SADT — Structured Analysis & Design Technique

Yuling Li

12/5/16
How to Make a Pizza?
Workshop: SADT: Structured Analysis & Design Technique
How to Make a Pizza (Process/Activities) Systematically?

- **Analysis**: Determine what the system will do
- **Design**: Define subsystems and their interfaces
- **Implementation**: Create the subsystem independently
- **Integration**: Combine the subsystems into a single unit
- **Testing**: Verify the system workings
- **Installation**: Make the system operational
- **Operation**: Use the system
“M models A if M answers questions about A”

---- Douglas T. Ross
Contents

1. Overview of SADT
2. Syntax and Semantics
3. SADT Diagrams
4. Cases
Overview of SADT

SADT

IDEF0

IDEF Family
Emergent

1950s
“hierarchic, layered modular system”

1960s
“system of systems for building systems”
“top-down hierarchic decomposition”

1970s
“Hori’s Human-directed activity cell model”

“language for blueprinting systems”

- Douglas T. Ross developed SADT (1969-1973)
- SADT is a trademark of SofTech, Inc. (After 1973)
Develop

- During the 1970s, the U.S. Air Force Program for Integrated Computer Aided Manufacturing (ICAM) sought to increase manufacturing productivity through systematic application of computer technology.

\[ \text{1970s} \quad \text{government (US Department of Defense) version IDEF0 introduced} \]

\[ \text{1980s} \quad \text{“under the name of IDEF0, SADT has been used in military and industrial organisations”} \]

\[ \text{1990s} \quad \text{renamed in 1999 as Integration DEFinition} \]

IDEF family of methods
IDEF Family

- **IDEF0**: Function modeling
- **IDEF1**: Information modeling
- **IDEF1X**: Data modeling
- **IDEF2**: Simulation model design
- **IDEF3**: Process description capture
- **IDEF4**: Object-oriented design
- **IDEF5**: Ontology description capture
- **IDEF6**: Design rationale capture
- **IDEF7**: Information system auditing
- **IDEF8**: User interface modeling
- **IDEF9**: Business constraint discovery
- **IDEF10**: Implementation architecture modeling
- **IDEF11**: Information artifact modeling
- **IDEF12**: Organization modeling
- **IDEF13**: Three schema mapping design
- **IDEF14**: Network design
Structured Analysis and Design Technique, is a **graphical language for describing systems together with a methodology** for producing such descriptions.

An SADT model contains a **set of diagrams that describe a system** from an identified **viewpoint** and for a **particular purpose**.

SADT as a language and as a methodology directs and disciplines **the analysis and design of systems**. (Dickover, 1977)

Structured Analysis and Design Technique, is a **graphical notation and an approach** to system description. (David A. Marca, 1988)
IDEF0 models comprising system functions (actions, processes, operations), functional relationships, and the data and objects that support systems analysis and design, enterprise analysis, and business process re-engineering.

Integrated Computer-Aided Manufacturing (ICAM) Function Modeling
2 Syntax and Semantics

- **Syntax**
  - The structure components and features of language and the rules that define relationships among them.
  - Boxes represent functions, defined as activities, processes or transformations.
  - Arrows represent data or objects related to functions.

- **Semantics**
  - Semantics refers to the meaning of syntactic components of a language and aids correctness of interpretation.
Boxes and Arrows

- Function name is a verb or a verb phrase.
- A box number is shown.

- Straight line arrow segment
- Curved arrow segment; corners are rounded with 90 degree arcs
- Forking arrows
- Joining arrows

Fig. Box and Arrow Syntax (Standard, 1993)
Workshop: SADT: Structured Analysis & Design Technique

Box and Arrow Semantics
—— SA box

Under control, input is transformed or consumed into output by the mechanism.
Labels and Names

- A function name shall be an active verb or verb phrase, such as:
  - process parts, plan resources, conduct review, monitor performance,
  - design system provide maintenance, develop detail design, fabricate component, inspect part…

- The arrows shall be labeled with a noun or noun phrase, such as:
  - specifications, test report, budget, design requirements, detail design,
  - directive, design engineer, board assembly, requirements…
Example

![Diagram showing the process of performing detailed design with inputs and outputs.]

