 Detailed Process Definitions;
The Data Dictionary

- The data dictionary is a database or repository of information about objects identified during systems development.
- Every object (and each of its components) must have a definition in the data dictionary.
- The data dictionary is a major source of documentation about the information system.

Data Dictionary Entries for Components of DFDs

- The data dictionary must contain precise definitions of all components of all data flow diagrams:
  - To fully explain the meaning of the DFDs.
  - To describe the contents of all data flows and data stores.
  - To describe the processing that occurs in primitive processes.
  - To ensure that names and meanings of components are used consistently (a common vocabulary).

Data Dictionary Entries

- A data dictionary entry must be included for each:
  - Data flow.
  - Data store.
  - Higher level process.
  - Primitive process.
  - External agent (source/sink).

Data elements

- Each data flow consists of a series of data elements.
  - A data element is a unit of data that cannot be further broken down into meaningful units of data.
  - Each data element should also have an entry in the data dictionary.
  - Data flows and data stores are made up of data elements.

Data Dictionary - Data element entry

<table>
<thead>
<tr>
<th>ACADEMIC</th>
<th>CONSULTANCY</th>
<th>AUTHOR: David Ross</th>
<th>DATE: 14 Oct 2002</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT ELEMENT:</td>
<td></td>
<td>Product code</td>
<td></td>
</tr>
</tbody>
</table>

- Alias: Inventory number, product number
- Description: Number to identify and differentiate each product held in warehouse.
- Values: Must be a positive integer.
- Range: 00001 to 99999.
Data Flows

- A data dictionary entry for a data flow describes the sequence of data elements and data structures in the data flow using the following connectors:
  - \( = \) is equivalent to
  - \( + \) and
  - \( [ ] \) select one of
  - \( \{ \} \) iterations of
  - \( ( ) \) optional
  - \( * \) comments
  
Data Dictionary - data flow entry

- **ACADEMIC CONSULTANCY**
- **DATA FLOW:** Sales order
- **Alias:** Customer order
- **Description:** Request for product from customer
- **Composition:**
  - sales order no. +
  - sales order date +
  - customer number +
  - (account customer cash customer) +
  - customer name +
  - customer address +
  - (customer telephone no) +
  - \( \{\text{item no} + \text{item desc} + \text{item price} + \text{item qty}\} \) +

Data Stores

- A data store is made up of data flows and data elements
- Where a data store consists of a collection of data flows it is described as repetitions of that data flow
- E.g.: Data Store: \{customer invoice\}

Data Dictionary - data store entry

- **ACADEMIC CONSULTANCY**
- **DATA STORE:** Sales
- **Description:** Store of all pertinent data about product sales made to customers
- **Composition:**
  - sales = date +
  - (product product quantity) +
  - sales total (customer)

Describing Processes

- Each process in higher level DFDs is defined by the DFD that decomposes the process at the next level down: these are parent processes
- Each such process should have a data dictionary entry which includes a brief description of the overall nature and purpose of the process

Describing Processes

- Example data dictionary entry for a process

  **Treat patients:**
  Patient consultations are carried out to determine the causes of patients' illnesses/medical problems. Further treatment/ follow up is recommended if appropriate. Details of consultations are recorded.

- Specific process description (minispecs)
Data Dictionary - data store entry

**ACADEMIC CONSULTANCY**

**DATA PROCESS** Treat patient

**Description:** Patient consultations are carried out to determine the causes of patients’ illnesses/medical problems. Further treatment/ follow up is recommended if appropriate. Details of consultations are recorded, and fees charged to the patient on the basis of these sessions.

**Data required:** (Patient)+ Fee schedule

**AUTHOR:** David Ross

**DATE:** 10 Oct 2002

---

Describing External Agents

- each external agent (source or sink) should have a data dictionary entry which describes its relationship with the system

- e.g.

  **Referring Doctors:**
  
  These are doctors who refer their patients to a specialist medical practitioner for treatment. They are usually general practitioners.

---

Data Dictionary - external agent entry

**ACADEMIC CONSULTANCY**

**DATA AGENT:** Referring doctor

**Alias:** Referrer, GP

**Description:** Doctors who refer their patients for treatment.

**Values:** Must be a positive integer

**Range:** 00001 to 99999

**AUTHOR:** David Ross

**DATE:** 14 June 2002

---

Building and Maintaining the Data Dictionary

- determine standard formats and information content for all types of data dictionary entries
- have a standard means of organising and storing the entries in the data dictionary
- ensure that all components of the DFDs have entries in the data dictionary and that they are kept up-to-date
- cross-referencing of entries in the data dictionary can help to check the completeness and consistency of the DFDs and other types of models

---

Detailed Process Definitions

- the processing that occurs within the bottom level (primitive) processes in DFDs needs to be defined

- detailed process descriptions are also known as minispecs

- detailed process descriptions form part of the data dictionary: they define the contents of primitive processes

---

Detailed Process Definitions

- many techniques can be used to define the details of processing:

  - e.g.
  
