A VEHICLE FOR DEVELOPING STANDARDS FOR SIMULATION PROGRAMMING

Henry Kleine

Jet Propulsion Laboratory

ABSTRACT

The objective of SDDL is to provide an effective communications medium to support the design and documentation of complex software applications. This objective is met by providing (1) a processor which can express design specifications in an intelligible, informative, machine-reproducible document, (2) a design and documentation language with forms and syntax that are simple, unrestrictive, and communicative, and (3) methodology for effective use of the language and processor.

The application of SDDL to the specific problems of simulation models is discussed, with emphasis on the potential of SDDL for developing and specifying design and documentation standards for simulation and modeling.

INTRODUCTION

Effective communication is an essential part of science and engineering. Although secondary in importance to the original creative thinking, without the ability to communicate creative thoughts effectively, progress in science would be impossible. Computer program development especially requires effective communication since the resultant product, aptly named "software," consists entirely of large volumes of complex creative thoughts.

SOFTWARE DEVELOPMENT TEAM COMMUNICATIONS

A complex software project usually involves many team members and many different kinds of communication links. Figure 1 identifies these team members and shows the many links over which information must flow. As suggested in the diagram, programming languages are satisfactory for only a few of the links. Older programming languages were barely suitable for the programmer-machine link, while modern languages which provide Structured Programming capability [1] are becoming very effective for this and also for part of the programmer-programmer communication needs.

The remaining links, which must convey such information as the program's justification, functional requirements, design/documentation, and development status, also have a critical need for an effective communication capability. This need is being met by the Software Design and Documentation Language (SDDL) [2] and other processors [3].

SOFTWARE DESIGN AND DOCUMENTATION LANGUAGE

The SDDL approach supports effective communication for software design and documentation by providing:

(1) A processor which can express design specifications in an intelligble, informative, machine reproducible Software Design Document (SDD).

(2) A program design and documentation language with forms and syntax that are simple, unrestrictive and communicative.

(3) A methodology for effective use of the language and the processor.

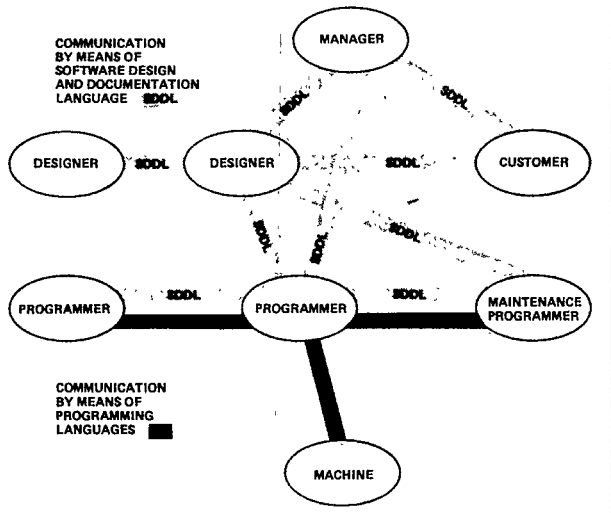
The purpose of the SDDL processor is to translate the designer's creative thinking into an effective communications document. The processor must perform as many automatic functions as possible, thereby freeing the designer's energy for the creative design effort. Many new automatic functions have been added to the processor's capability and more are being discovered through continuing SDDL applications.

The SDDL syntax is the means by which the designer communicates the design to the SDDL processor. The syntax is comprised of keywords, used to invoke design structures, and a collection of directives which provide the user with control of processor actions such as indentation, page width, start of a new page, etc.

The third component of the SDDL approach is the methodology for using the language and

This paper presents the results of one phase of research carried out at the Jet Propulsion Laboratory, California Institute of Technology, under Contract NAS 7-100, sponsored by the National Aeronautics and Space Administration.

FIGURE I



SOFTWARE DEVELOPMENT TEAM COMMUNICATIONS

the processor to express program design concepts in lucid, meaningful, precise terms. Since SDDL functions by operating on certain keywords, any or all of which the user may select, complete freedom of creative expression is provided, and therefore the methodology is a valuable guideline. Most of the existing SDDL syntax and processor functions were implemented to support the methodology developed while SDDL was being used on two simulation programs [4].

