Data Flow Diagram Process

Sui Generis Team

Process for Data Flow Diagram

Process Documentation Template:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Title</td>
<td>Data Flow Diagram Process</td>
</tr>
<tr>
<td>Process #</td>
<td>CMPE202-5-Sui2</td>
</tr>
<tr>
<td>Date</td>
<td>September 29, 2006</td>
</tr>
<tr>
<td>Created/Modified By</td>
<td>Sui Generis Team</td>
</tr>
<tr>
<td>Rationale</td>
<td>The Data Flow Diagram is a tool for explaining business processes. It does so by showing the flow of data through a system.</td>
</tr>
</tbody>
</table>

Definitions

A) Data Flow Diagram

A data flow diagram explains business processes and activities in a clear, concise way by illustrating how data flows through the system from one process to another. It is a structured, diagrammatic technique representing external entities, logical storage, data sinks and data flows in the system.

B) Data Flow Diagram Principles

1. A system can be decomposed into subsystems, and subsystems can be further decomposed into lower level subsystems.
2. Each subsystem represents a process or activity in which data is processed.
3. At the lowest level, processes can no longer be decomposed.
4. Each ‘process’ has the characteristics of a system. A process must have input and output.
5. Data enters the system from the environment, data flows between processes within the system and data is produced as output from the system.

C) Data Flow Diagram Rules

1. Any data flow leaving a process must be based on data that are input to the process.
2. All data flows are named and the name reflects the data flowing between processes, data stores, sources, or sinks.
3. Only data needed to perform the process should be an input to the process.
4. A process should know nothing about, that is, be independent of, any other process in the system, it should
5. Processes are always running, they do not start or stop. Analysts should assume that a process is always ready to function or perform necessary work.

6. Output from processes can be an input data to other process, modified data such as change of status and change of content.

D) Data Flow Diagram Types

1. Physical Data Flow Diagram: Physical data flow diagrams are implementation-dependent and show the actual devices, department, people, etc., involved in the current system.

2. Logical or Conceptual Data Flow Diagram: Logical data flow diagram represents business functions or processes. It describes the system independently of how it is actually implemented, and focuses rather on how an activity is accomplished.

E) Data Flow Diagram Components and Corresponding Notations

1. External Entities: External Entities are sources and destinations of the system’s inputs and outputs.

2. Processes: A data process represents the transformation of data in the system modifying the inputs in the process of generating the outputs.

4. Data Flows: A data flow is a pipeline through which packets of information flow.
5. **Data Stores**: A data store represents either a temporary or permanent place where the data comes to rest.

F) **Data Flow Diagram Levels**
A Data Flow Diagram can be subdivided into several levels where the highest level is known as the **context level** or **top-level**. Each level can further be decomposed into subsystems until all of the processes and subsystems are identified.

G) **Data Flow Diagram Approach**
The Data Flow Diagram can be developed in one of the following two ways:

1. **Top-Down Approach**
   This type of approach starts by drawing a context-level diagram. The context-level diagram is then decomposed into sets of processes, data stores and data flows until the sufficient level of detail has been reached.

2. **Event Partitioning Approach**
The opposite of the top-down approach, the event partitioning approach starts by identifying subsystems from the lowest level. The subsystems are then aggregated into processes until the context-level DFD can be determined.

