

## Object table

Object	Space	Comment
Central computer	P	
NC blocks	S	
Numerical control	P	
Software	S	For NC
NC program file	S	Composed of NC blocks
Control words	S	For position and special functions
Functions	P	Activities performed by the machine servosystem
Position control commands	S	
Special control commands	S	
Machine servosystem	P	
Operator commands	S	
Keyboard interface	S	
Operator	P	
CRT	P	
NC program	S	Synonymous with NC program file

P = problem space. S = solution space

## Object-operation table

Object	Space	Operations
Central computer	P	Transmits
NC blocks	S	Read, store. read-from-NC-program-file, decompose. insert-into-existing-file.
Numerical control	P	Contains
Software	S	Reads, stores, display, decompose. process, encode, sent
NC program file	S	Stores-in, read-from, display, execute
Control words	S	Processed, encoded
Functions	P	Decomposed-into
Position control commands	S	Sent-to-servosystem
Special control commands	S	Sent-to-servosystem
Machine servosystem	P	
Operator commands	S	Are-input-via-a-keyboard
Keyboard interface	S	
Operator	P	
CRT	P	
NC program	S	Synonymous with NC program file

P = problem space. S = solution space

### Object-operation table—refined

Object	Space	Operations
NC blocks	S	Read-from-central-computer Read-from-NC-program-file Decompose-into-control-words
NC program file	S	Insert-into-existing-file Stores-in Read-from Display-on-CRT Execute
Control words	S	Processed Encoded
Position control commands	S	Send-to-servosystem
Special control commands	S	Send-to-servosystem
Operator commands	S	Are-input-via-a-keyboard

P = problem space, S = solution space

### Object-operation table with attributes

Object	Space	Attribute	Operation	Space	Attribute
Condition	S	Out-of-bounds	Produces	S	—
Alarm	S	Multipitch	Produces	S	—
			Transmitted	S	Immediately
Station	P	Operator	Transmitted	S	—

PACKAGE program-component-name IS  
TYPE specification of data objects

PROC specification of related operations

PRIVATE

data structure details for objects

PACKAGE BODY program-component-name IS  
PROC operation.1 (interface description) IS

END

PROC operation.n (interface description) IS

END

END program-component-name

PACKAGE nc-program-file IS  
  TYPE nc.program.file structure  
  PROC insert-into-existing-file  
  PROC stores-in  
  PROC display-on-CRT  
  PROC read-from  
  PROC execute  
END nc-program-file

PACKAGE nc-blocks IS  
  TYPE nc.block.structure  
  PROC read-from  
  PROC decompose  
END nc-blocks

PACKAGE control-words IS  
  TYPE control-words  
  PROC process  
  PROC encode  
END control-words

PACKAGE commands IS  
  TYPE operator-command  
  TYPE control-command  
  PROC send  
  PROC input  
END commands

All NC blocks, regardless of origination, are read and placed into an nc-block-buffer. The point of origin for an NC block is indicated with an origination-pointer. The buffer is read and appropriate follow-on processing occurs.

```
PACKAGE nc-blocks IS
  TYPE nc-block-structure
  TYPE nc-block-buffer
  TYPE origination-pointer
  PROC read-from
  PROC decompose
END nc-blocks
```