STANDARDS FOR SOFTWARE DESIGN AND DOCUMENTATION

Effective communication of software design can be further enhanced by adopting standards or conventions for using systems such as SDDL to express design concepts and other software project information. SDDL, which was designed for maximum flexibility, may be used as a vehicle for creating and testing methods and techniques to be considered for adoption as software design standards. Once standards have been agreed upon they could be enforced by generating a special version of the SDDL processor which accommodates the standards without the flexibility of the general version.

DESIGN STANDARDS FOR SIMULATION MODELING

To illustrate some candidate standards for expressing simulation and modeling design concepts, the design of a barber shop simulation is presented below. All information regarding the simulation model itself is contained entirely within the example, but some preliminary remarks will help the reader to approach the examination of the illustration from an appropriate perspective:

1. The SDDL processor reads the designer's input expressed in SDDL syntax, augments the input data with structure indentation, cross reference tables, etc., and produces the SDD.

2. The SDD is the medium for communicating the high-level design and project management information among the members of the software development team. It serves as the current, definitive statement of the status of the project and the design.

3. The SDD shown below presents a simulation program under development. It should be evaluated with respect to how well it performs its function of communicating information regarding the status and content of the model and of the project. The SDD must communicate to programmers who will implement the design and to other team members who must evaluate the progress and direction of the design effort.

4. The organization, structures, and statements used in the SDD to convey the information should be evaluated with respect to their usefulness as standards for specifying simulation programs.

5. Following the evaluation of the SDD, the SDDL system should be evaluated with respect to how well it supports and facilitates the production of the SDD. It is difficult to judge the efficacy of the SDDL system without actually using it, but some evaluation can be made by comparing the SDD to the input data from which it was generated. Documentation of the SDDL system can be obtained from the Jet Propulsion Laboratory.

Items to note while making this comparison are:

a. Line numbers on the SDD correspond exactly to input line numbers of the source data.

b. Indentation in the style of Structured Programming is provided by the processor by recognition of certain keywords.

c. Keywords can be established freely by the user.

d. Flow lines and page reference numbers augment the information content.

e. Parts of lines may be automatically right justified for emphasis.

f. Document format (page width, indentation amount, etc.) can be specified by the user.

g. Cross reference tables and the table of contents are supplied automatically.

e. Logic errors are detected and reported by the processor.

```
*****************************************
*                                       *
*                                       *
*                THE                    *
*                                       *
*  SOFTWARE DESIGN AND DEVELOPMENT LANGUAGE
*                                       *
*             SAMPLE DESIGN FOR         *
*                                       *
*                                       *
*          YE OLDE BARBER SHOPPE        *
*                                       *
*                                       *
*****************************************
```

