Algebraic Notation

The symbols used are:

- = sign, meaning “consists of”.
- + sign, meaning “and”.
- Braces {} meaning repetitive elements, a repeating element or group of elements.
- Brackets [] for an either/or situation.
- The elements listed inside are mutually exclusive.
- Parentheses () for an optional element.

Data Structure Example

Customer Order = Customer Number + Customer Name + Address +Telephone + Catalog Number + Order Date +{Order Items} + Merchandise Total + (Tax) +Shipping and Handling + Order Total + Method of Payment +(Credit Card Type) + (Credit Card Number) +(Expiration Date)

Data dictionary components

Specific contents in a data dictionary can vary. In general, these components are various types of metadata, providing information about data.

- Data object listings (names and definitions)
- Data element properties (such as data type, unique identifiers, size, nullability, indexes and optionality)
- Entity Relationship (ER)Diagrams
- System-level diagrams
- Reference data
- Missing data and quality-indicator codes
- Business rules (such as for validation of data quality and schema objects)

Example

- Address = House No + (Street / Area) + City + State
- Course ID = Course Number + Course Name + Course Level + Course Grades

Data Store

Datastores hold information for later use, like a file of documents from which necessary data can be retrieved for processing. Data inputs flow from the external entity to a data store through a process, while data outputs flow out of a data store to the entity through a process. It is to be noted that data flow can never be possible between the entity and the data store. There are two variants of data storage - it can either be represented as a rectangle with the absence of both smaller sides or as an open-ended rectangle with only one side missing.

Data Flow

Data Flow refers to a path by which the system’s information passes from external entities to datastores through processes. With arrows along with brief labels, the DFD presents the direction of the data flow. In general, data flow is not possible in between two processes or in between two data sources. However, data movement is shown from the base of the arrow as its source towards the head of the arrow as the destination.

6.3.2 Construction of DFD

Certain standard guidelines are to be followed while creating a DFD. These include the following:

a) DFD notations should be assigned meaningful names by using the proper part of speech. For example, a process name should be a verb whereas nouns should be used for naming external entities, data storage, and data flow.

b) Each process should be numbered uniquely, and numbering should be consistent. For example, the single process in the context diagram should be numbered with 0. Subsequently, processes in 1st level DFD to be numerically assigned as 0.1, 0.2, … so on.

c) Abbreviations should be avoided in DFD notations.

d) Looping concept can’t be presented in DFD.

e) A DFD refinement should be continued in a series of levels of DFDs until each process performs a simple operation.

f) The data store should not be depicted in context diagram or Level 0 DFD, but all the data stores required for a system must be included in 1st level or next level DFDs.

g) Each process should have at least one input and one output.

h) Each data store should have at least one data flow in and one data flow out.

i) Data flow should not be given in between two data stores.

Levels of Data Flow Diagrams

There are various levels of DFD ranging from simple overviews to complex, granular representations of a system or process with deeper levels, starting with level 0. These provide details about the input, processes, and outputs of a system. The most common and intuitive DFDs are level 0 DFDs, which are also known as context diagrams.

Level 0 DFD or Context Diagram

Level 0 DFD focuses on high-level system process or function and the data sources that flow to or from them. The entire system is represented by a single bubble (circle), and all the external entities that will feed the data into the system are shown in the context diagram. They are designed to be simple, straightforward overviews of a process or system. It is to be noteworthy that no data store should be included in the context diagram.

Level 1 DFD

Level 1 DFD elaborates the system by splitting the system's single process (bubble) in level 0 DFD into more detailed form showing all the high-level functions of the system and flow of data among various functions. It also includes all the data stores required for the system. However, external entities should not be a part of level 1 DFD or other deeper level DFDs.

Level 2 DFD

The level 2 DFDs present more elaboration by splitting each process of level 1 DFD into granular subprocesses (if required). Decomposing the bubble is called factoring or exploding. Each bubble can be factorized into from 3 to 7 bubbles.

- ID - identification number.
- Label, the text that should appear on the diagram.
- A general description of the data flow.
- The source of the data flow which may be an external entity, a process, or a data flow coming from a data store.
- The destination of the data flow.
- Type of data flow, either a record entering or leaving a file.
- The name of the data structure or elements.
- An area for further comments and notations about the data flow.

When developing programs that use the data model, a data dictionary can be consulted to understand where a data item fits in the structure, what values it may contain and what the data item means in real-world terms. For example, a bank or group of banks could model the data objects involved in consumer banking. They can provide a data dictionary for a bank's programmer. The data dictionary would describe each of the data items in its data model for consumer banking, such as "Account holder" and "Available credit."

A data dictionary stores the following information:

Name	Description
Name	Name generally includes primary name of all composite data or control item available, name of external entity or data store.
Aliases	Any other word used in place of Name
Where or How it's used?	Data dictionary generally gives information about where and how data or control items are used which may include as an input/ output to process.
Description	A notation for representing content

- Data structures are a group of smaller structures and elements.
- An algebraic notation is used to represent the data structure.

