Foundations of specification
Part I: The operational style

Topics:
– Notations and methods for writing specifications
  – Examples of operational notations
  • Data flow diagrams
  • State diagrams
  • Petri nets

Specification qualities

Precise, clear, unambiguous
Consistent
Complete
  – internal completeness
  – external completeness
Incremental
  – helpful during analysis if one can make a "first cut" and then incrementally refine
  – document is structured; can be understood in increments

Specification styles

Degree of formality:
– Formal specification:
  • is a mathematical entity
  • process for constructing is grounded in mathematics
– Informal specification:
  • less rigorous, but often more amenable to visualization
  • e.g., UML diagrams

Operational:
– Behavior specification in terms of some abstract machine

Descriptive
– Behavior described in terms of properties

Example: Operational specification of an ellipse

Ellipse is a geometric figure that can be drawn as follows:
1. Select two points P1 and P2 on a plane
2. Get a string of a certain length and fix its ends to P1 and P2
3. Position a pencil as shown in next figure
4. Move the pen clockwise, keeping the string tightly stretched, until you reach the point where you started drawing

Example: Descriptive specification of an ellipse

Ellipse is a two-dimensional closed curve, the points \((x, y)\) of which are solutions to the following equation:

\[ ax^2 + by^2 + c = 0 \]

where \(a\), \(b\), and \(c\) are suitable constants in \(R\)

Implementing the specification of a ellipse
Another example

"Let a be an array of n elements. The result of its sorting is an array b of n elements such that the first element of b is the minimum of a (if several elements of a have the same value, any one of them is acceptable); the second element of b is the minimum of the array of n-1 elements obtained from a by removing its minimum element; and so on until all n elements of a have been removed."

"Result of sorting array a is an array b which is a permutation of a and is sorted."

Data Flow Diagrams (DFDs)

- Notation for writing semi-formal operational specifications
  - Graphical notation
  - Well-suited for analyzing/specifying traditional information-systems problems
- System modeled as collection of data, which are manipulated by "functions"
  - Data can be persistent (stored in data repositories)
  - Data can flow between/among functions
  - Each function can be systematically decomposed into a more refined DFD

Graphical notation

- bubbles represent functions
- arcs represent data flows
- open boxes represent persistent store
- closed boxes represent I/O interaction

Example: DFD for simple calculation

Example: Compiler

A construction “method” (1)

1. Start from the "context" diagram
A construction “method” (2)

2. Proceed by refinements until you reach “elementary” functions (preserve balancing).

Refinement of “Get a book”

- Book
- Author
- Title
- List of shelves
- List of books borrowed
- Details of book requested by the user

Patient monitoring systems

The purpose is to monitor the patients’ vital factors—blood pressure, temperature, ..., reading them at specified frequencies from analog devices and storing readings in a DB. If readings fall outside the range specified for patient or device, an alarm must be sent to a nurse. The system also provides reports.

More refinement

- Limits
- Pressure, pulse
- Pulse
- Social monitoring
- Patient data
- Recent data
- limits violations
- Patient nurse
- Nurse
- Social monitoring
- Patient data
- Limits for patient
- Limits
- Temperature
- Block data
- Date

Note: Adapted from Ghezzi, Jazayeri, Mandrioli, and Stirewalt.
Evaluation of DFDs (1)

Easy to read
Informal semantics
- How to define "most refined" functions?
- Inherent ambiguities

Are outputs from A, B, C all needed for D to compute?
Are outputs to E and F produced at the “same time?”

Evaluation of DFDs (2)

Control information is absent

Possible interpretations:
(a) A produces datum, waits until B consumes it
(b) B can read the datum many times without consuming it
(c) a pipe is inserted between A and B