Guidelines for Drawing Dataflow Diagrams

- Naming conventions:
  - Processes: strong verbs
  - Data-flows: nouns
  - Data-stores: nouns
  - external entities: nouns
- No more than 7 - 9 processes in each DFD.
- Data-flows must begin, end, or both begin & end with a process.
- Data-flows must not be split.
- A process is not an analog of a decision in a systems or programming flowchart. Hence, a dataflow should not be a control signal. Control signals are modeled separately as control flows.
- Loops are not allowed.
- A dataflow cannot be an input signal. If such a signal is necessary, then it must be a part of the description of the process, and such process must be so labeled. Input signals as well as their effect on the behavior of the system are incorporated in the behavioral model (say, state transition graphs) of the information system.
- Decisions and iterative controls are part of process description rather than data-flows.
- If an external entity appears more than once on the same DFD, then a diagonal line is added to the north-west corner of the rectangle (representing such entity).
- Data-flows that carry a whole record between a data-store and a process is not labeled in the textbook since there is no ambiguity. This is also not a universal convention. I would rather you labeled such data-flows explicitly.

- **Conservation Principles:**

  Data-stores & Data-flows: Data-stores cannot create (or destroy) any data. What comes out of a data-store therefore must first have got into a datastore through a process.
  Processes: Processes cannot create data out of thin air. Processes can only manipulate data they have received from data-flows. Data outflows from processes therefore must be derivable from the data inflows into such processes.

- **Labeling Conventions:**

  Numbering: The system under study in the context diagram is given number `0'. The processes in the top level DFD are labeled consecutively by natural numbers beginning with 1. When a process is exploded in a lower level DFD, the processes in such lower level DFD are consecutively numbered following the label of such parent process ending with a period or full-stop (for example 1.2, 1.2.3, etc.).

  Balancing: The set of DFDs pertaining to a system must be balanced in the sense that corresponding to each dataflow beginning or ending at a process there should be an identical dataflow in the exploded DFD.

  Data-stores: Data-stores may be local to a specific level in the set of DFDs. A data-store is used only if it is referenced by more than one process. External entities: Lower level DFDs cannot introduce new external entities. The context diagram must therefore show all external entities with which the system under study interacts. In order not to clutter higher level DFDs, detailed interactions of processes with external entities are often shown in lower level DFDs but not in the higher level ones. In this case, there will be some data-flows at lower level DFDs that do not appear in the higher level DFDs. In order to facilitate unambiguous balancing of DFDs, such data-flows are crossed out to indicate that they are not to be considered in balancing. This convention of crossing is quite popular, but this text does not follow it. I would rather you followed this convention.