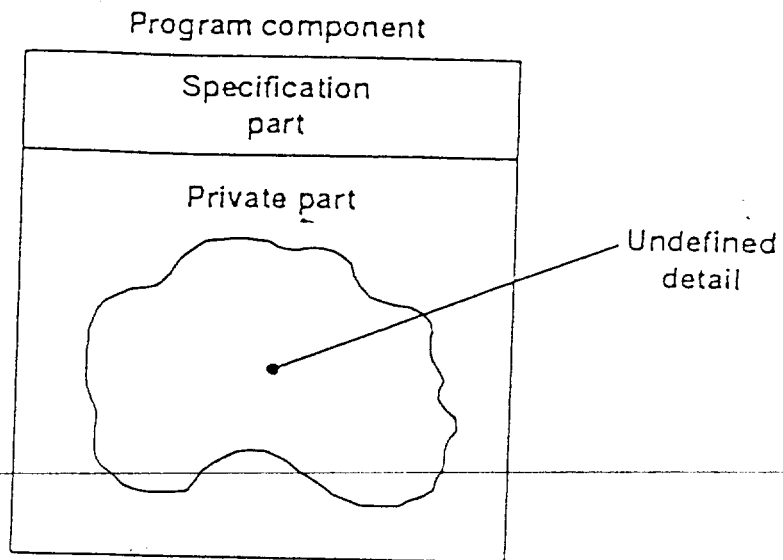


FIGURE 9.5  
Graphical representation for a program component.

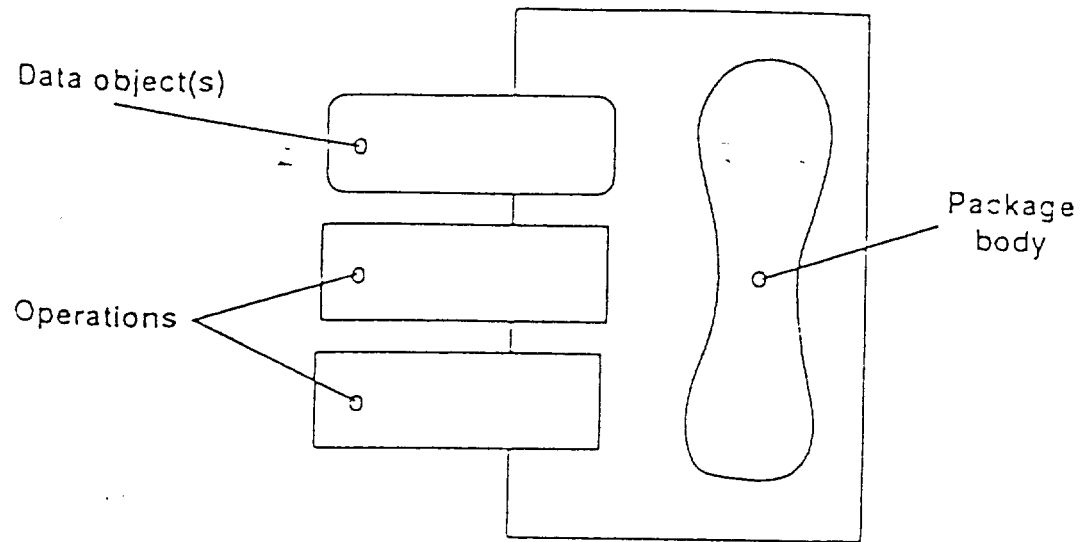


FIGURE 9.6  
Package (object) notation.

