COMPUTER AIDED SOFTWARE ENGINEERING (CASE)

Dr. Ebadati
Kharazmi University
PROCESS MODELING

Chapter 5
KEY DEFINITIONS

• **Process model**
  • A *formal way* of representing how a *business system* operates
  • Illustrates the *activities* that are performed and how *data moves* among them

• **Data Flow Diagramming**
  • A *common technique* for creating process models
**KEY DEFINITIONS**

- *Logical* process models describe processes without suggesting how they are conducted.

- *Physical* process models provide information that is needed to build the system.
DATA FLOW DIAGRAMS
READING A DFD
(PATIENT APPOINTMENT)
4 ELEMENTS OF A DFD

1. Process
   - An **activity** or function performed for a specific business reason
   - Manual or computerized

2. Data flow
   - A **single piece of data** or a logical collection of data
   - Always starts or ends at a process
3. Data Store
   • A collection of data that is stored in some way
   • Data flowing out is retrieved from the data store
   • Data flowing in updates or is added to the data store

4. External entity
   • A person, organization, or system that is external to the system but interacts with it.
### Naming and Drawing DFD Elements

<table>
<thead>
<tr>
<th>Data Flow Diagram Element</th>
<th>Typical Computer-Aided Software Engineering Fields</th>
<th>Gane and Sarson Symbol</th>
<th>DeMarco and Yourdan Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Every process has A number A name (verb phase) A description One or more output data flows Usually one or more input data flows</td>
<td>Label (name) Type (process) Description (what is it) Process number Process description (Structured English) Notes</td>
<td>Name</td>
<td>Name</td>
</tr>
<tr>
<td>Every data flow has A name (a noun) A description One or more connections to a process</td>
<td>Label (name) Type (flow) Description Alias (another name) Composition (description of data elements) Notes</td>
<td>Name</td>
<td>Name</td>
</tr>
<tr>
<td>Every data store has A number A name (a noun) A description One or more input data flows Usually one or more output data flows</td>
<td>Label (name) Type (store) Description Alias (another name) Composition (description of data elements) Notes</td>
<td>D1 Name</td>
<td>D1 Name</td>
</tr>
<tr>
<td>Every external entity has A name (a noun) A description</td>
<td>Label (name) Type (entity) Description Alias (another name) Entity description Notes</td>
<td>Name</td>
<td>Name</td>
</tr>
</tbody>
</table>
USING A DFD TO DEFINE BUSINESS PROCESSES

• Business processes are too complex to be shown on a single DFD

• Decomposition is the process of representing the system in a hierarchy of DFD diagrams
  • Child diagrams show a portion of the parent diagram in greater detail
KEY DEFINITION

- **Balancing** involves insuring that information presented at one level of a DFD is accurately represented in the next level DFD.
RELATIONSHIP AMONG LEVELS OF DFDS

Context diagram

Level 0 diagram

Level 1 diagram

Level 2 diagram
• **First DFD** in every business process
• Shows the **context** into which the **business process fits**
• Shows the **overall business process** as just **one** process (process 0)
• Shows **all the external entities** that receive information from or contribute information to the system
LEVEL 0 DIAGRAM

• Shows **all the major processes** that comprise the **overall system** – the internal components of process 0
• Shows how the **major processes** are **interrelated** by data flows
• Shows **external entities** and the major processes with **which they interact**
• Adds **data stores**
LEVEL 1 DIAGRAMS

• Generally, one level 1 diagram is created for every major process on the level 0 diagram
• Shows all the internal processes that comprise a single process on the level 0 diagram
• Shows how information moves from and to each of these processes
• If a parent process is decomposed into, for example, three child processes, these three child processes wholly and completely make up the parent process
LEVEL 2 DIAGRAMS

• Shows **all processes** that **comprise a single process** on the level 1 diagram
• Shows **how information moves** from and to each of these processes
• Level 2 diagrams **may not be needed** for all level 1 processes
• Correctly numbering each process **helps the user understand** where the process fits into the overall system
ALTERNATIVE DATA FLOWS

• Where a process can produce different data flows given different conditions
• We show both data flows and use the process description to explain why they are alternatives
• Tip -- alternative data flows often accompany processes with IF statements
• At this point in the process it is easy to lose track of the “big picture”.

• Describe the difference between data flows, data stores, and processes.