6.4 DATA DICTIONARY

The first step in analysing a system of interactive objects is to identify each one and its relationship to other objects. This process is called data modelling and results in a picture of object relationships. After each data object or item is given a descriptive name, its relationship is described, or it becomes part of some structure that implicitly describes the relationship. The type of data such as text or image or binary value is described, possible predefined default values are listed, and a brief textual description is provided in the data dictionary.

A data dictionary is a collection of descriptions of the data objects or items in a data model for the benefit of programmers and others who need to refer to them. It is basically a centralized structured repository of data elements in the system. It stores the descriptions of all DFD data elements, i.e., details and definitions of data flows, data stores, data stored in data stores, and the processes. It is often referenced as a meta-data (data about data) repository. It is created along with the DFD model of the software program and is expected to be updated whenever DFD is changed or updated.

It makes it easier for users and analysts to use data properly as well as to understand the format of data objects. As a centralized collection of information about data, data dictionary stores the meaning and origin of data, its relationship with other data, the data format for usage, etc.

Data Dictionary Uses:

The data dictionary may be used for the following reasons:

- Provide documentation.
- Eliminate redundancy.
- Validate the data flow diagram.
- Provide a starting point for developing screens and reports.
- To develop the logic for DFD processes.

Each data flow should be defined with descriptive information and its composite structure or elements.It may include the following information:

Level 3 DFD

Level 3 and higher-numbered DFDs are usually not to be defined. It is mainly due to the requirements of a large amount of detail which increases the complexity of the system. Higher-level DFDs can be transformed into more specific lower-level DFDs with a deeper level of understanding.

Balancing a DFD

The data flow into or out from a process in Level 1 DFD must be matched with the subprocesses in Level 2 DFD or deeper level DFDs. It is called balancing a DFD.

Synchronous and asynchronous process

When the execution of one process depends on the data coming out from another process directly (without any data store), then that process is known as synchronous process. Otherwise, the process is termed as asynchronous where data flow is maintained between two processes via a data store.

Let us consider an example to explain DFD for a system “Online Examination System”. Online Examination System allows the user (examinee) to appear in an examination in online mode by sitting in an examination venue. This system is handled by two types of users: administrator and examinee. Accordingly, two external entities representing the admin and examinee involved in this system are shown in level 0 DFD (context diagram). The context diagram of this system is presented in figure 6.2.

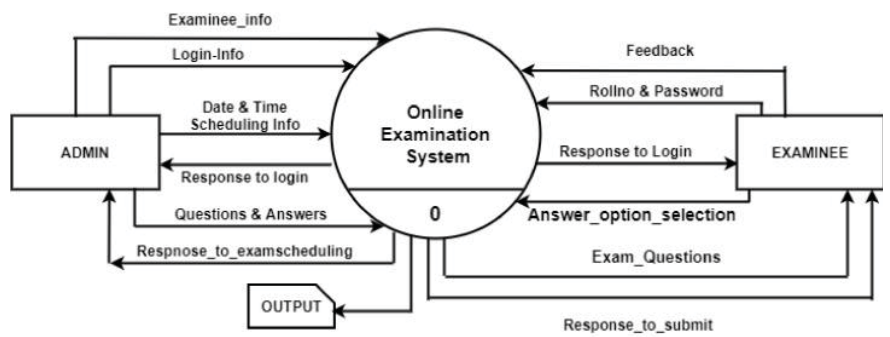


Figure 6.2 Context Diagram of Online Examination system

Further, the bubble in the context diagram is decomposed into Level 1 DFD for Admin and Examinee separately concerning their functionalities in the system. Figure 6.3 and figure 6.4 depict level 1 DFD for Admin and Examinee respectively. Processes in level 1 DFD for Examinee act as synchronous because of their inter dependability as shown in figure 6.4.

