notation [BU88] were suggested in 1988.
- CASE tools [YOU89] are used in 1989.
The Elements of the Analysis Model

Three primary objectives of the analysis model:
- to describe what the customer requires
- to establish a base for the creation of a software design
- to define a set of requirements that can be validated

The core - data dictionary
- contains the descriptions of all data objects in the software

The entity-relationship diagram (ERD)
- specify data attributes of data objects.
- depicts relationships between data objects.

The data flow diagram (DFD)
- provide an indication of how data are transformed as they move through the system.
- depicts the functions that transform the data flow.

The state-transition diagram (STD)
- indicate how the system behaves as a consequence of external events.
Data Modeling

Data modeling answers a set of specific questions:

- What are the primary data objects to be processed in the system?
- What is the composition of each data object?
- What attributes describe the object?
- What are the relationships between each object and other objects?
- What are the relationships between the objects and the related processes?

What problems will this solution address?

ERD model:

- ERD is a very useful method for data modeling.
- It focuses on data, representing a “data network” in the system.
- It represents:
  - data objects, object attributes, and relationships between objects

Figure. 12.2, 12.3, 12.4
Data Modeling

Cardinality:

- Specification of the number of occurrence of one (object) that can be related to the number of occurrences of another (object).

Three types:
- One-to-one (1:1)
- One-to-many (1:N)
- Many-to-many (M:N)

Modality to the object relationship pair.

Mandatory:
- specifies that the object is necessary for all situations to a relationship.

Optional:
- implies that there may be a situation in which the object is not necessary for a relationship.

Figure. 12.5, 12.6, 12.7, 12.8, 12.9
Functional Modeling and Information Flow

Information is transformed as it flows through a computer-based system.

Several models are used:
- Data flow diagrams (DFDs)  
  Figure 12.10, 12.11, and 12.12
- Extension for real-time systems
  - Ward and Mellow extension
  - Hately and Pirbhai extension

Ward and Mellow did their extensions for a real-time system:
- information flow that is gathered or produced on a time-continuous basis
- control information passed throughout the system and associated control processing.
- multiple instance of the same transformation with multitasking situations.
- system statuses and the mechanism that causes transition between states.

Figure 12.13, 12.14, 12.15, 12.16
The Mechanics of Structured Analysis

- Use ERDs to specify
  - the data objects (input, internal, and output data objects),
  - the attributes inside data objects
  - the relationships between them
  - their cardinality and modality

  In addition, add data object hierarchies, and data aggregation and association relationships. (Figure 12.22)

- Use DFDs to specify models of the information domain and function domain.

  Level 0 DFD - a fundamental system model (a context model)
  - represents the entire software as one process and its inputs and outputs

  Level 1 DFD - represents the data flow in terms of partitioned processes
  ..... (Figure 12.23, 12.24, 12.25)

- Use CFDs to specify models of the control flow of the system. (Figure 12.27)

- Use the CSPECs (such as STDs) to specify the state behavior of the system. (Figure 12.28)
Control Flow Specification

The control flow specification uses a control flow diagram (CFD) to represent the control flow in a system.

- a state-transition diagram (STD) - a sequence specification of behavior.  Figure 12.27
Behavior Models

The control specification (CSPEC) represents the behavior of the system.
- a state-transition diagram (STD) - a sequence specification of behavior. Figure 12.28

STD example.

Transition between states

Start state: $S_0$

Condition: Cond1: $T_1$

$S_0$ → $S_1$ → $S_2$

Condition: Cond2: $T_2$

$S_1$ → $S_3$ → $S_4$ → $S_5$

Conditions: Cond3: $T_3$, T4, T5, T6, T7

End state: $S_5$
Process Specification

The process specification (SPECS) is used to describe all flow model processes that appear at the final level of refinement.

Two representation methods:
- narrative text
- a program design language (PDL)
Data Dictionary

Why data dictionary:
- To provide an organized approach for representing the characteristics of each data object and control item.

What is data dictionary?
“The data dictionary is an organized listing of all data elements that are pertinent to the system, with precise, rigorous definitions so that both user and system analyst will have a common understanding of inputs, outputs, and components of stores, and even intermediate calculations.”

Data dictionary format:
- name
- alias
- where-used/how-used
- content description
- supplementary information (type, restrictions or limitations)

Composite data:
- as a sequence of data items
- as a selection from among a set of data items
- as a repeated grouping of data items

Figure 12.32