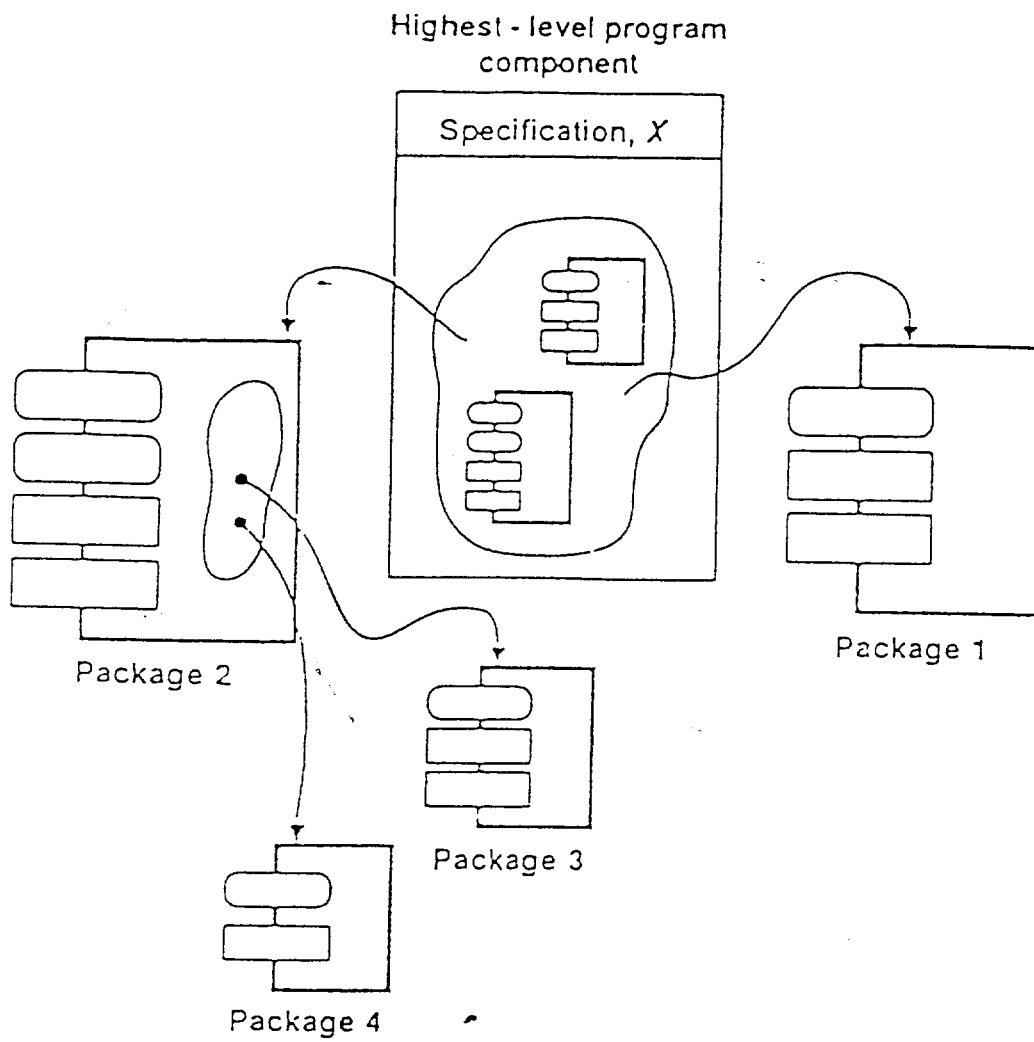


FIGURE 9.7  
Program components and interfaces, and Booch diagram [BOO83].



