The Frontier School was established in 1990 with the objective of providing quality education to children from all walks of life. At present about 1000 students are studying at the school in various classes from class 1 to O-levels. The handling of such a large number of students is getting increasingly difficult. The management has decided to automate the fee receipt system. The fee receipt system must provide information in a format compatible with the student registration system which is already installed at the school. The system should be flexible and should allow for changes in fee structures and rates.

At present the fee receipt procedure is as follows:

At the beginning of every month, a student takes his fee card to the fee receipt clerk. This fee card contains his registration number, his name and class information. The fee receipt clerk uses the registration number to retrieve the student’s record from the student register. This register contains the fee amount for the student (under different categories) e.g. tuition fee, lab charges, bus charges (incase the student avails the school’s transport facilities). These amounts are added to compute the total fee for the student.

After the total fee has been calculated, the fine register is consulted to check whether there is any additional amount to be paid by the student in the form of fines. This amount is added to the total fee, and a fee slip is issued to the student who must take this slip to the school bank for payment.
Understanding the System

- What steps/activities are carried out
- Where and by whom
- What is the resulting information
- For whom is the information useful

For each activity:
- What is the input to the activity
- What/who triggers the activity
- What is the frequency of the activity
- What procedures convert input to output
- What areas require specific control
- What special conditions effect the way a step is carried out
- What is the volume of data
- What is the output
- For whom is the output useful
While asking the questions, analyze whether

- a process can be simplified
- performance can be improved
- better checks/controls can be implemented
- errors can be reduced
- better information can be provided
Gaining System Understanding – Some questions that help

- What steps/activities are carried out
- Where and by whom
- What is the resulting information
- For whom is the information useful

The questions above help in establishing an overall understanding of the system’s working and its objectives. For each activity, further information can be gathered by asking the following:

- What is the input to the activity
- What data items are of interest and how are they structured
- What/who triggers the activity
  - Sometimes, through software, we can initiate an activity automatically. e.g. Whenever items are sold, the inventory is updated to reflect the change. The inventory update activity is thus triggered by the sales activity and can be carried out automatically without human intervention. This can result in reduction of errors.
- What is the frequency of the activity
  - Frequent activities may need to be more efficient. If 100 forms are to be filled in a day, and it takes a user 10 minutes to fill a form, the process will take 100*10=1000 minutes i.e. > 16 hours to fill the forms. If we have an 8 hour day, one operator is not sufficient for the task. Another option is to design the form so that it takes less time to fill.
- What procedures convert input to output
- What areas require specific control
  - Checks and controls are helpful in reducing errors and ensuring correct working of the system e.g. if a 10% discount is to be given for amount > 100,000/- this condition must be understood and implemented.
- What special conditions effect the way a step is carried out
  - Sometimes, special checks may be required for out-of-the ordinary circumstances e.g. an order of amount > 100,000/- must be accompanied by an advance of at least 5%
- What is the volume of data
  - This question helps in determining the hardware/software requirements
- What is the output
- For whom is the output useful
  - This person should be interviewed for suggestions on what may be more useful, what is missing in the current output etc.

While asking the questions, analyze whether

- a process can be simplified
- performance can be improved
- better checks/controls can be implemented
- errors can be reduced
- better information can be provided
**DFD Characteristics**

- Focuses on flow of data
- Shows ‘what’ and not ‘how’
- Constructed top down
- Partitioned
- Does not show sequence explicitly
- Does not show iterations explicitly
- Does not show conditions explicitly

**Context Diagram**

- Contains only one process/bubble representing domain of study
- Contains net inputs/outputs of the system and external entities

**1st Level DFD (Intermediate Levels)**

- Refinement of the 0 level DFD, partitioning takes place
- Data stores are introduced
- Data Flows must be balanced between levels
- Process names are different between levels

**Bottom Level DFD (Primitive Level)**

- Bubbles are ‘zoomed in’
- Data Flows must be balanced between levels
- Process names are different between levels
Customers send their orders by post. Each customer order is checked for general legibility by the validation clerk. Orders which are wrong (illegible, incomplete etc.) are send to the trouble shooter. Valid orders are passed to credit control, where they are divided into cash orders and credit orders. Cash orders are send to order processing. Credit orders are sent to credit control, where customer’s credit is checked. Customers without satisfactory credit have their orders sent to the trouble shooter. Customers with ok credit have their orders send to order processing. This department checks that each item ordered is in stock. Any order that includes an item not in stock is put in the back orders file. All other orders are sent to sales. Sales gets the items ordered, makes up a shipping package and sends the package plus a shipping list to the customer