- **Preliminary Design Data**
- **Design Requirements**
- **Recommended Detailed Design**
- **PERFORM DETAILED DESIGN**
- **Design Engineer**
- **MFG/A631**
SADT Diagrams

- Types of Diagram
- Diagram Features
- Diagram Syntax Rules
- Diagram Reference Expressions
- Models
Type of Diagram

Top-Level Context Diagram

- Subject of model represented by single box with bounding arrows.
- Called A-0 ("A minus zero")
- Box and arrows are very general
- Sets model scope or boundary and orientation.
- Should include
  - Purpose
  - Viewpoint
Example

PURPOSE: The assessment, planning, and streamlining of information management functions.
VIEWPOINT: The Information Integration Assessment Team.

QA/A-0  MANAGE INFORMATION RESOURCES
Child Diagram

- Single process in Context Diagram (A-0) may be decomposed into sub-processes and modeled in a child (A0) diagram.
- Each process in the A0 diagram may be decomposed further into sub-processes and modeled in (grand-) child (A1, A2, … A6) diagrams.
- Each (grand-) child process may be decomposed further into sub-processes and modeling (great-grand-) child diagrams.

Parent Diagram

- A parent diagram is one that contains one or more parent box.
This box is the parent of this diagram.

NOTE: Node numbers shown here indicate that the box has been detailed. The C-number or page number of the child diagram could have been used instead of the node number.
Diagram Features

Arrow as Constraints

- Arrow as Constraints
  - and this data/object provided by function 1

- Performance of function 3 requires this data/object output from function 2...

Arrow as Pipelines

- Concurrent Operation
  - Once this data/object is provided, functions 2 and 3 may operate concurrently.

- Pipeline A splits into B and C to provide controls to X and Y.
Workshop: SADT: Structured Analysis & Design Technique

Arrow Between Boxes

This fork means that “Files” contains “Customer Records” (needed by Function 2) and “Price & Tax Tables” (needed by Function 3).

This join means “Account Entries” are created by “Deliver Products” and "Do Billings".

Ordered Product

Account Clerk

Invoices
Boundary Arrows and Coding

• Boundary arrows on an ordinary graphic diagram represent the inputs, controls, outputs, or mechanisms of the diagram’s parent box.

• All boundary arrows on a child diagram (except for tunneled arrows) shall correspond to the arrows that connect to its parent box.
Tunneled Arrows

• Provide information at a specific level of decomposition that is not required for understanding at some other levels.
Diagram Syntax Rules

- Diagrams have boxes and arrows
- Box represent activities
- Boxes have dominance
- Arrows represent things
- Arrows represent interconnections among boxes
- Arrows are collections of things: branches and joins
Diagram Reference Expressions

...  Optional higher-level context diagrams
A-1  Optional context diagram
A-0  Required top-level context diagram
(contains A0 top box)
A0 Top level child diagram
A1, A2, ..., A6  Child diagrams
A11, A12, ..., A16, ..., A61, ..., A66  Child diagrams
A111, A112, ..., A161, ..., A611, ..., A666  Child diagrams
...  Lower-level child diagrams

Fig. Negative Node-Numbered Context (Standard, 1993)
Example

A0 Manufacture Product
   A1 Plan For Manufacture
      A11 Identify Manufacturing Methods
      A12 Estimate Requirements, Time, Cost to Produce
      A13 Develop Production Plans
      A14 Develop Support Activities Plan
   A2 Make and Administer Schedules and Budgets
      A21 Develop Master Schedule
      A22 Develop Coordinating Schedule
      A23 Estimate Costs & Make Budgets
      A24 Monitor Performance To Schedule & Budget
   A3 Plan Production
Models

- A model answers questions
- A model has a single subject
- A model has one viewpoint
- Models are coordinated sets of diagrams
- A system is represented by a single box
- Identifying decompositions with node numbers
- Linking decomposition with C-number
Deposition

“M is a model of S if M can be used to answer questions about S with an accuracy of A”
Different Viewpoints

Diagram = Whole
Box = Part
Arrow = Interface
Boxes and arrows form a complete network.

Fig. Two interconnected models of from different viewpoints. (Dickover, 1977)
Call Notation

- A call arrow is a special arrow of mechanism;
- The caller box does not have its own child diagram to detail it;
- But rather is detailed by another box in the same or another model;
- Multiple caller boxes may call the same box.
Example

The decomposition of Model X is continued in Model Y; Model Y "support" model X

Situation 1: Model X calling Model Y

Situation 2: a box in Model X calling a box in Model Y

Fig. The call notation. (Dickover, 1977)