<table>
<thead>
<tr>
<th>Data Flow Diagram Process</th>
<th>The following are the different phases involved in the Data Flow Diagram Process.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Proposition</strong></td>
<td>1.1 The Project Manager initiates the proposition for the data flow diagram.</td>
</tr>
<tr>
<td></td>
<td>1.2 During this phase, the requestor requests for a data flow diagram of a system.</td>
</tr>
<tr>
<td><strong>2. Analysis</strong></td>
<td>2.1 The Analyst gathers the business requirements and identifies all possible functionalities of the system which include the various processes and the flows of information and material linking them to each other, to inventories and to various external agents of the system.</td>
</tr>
<tr>
<td></td>
<td>2.2 The Analyst creates a high level business design with the identified processes of the system.</td>
</tr>
<tr>
<td></td>
<td>2.3 The Analyst reviews the high level business design with the Project Manager. If the high level business design requires any revisions, go to <strong>step 2.1</strong>.</td>
</tr>
<tr>
<td></td>
<td>2.4 The Analyst conducts the high level business design walkthrough with the Technical Reviewer and the Developer.</td>
</tr>
<tr>
<td><strong>3. Implementation</strong></td>
<td></td>
</tr>
</tbody>
</table>
3.1 The Technical Reviewer along with the Developer determines the type of data flow diagram (Physical or Logical data flow diagram) to be used.
3.2 The Developer determines the approach (Top-Down or Event Partitioning approach) for implementing the data flow diagram process.
3.3 If top-down approach go to step 3.4 else if event partitioning approach go to step 3.5

3.4 Top-Down Approach
   3.4.1 Determine the context level of the Data Flow Diagram.
   3.4.2 Determine whether the subsystems inside the process are decomposable. If yes, go to step 3.4.3. If no, go to step 3.4.4.
   3.4.3 Decompose each subsystem into processes.
   3.4.4 Identify the processes, external entities, data stores and data flows inside each subsystem.
   3.4.5 If the lowest level of the system is reached covering all necessary details of the system, then go to step 3.4.7, else go to step 3.4.6.
   3.4.6 Go to step 3.4.2 to iterate through each processes identified inside a subsystem are decomposable or not.
   3.4.7 Draw the data flow diagram with the identified processes, external entities, data stores and data flows.

3.5 Event Partitioning Approach
   3.5.1 Identify every event in a process.
   3.5.2 If there are no more event inside the process go to step 3.5.4.
   3.5.3 Construct a process for each event and identify the processes, external entities, data flows and the data stores associated with the process.
   3.5.4 Aggregate group of related processes into a process of higher level.
   3.5.5 If the context level of the Data Flow Diagram is reached then go to step 3.5.6, else go to step 3.5.1.
   3.5.6 Draw the data flow diagram using event partitioning approach.

3.6 Submit the Data Flow Diagram to the Technical Reviewer for review.

4. Review
   4.1 The Technical Reviewer reviews in conjunction with the Developer reviews the data flow diagram to determine whether the data flow diagram rules are met and if there any revisions are needed.
   4.2 If revisions are needed, go to step 4.3 else go to step 4.4
   4.3 The Technical Reviewer requests the Developer to revise the data flow diagram (proceed to step 3).
   4.4 Once revisions are complete, the Technical Review forwards the data flow diagram to the Analyst (proceed to step 5).
5. Acceptance

5.1 The Analyst submits the completed data flow diagram to the Project Manager.
5.2 The Project Manager accepts the data flow diagram.

Roles & How

The development process of a Data Flow Diagram includes the following roles

Project Manager
- The Project Manager initiates the data flow diagram development process.
- The Project Manager reviews the high level business design of the data flow diagram along with the Analyst to determine whether all requirements are met.
- The Project Manager finally accepts the completed Data Flow Diagram.

Analyst
- The Analyst studies in depth the business requirements and creates a high level business design.
- The Analyst together with the Technical Reviewer reviews the high level business design.
- The Analyst then conducts a walk-through of the high level business design with the Technical Reviewer and Developer.

Developer
- The Developer is responsible for determining the data flow diagram approach.
- The Developer determines the processes and other components that will comprise the data flow diagram. Once these have been determined, the Developer then models and draws the data flow diagram.
- The Developer is responsible for revising the data flow diagram if any revisions are needed upon review with the Technical Reviewer.