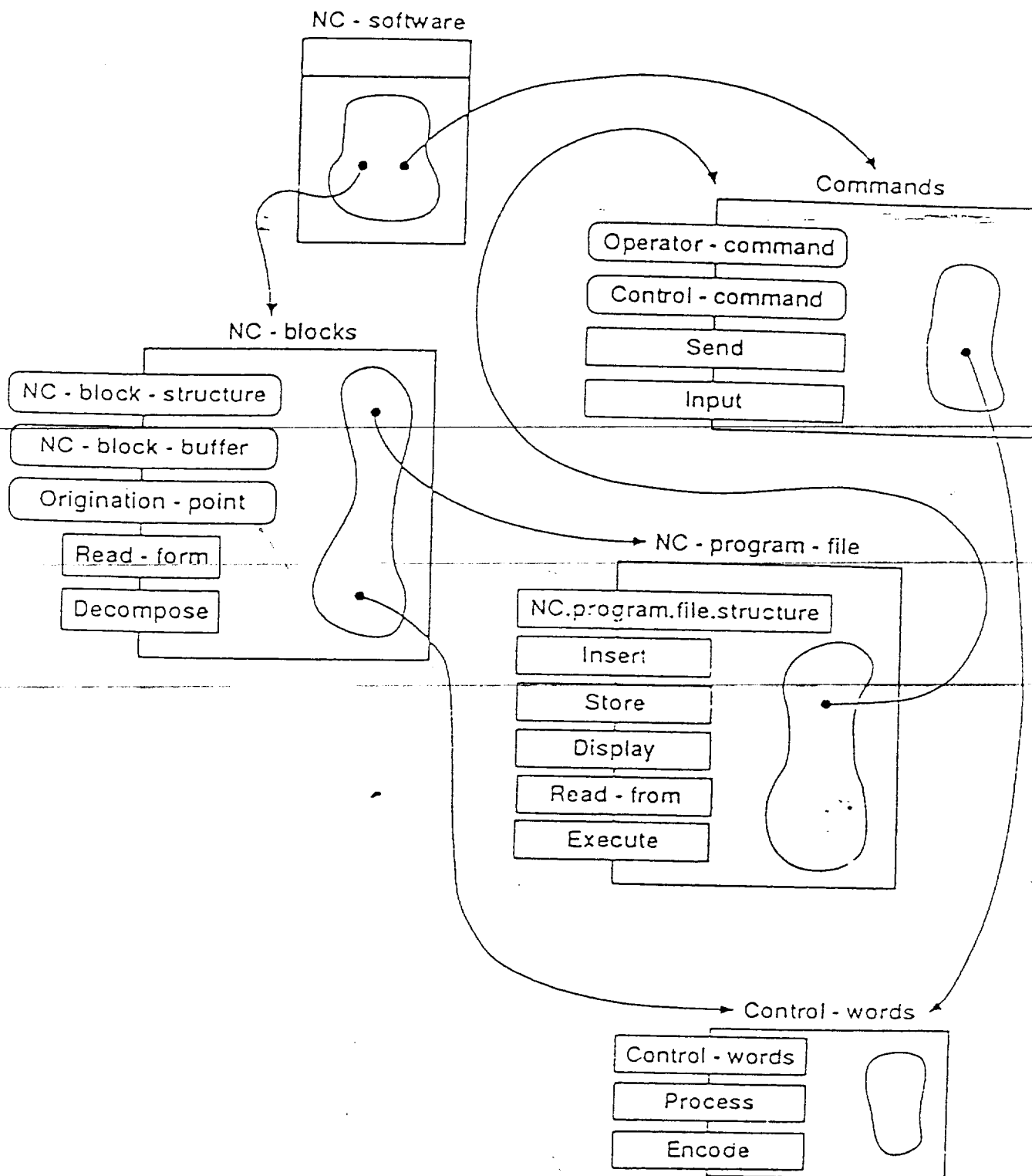


FIGURE 9.8  
NC software design representation.

PACKAGE program-component-name IS  
TYPE specification of data objects

PROC specification of related operations

PRIVATE

data structure details for objects

PACKAGE BODY program-component-name IS

PROC operation.1 (interface description) IS

END

PROC operation.n (interface description) IS

END

END program-component-name

PACKAGE control-words IS

TYPE control-words

PROC process

PROC encode

PRIVATE

... control-words data structure

PACKAGE BODY control-words IS

PROC process

... procedural detail

END

PROC encode

... procedural detail

END

END control-words

```
PACKAGE control-words IS
  TYPE control-words IS PRIVATE;
  PROC process (nc-block: IN ; control-words: OUT);
  PROC encode (control-words: IN; command: OUT);
PRIVATE
  TYPE control-words IS STRUCTURE DEFINED
    word-type IS STRING LENGTH (1);
    value IS INTEGER;
  END control-words TYPE;
END control-words
```

## PACKAGE BODY control words IS

```
PROC encode (control-words: IN; commands: OUT);
  -- a control word can take on values that range
  -- from  $-100.00 \leq x, y \leq +100.00$  where command
  -- value 0 corresponds to  $-100.0$  and command value
  --  $+100.0$  corresponds to 32678.
  TYPE scale-factor IS SCALAR FLOAT;
  TYPE s-min, s-max, c-min, c-max IS INTEGER;
  command [1] = control-words [word-type];
  s-max := 100;
  s-min := -100;
  c-max := 32678;
  c-min := 0;
  scale-factor := float.convert ((s-max - s-min)/
                                (s-max - s-min));
  IF control-words [word-type] <> "x" OR
     control-words [word-type] <> "y"
     THEN commands [2..] = control-word [value];
     ELSE commands [2..] = scale-factor *
                           control-words [value];
  ENDIF
END encode
```