  - narrative text
  
  - Structured English
  
  - decision tables
  
  - decision trees
  
  - flow charts
Detailed Process Definitions

detailed process descriptions should:

- express what the process does (i.e. policy), not how the process is carried out (i.e. procedure)
- be in a form that can be easily understood and verified by both users and systems analysts
- be in a form that can be easily communicated to all potential stakeholders:
  e.g. end-users, systems analysts, managers, system designers, project leaders, programmers

Structured English

Structured English is a modified form of English with some major restrictions on vocabulary and structure:

- only action (imperative) verbs such as get, put, add, calculate, find, delete are used
- only nouns/noun phrases which refer to components of the DFDs should be used, i.e. data flows, data stores, data elements

Structured English

- sentences consist of action verbs and DFD components
- sentences are combined to form process descriptions using the control structures of sequence, condition, and repetition

Control Structures

- Condition uses If... Then...Else or Select Case Case 1...Case 2...End Case
- E.g.
  
  if qty-in-stock is less than minimum-order-qty
  then update stock-reorder indicator
  else do nothing

Data Dictionary -data store entry

<table>
<thead>
<tr>
<th>ACADEMIC CONSULTANCY</th>
<th>AUTHOR: David Ross</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stock reorder</td>
<td>DATE: 10 Oct 2002</td>
</tr>
</tbody>
</table>

- Description: Current levels of stock in warehouse are matched to optimal requirements. Shortfalls are ordered immediately.
- Select Case
  - CASE 1 qty-in-stock is less than minimum-order-qty
    do update stock-reorder indicator
  - CASE 2 qty-in-stock is equal to minimum-order-qty
    do nothing
  - CASE 3 qty-in-stock is greater than minimum-order-qty
    do nothing

End Case
Control Structures
- Sequence is represented with one sentence following another in sequence:
  - Add student to class list
  - Decrease available-places
  - Calculate class-fee

Control Structures
- Repetition uses Do-Until or Do-While loops:
  - Do
  - Accept customer-account-details
  - Calculate daily-interest = daily balance * daily interest rate
  - Add daily-interest to monthly-interest-due
  - Until no more customer-accounts

Example Structured English
Accept sales-order
Find customer-details
If customer-details not found
  Then reject sales-order
Else
  Create sales-order-header
  Do while more sales-order-items
    find item-details
    calculate sales-order-item price = item price * order-qty
  Enddo
  Authorise sales-order
Endif

Guidelines for Structured English
- use indentation to indicate control structures and their scope: assists readability and understanding
- avoid more than three levels of nesting: complicated logic can be represented using other techniques
- Structured English descriptions should illustrate the logic of the processing, not the implementation of the processing

Decision Tables
- decision tables are useful for describing processes where several different conditions apply and the particular actions that are taken are determined by combinations of the values of the conditions
- decision tables are useful where the process logic is complex
- decision tables show all the possible choices and the conditions they depend on in a tabular form

Decision Tables
decision tables have three stubs (four quadrants):

<table>
<thead>
<tr>
<th>conditions</th>
<th>rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>actions</td>
<td></td>
</tr>
</tbody>
</table>

Combinations of condition values
Outcomes for each set of condition values
Example Decision Table

<table>
<thead>
<tr>
<th>Condition</th>
<th>Y</th>
<th>Y</th>
<th>Y</th>
<th>N</th>
<th>N</th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>avg account bal &gt; $1,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>overdraft amount &lt; $50,000</td>
<td></td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>previous paid-out loan</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>approve</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>conditional approval</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reject</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Decision Trees

decision trees are an alternative graphical representation of a decision situation as a connected series of nodes and branches

### Determine Customer Discount

- wholesale customer: 15%
- retail customer: 12%
- local item: 10%
- imported item: 7%

Selecting Techniques for Process Descriptions

- Structured English is useful where a process has a sequence of activities and there is no more than three levels of nesting of decisions.
- Decision trees and decision tables are useful where a process involves a decision based on combinations of values of several conditions.

Overview of Process Modelling

- Several techniques are available for representing the processing within systems.
- The aim of process modelling in systems analysis is to define the processing that occurs within a system.
References