• Describe in your own words the relationship between the DFD and the ultimate new application being developed.
PROCESS DESCRIPTIONS

• Text-based process descriptions provide more information about the process than the DFD alone

• If the logic underlying the process is quite complex, more detail may be needed in the form of
  • Structured English
  • Decision trees
  • Decision tables
CREATING DATA FLOW DIAGRAMS
INTEGRATING SCENARIO DESCRIPTIONS

- DFDs start with the use cases and requirements definition
- Generally, the DFDs integrate the use cases
- Names of use cases become processes
- Inputs and outputs become data flows
- “Small” data inputs and outputs are combined into a single flow
STEPS IN BUILDING DFDS

• Build the context diagram
• Create DFD fragments for each use case
• Organize DFD fragments into level 0 diagram
• Decompose level 0 processes into level 1 diagrams as needed; decompose level 1 processes into level 2 diagrams as needed; etc.
• Validate DFDs with user to ensure completeness and correctness
CREATING THE CONTEXT DIAGRAM

• Draw one process representing the entire system (process 0)
• Find all inputs and outputs listed at the top of the use cases that come from or go to external entities; draw as data flows
• Draw in external entities as the source or destination of the data flows
A CONTEXT DIAGRAM EXAMPLE (PATIENT-DOCTOR)
CREATING DFD FRAGMENTS

- Each use case is converted into one DFD fragment
- Number the process the same as the use case number
- Change process name into verb phrase
- Design the processes from the viewpoint of the organization running the system
CREATING DFD FRAGMENTS (CONT.)

• Add data flows to show use of data stores as sources and destinations of data

• Layouts typically place
  • processes in the center
  • inputs from the left
  • outputs to the right
  • stores beneath the processes
CREATING THE LEVEL 0 DIAGRAM

• Combine the set of DFD fragments into one diagram
• Generally move from top to bottom, left to right
• Minimize crossed lines
• Iterate as needed
  • DFDs are often drawn many times before being finished, even with very experienced systems analysts
A LEVEL 0 DFD EXAMPLE
CREATING LEVEL 1 DIAGRAMS (AND BELOW)

• Each use case is turned into its own DFD
• Take the steps listed on the use case and depict each as a process on the level 1 DFD
• Inputs and outputs listed on use case become data flows on DFD
• Include sources and destinations of data flows to processes and stores within the DFD
• May also include external entities for clarity
Creating Level 1 Diagrams (and Below)

• When to stop decomposing DFDs?
  • Ideally, a DFD has at least three processes and no more than seven to nine.
VALIDATING THE DFD

• Syntax errors – diagram follows the rules
  • Assure correct DFD structure

For each DFD:
  Check each process for:
  A unique name: action verb phrase; number; description
  At least one input data flow
  At least one output data flow
  Output data flow names usually different than input data flow names
  Between 3 and 7 processes per DFD
VALIDATING THE DFD (CONT.)

<table>
<thead>
<tr>
<th>For each DFD:</th>
<th>Check each <strong>data flow</strong> for:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A unique name: noun; description</td>
</tr>
<tr>
<td></td>
<td>Connects to at least one process</td>
</tr>
<tr>
<td></td>
<td>Shown in only one direction (no two-headed arrows)</td>
</tr>
<tr>
<td></td>
<td>A minimum number of crossed lines</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Check each <strong>data store</strong> for:</th>
</tr>
</thead>
<tbody>
<tr>
<td>A unique name: noun; description</td>
</tr>
<tr>
<td>At least one input data flow</td>
</tr>
<tr>
<td>At least one output data flow</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Check each <strong>external entity</strong> for:</th>
</tr>
</thead>
<tbody>
<tr>
<td>A unique name: noun; description</td>
</tr>
<tr>
<td>At least one input or output data flow</td>
</tr>
</tbody>
</table>
VALIDATING THE DFD (CONT.)

Across DFDs:

**Context Diagram:**
Every set of DFDs must have one Context Diagram

**Viewpoint:**
There is a consistent viewpoint for the entire set of DFDs

**Decomposition:**
Every process is wholly and complete described by the processes on its children DFDs

**Balance:**
Every data flow, data store, and external entity on a higher level DFD is shown on the lower level DFD that decomposes it.
No data stores or data flows appear on lower-level DFDs that do not appear on their parent DFD.
VALIDATING THE DFD (CONT.)

• Semantics errors – diagram conveys correct meaning
  • **Assure accuracy of DFD** relative to actual/desired business processes

• To **verify correct representation**, use
  • User walkthroughs
  • Role-play processes

• **Examine lowest level DFDs to ensure** consistent decomposition

• **Examine names carefully** to ensure consistent use of terms
A QUICK REVIEW OF DECOMPOSITION FOR CD SELECTIONS
LEVEL 0 DFD FOR CD SELECTIONS
INTERNET SYSTEM
LEVEL 1 DFD FOR CD SELECTIONS
PROCESS 1: TAKE REQUESTS
• The Data Flow Diagram (DFD) is an essential tool for creating formal descriptions of business processes.

• Use cases record the input, transformation, and output of business processes and are the basis for process models.

• Producing use cases and modeling business processes are critically important skills for the systems analyst to master.
موفق باشید