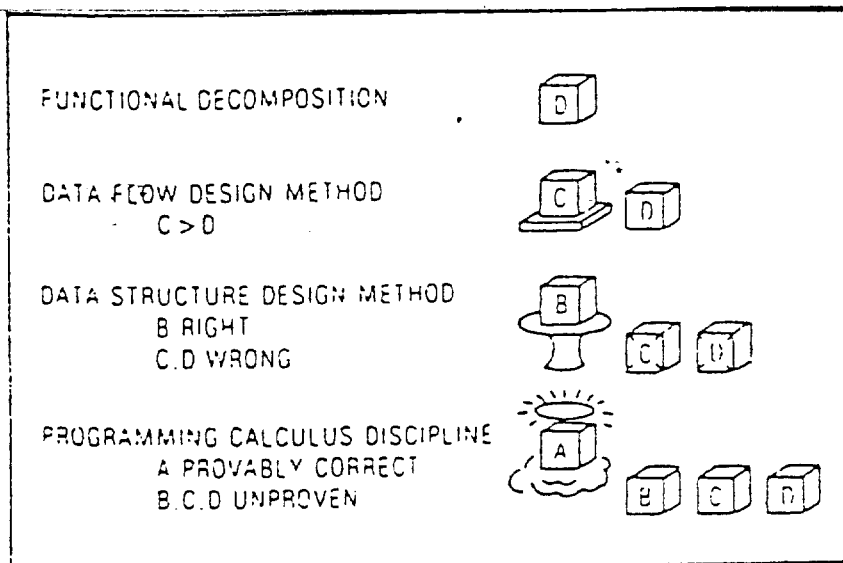


Figure 43. Summary of program design methodology claims.