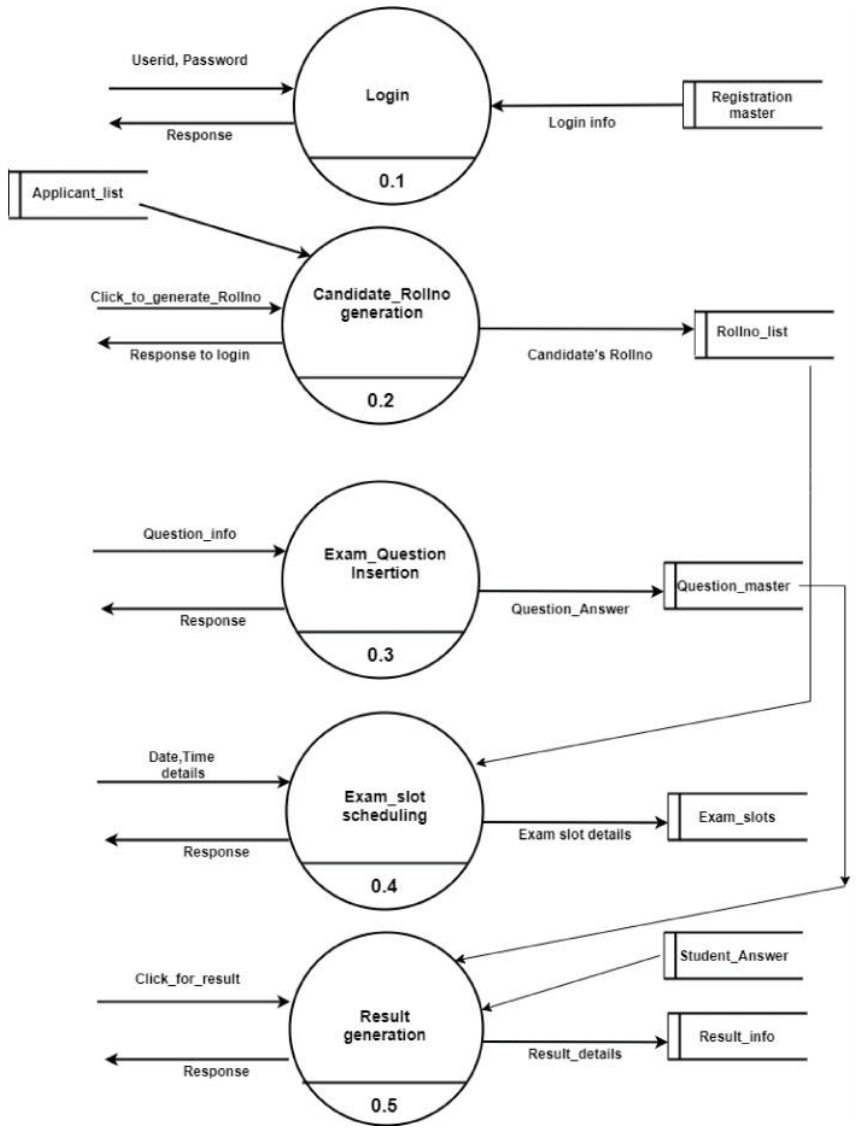


Figure 6.3Level 1 DFD for ADMIN

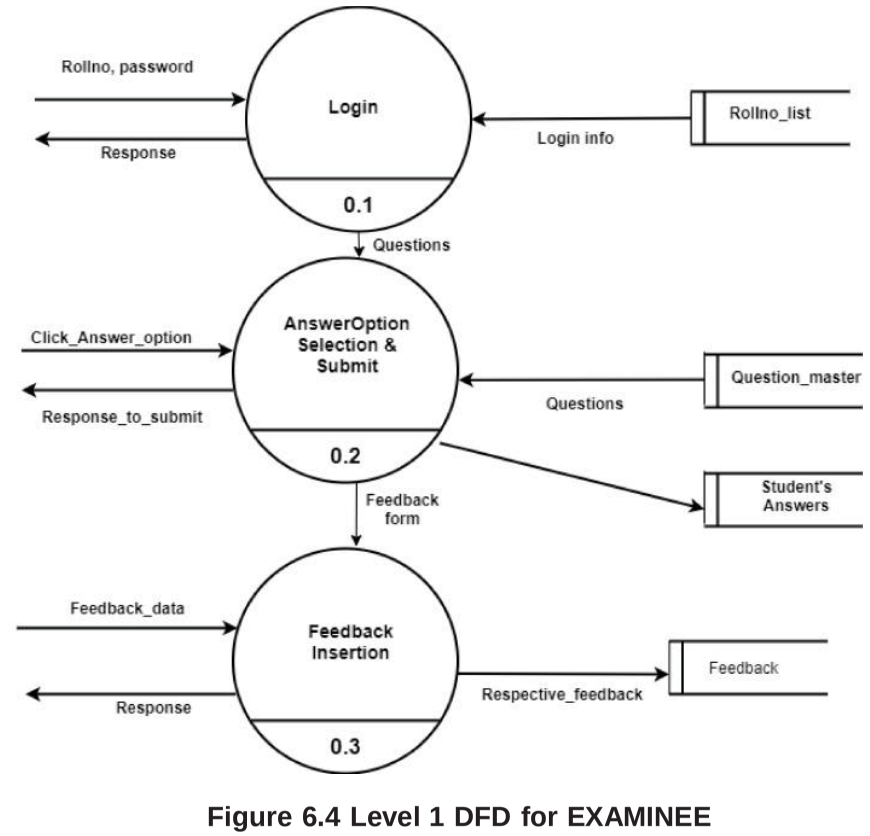


Figure 6.4 Level 1 DFD for EXAMINEE



CHECK YOUR PROGRESS

Q.1: Choose the correct options from the followings:

i) Which of the following is not a component in DFD?
   - A. Entities
   - B. Attributes
   - C. Process
   - D. Data Flow

ii) A data flow can
   - A. Only emanate from an external entity
   - B. Only terminate in an external entity
   - C. May emanate and terminate in an external entity
   - D. May either emanate or terminate in an external entity but not both