Technical Reviewer
- The Technical Reviewer determines the type of data flow diagram to be used in conjunction with the Developer.
- The Technical Reviewer reviews the completed data flow diagram according to the data flow diagram rules and determines whether all the functionalities of the system are covered or not.
- If any revisions are needed, the Technical Reviewer instructs the developer to revise the data flow diagram. Otherwise the Technical Reviewer forwards the data flow diagram to the Analyst.

When

A Data Flow Diagram is prepared whenever any of below scenario occur:
- business process re-engineering
- introduction of new technologies/systems
- enhancement of the existing business processes/system

Regulations

Costs
<table>
<thead>
<tr>
<th>Role</th>
<th>Hours</th>
<th>Rate</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Project Manager</td>
<td>2</td>
<td>$160</td>
<td>$320</td>
</tr>
<tr>
<td>1 Analyst</td>
<td>8</td>
<td>$32</td>
<td>$256</td>
</tr>
<tr>
<td>1 Developer</td>
<td>16</td>
<td>$40</td>
<td>$640</td>
</tr>
<tr>
<td>1 Technical Reviewer</td>
<td>8</td>
<td>$60</td>
<td>$480</td>
</tr>
</tbody>
</table>

Total Estimated Costs: $1696

<table>
<thead>
<tr>
<th>Feature</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Checklists</td>
<td>None</td>
</tr>
<tr>
<td>Forms</td>
<td>None</td>
</tr>
<tr>
<td>Notes</td>
<td>None</td>
</tr>
</tbody>
</table>
Mapping the Graphical Inspection Process to Roles

This document describes the activities of each person with a role in the data flow diagram process for each phase.

**PROJECT MANAGER:**

*Proposition Step:*
- Request data flow diagram of the target system.

*Analysis Step:*
- Review the high level business design of the target system.
- Recommend any necessary changes.

*Acceptance Step:*
- Accept the completed data flow diagrams of the requested target system.

**ANALYST:**

*Analysis Step:*
- Understand the business requirements of the target system.
- Identify the business processes of the target system.
- Prepare a high level business design.
- Present and review the completed high level business process design with the Project Manager.
- If any corrections are needed, revise the high level business process design.

*Implementation Step:*
- Conduct a walkthrough with the Technical Reviewer and Developer to go over the requirements and high level design.

*Acceptance Step:*
- Receive the completed data flow diagrams of the target system from the Technical Reviewer.
- Present the completed data flow diagrams of the target system to the Project Manager for signoff.

**TECHNICAL REVIEWER:**

*Implementation Step:*
- Attend the high level design walkthrough presented by the Analyst.
- Discuss with Developer what type of data flow diagram needs to be created for the target system - either Physical data flow diagram or Logical data flow diagram.
Review Step:
- Review the completed data flow diagrams presented by the Developer and verify if the completed data flow diagrams are compliant to the general data flow rules.
- Recommend any necessary changes. If any, forward the completed data flow diagrams to Analyst.

DEVELOPER:
Implementation Step:
- Attend the high level design walkthrough presented by the Analyst.
- Discuss with Technical Reviewer what type of data flow diagram needs to be created for the target system - either Physical data flow diagram or Logical data flow diagram.
- Identify the data flow diagram modeling approach to use - either top-down approach or event-partitioning approach.
- If using the top-down approach, identify context-level processes and corresponding subsystems until the bottom-most processes have been identified. Alternatively, if using the event-partitioning approach, identify the bottom-most processes and subsystems until context-level view is reached.
- Determine the processes, external entities, data stores and data flows of every identified process
- Draw the data flow diagrams corresponding to these identified elements.

Review Step:
- Present the completed data flow diagrams to Technical Reviewer for review.
- If any revisions are needed, modify the data flow diagrams.
## COMPARISON
(Data Flow Diagram)

### Techniques for Comparison

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Symbolic Notation</th>
<th>Map the Graphical Process to Roles</th>
<th>Process Documentation Template</th>
<th>UML Activity Diagram</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documentable</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tailorable or Scalable</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Measurable</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>