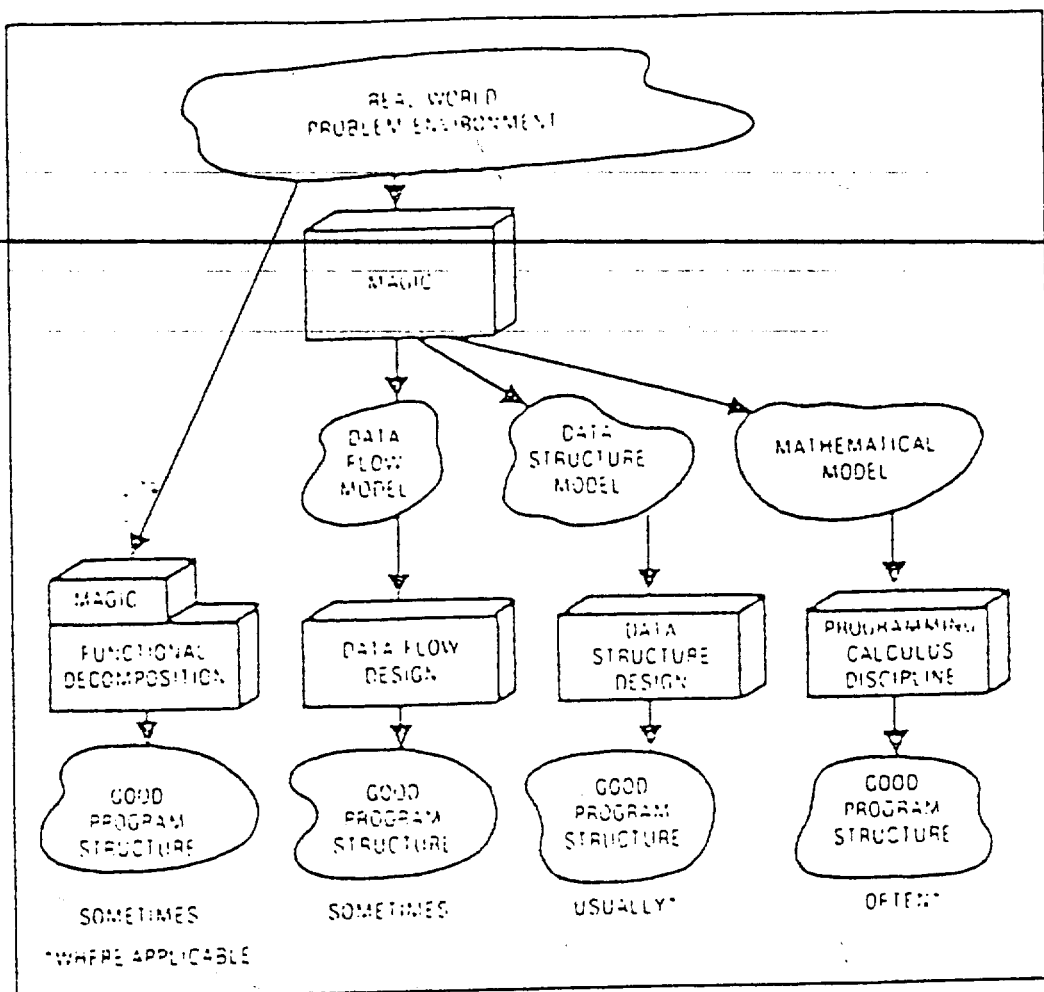


Figure 44. Current state of the art.

---

# PROGRAM UNIT TEST

---

- O FUNCTIONAL
- O PERFORMANCE
- O STRESS
- O STRUCTURAL



9

1000

2000

# BLACK BOX TESTING

- o FUNCTIONAL
- o PERFORMANCE
- o STRESS

# EQUIVALENCE PARTITION

- o GUIDELINES
- o EXAMPLES

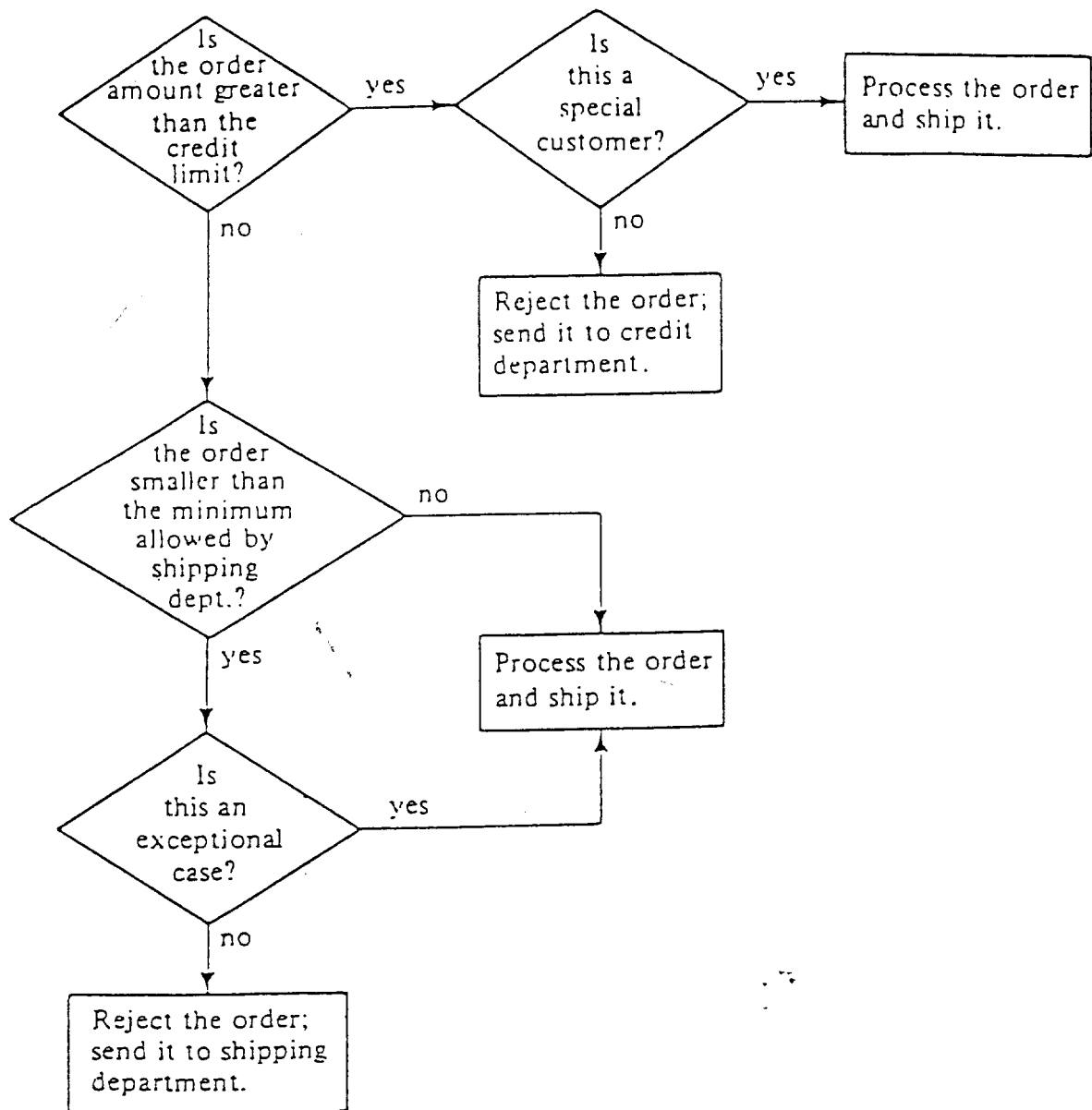
# BOUNDARY VALUE ANALYSIS

## OTHER TESTING TECHNIQUES

- o CAUSE - EFFECT
- o DATA VALIDATION
- o DELIMITER VALIDATION
- o PROGRAM CORRECTNESS

## DECISION TABLES

CAUSE-EFFECT GRAPHING IS A TECHNIQUE  
PROVIDES CONCISE REPRESENTATION OF  
LOGICAL CONDITIONS/CORRESPONDING ACTIONS



## Rules

Conditions and Actions	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1. Dollar amount of order exceeds credit limit.	F	F	F	F	F	F	F	F	T	T	T	T	T	T	T	T
2. Customer has special approval from credit dept.	F	F	F	F	T	T	T	T	F	F	F	F	T	T	T	T
3. Size of order is less than minimum allowed.	F	F	T	T	F	F	T	T	F	F	T	T	F	F	T	T
4. Customer has special approval from shipping dept.	F	T	F	T	F	T	F	T	F	T	F	T	F	T	F	T
1. REJECT ORDER, SEND TO CREDIT DEPT.									X	X	X	X				
2. REJECT ORDER, SEND TO SHIPPING DEPT.			X				X				X				X	
3. PROCESS ORDER, AND SHIP IT.	X	X		X	X	X		X					X	X		X



Conditions and Actions	Rules					
	1	2	3	4	5	6
1. Dollar amount of order exceeds credit limit.	T	-	F	T	F	T
2. Customer has special approval from credit dept.	F	-	-	T	-	T
3. Size of order is less than minimum allowed.	-	T	F	F	T	T
4. Customer has special approval from shipping dept.	-	F	-	-	T	T
1. REJECT ORDER, SEND TO CREDIT DEPT.	X					
2. REJECT ORDER, SEND TO SHIPPING DEPT.		X				
3. PROCESS ORDER, AND SHIP IT.			X	X	X	X