TABLE OF CONTENTS PAGE 1

LINE PAGE 1

```
  16 PROGRAM MEMORANDUM
     ***************************************************************
  17 *                                                             *
  18 *                                                             *
  19 *    NOTE: BECAUSE OF SPACE LIMITATIONS SOME DESIGN MODULES AND DESIGN DETAIL *
  20 * HAD TO BE EXCLUDED.  TO ACCOUNT FOR THIS INCOMPLETENESS THE SDD SHOULD BE *
  21 * UNDERSTOOD TO REPRESENT AN IN-PROGRESS, PARTIALLY COMPLETE DESIGN.  *
  22 *                                                             *
  23 *    SOME OF THE PROJECT CONTROL MODULES WHICH HAD TO BE OMITTED ARE:  *
  24 *       CALENDAR OF TEAM MEETINGS AND AGENDA                  *
  25 *       ACKNOWLEDGEMENTS OF TEAM MEMBERS AND OTHER CONTRIBUTORS *
  26 *       DOCUMENT READING CONVENTIONS                          *
  27 *       LIST OF HIGH PRIORITY PROBLEM AREAS                   *
  28 *       DATA STRUCTURE LIST AND EXPLANATIONS                  *
  29 *                                                             *
  30 *                                                             *
     ***************************************************************
  31 ENDPROGRAM
```

```
LINE                                                                PAGE    2
  32 PROGRAM OBJECTIVES
  33       SIMULATE ONE DAY'S OPERATION OF A BARBER SHOP WITH THE
  34    FOLLOWING OPERATING CHARACTERISTICS:
  35
  36       THE SHOP:
  37         1. OPENING AND CLOSING TIMES ARE INPUT PARAMETERS.
  38         2. CUSTOMERS IN THE SHOP BEFORE CLOSING TIME MUST BE SERVICED.
  39
  40       THE CUSTOMERS:
  41         1. CUSTOMER ARRIVAL TIMES ARE EXPONENTIALLY DISTRIBUTED.
  42         2. MEAN TIME BETWEEN ARRIVALS IS AN INPUT PARAMETER WHICH VARIES
  43            THROUGHOUT THE DAY.
  44         3. CUSTOMERS HAVE BARBER PREFERENCES.
  45         4. EACH CUSTOMER HAS A WAITING TIME PATIENCE FACTOR.
  46
  47       THE BARBERS:
  48         1. THE NUMBER OF BARBERS IS AN INPUT PARAMETER.
  49         2. BARBERS SHOULD HAVE A LUNCH BREAK AFTER 3 AND BEFORE 5
  50            HOURS OF WORK.
  51         3. BARBERS ARE PAID 1.5 TIMES NORMAL RATE FOR TIME WORKED IN
  52            EXCESS OF 5 HOURS WITHOUT A BREAK AND 8 HOURS IN ONE DAY
  53         4. EACH BARBER HAS A SERVICE RATE.
  54         5. EACH BARBER HAS A CUSTOMER POPULARITY FACTOR.
  55
  56       SIMULATION OUTPUT:
  57         1. CUSTOMER WAITING TIME
  58         2. QUEUE LENGTHS
  59         3. BARBER UTILIZATION
  60 ENDPROGRAM OBJECTIVES
```

```
LINE                                                                PAGE    3
  62 PROGRAM MAIN ROUTINE
  63    CALL INITIALIZATION TO READ DATA AND SET UP THE MODEL---------->(   )
  64
  65    SCHEDULE CUSTOMER_ARRIVAL ON THE EVENT.SCHEDULE---------------->(  5)
  66    * GIVING EVENT TIME = IMMEDIATE
  67    SCHEDULE ARRIVAL_RATE_CHANGE ON THE EVENT.SCHEDULE------------->(  7)
  68    * GIVING EVENT TIME = IMMEDIATE
  69    SCHEDULE END_OF_SIMULATION ON THE EVENT.SCHEDULE--------------->(   )
  70    * GIVING EVENT TIME = CLOSING.TIME
  71
  72    CALL EVENT_SELECTOR TO BEGIN THE SIMULATION-------------------->(  4)
  73
  74    CALL SUMMARY_AND_REPORT ROUTINE-------------------------------->(   )
  75
  76 ENDPROGRAM MAIN ROUTINE
```

```
LINE                                                                    PAGE    4
  77 PROCEDURE EVENT_SELECTOR
     ***********************************************************
  78 *                                                         *
  79 * IN SOME PROGRAMMING LANGUAGES (E.G. SIMSCRIPT) THIS    *
  80 * FUNCTION IS SUPPLIED AUTOMATICALLY.                     *
  81 *                                                         *
     ***********************************************************
  82
  83
  84    SEARCH THE EVENT.SCHEDULE FOR THE NEXT MOST IMMINENT EVENT
  85    SELECT THE APPROPRIATE EVENT ROUTINE
  86    CASE 1
  87       CALL CUSTOMER_ARRIVAL EVENT------------------------------------>(  5)
  88    CASE 2
  89       CALL HAIRCUT_COMPLETION EVENT---------------------------------->(  6)
  90    CASE 3
  91       CALL END_OF_BREAK EVENT---------------------------------------->(   )
  92    CASE 4
  93       CALL ARRIVAL_RATE_CHANGE EVENT--------------------------------->(  7)
  94    ENDSELECT
  95 ENDPROCEDURE
```

```
LINE                                                                    PAGE    5
  96 EVENT FOR CUSTOMER_ARRIVAL ACTIONS
  97    IF IT IS PAST CLOSING.TIME
  98 <-----EXITEVENT WITHOUT FURTHER ACTION
  99    ELSE
 100       SCHEDULE CUSTOMER_ARRIVAL EVENT ON THE EVENT.SCHEDULE---------->(  5)
 101       * GIVING DELAY TIME = EXPONENTIAL DRAW (MEAN.TIME.BETWEEN.CUSTOMERS)
 102       MAKE A RANDOM SELECTION OF A BARBER CHOICE
 103       * USING BARBER.PREFERENCE FACTORRS
 104
 105       SELECT ACTION BASED ON BARBER ACTIVITY.STATUS
 106       CASE:  BARBER IS ON A BREAK
 107          REMOVE THE CUSTOMER FROM THE SHOP
 108          TALLY  LAST CUSTOMER STATISTICS
 109       CASE:  BARBER IS BUSY
 110          PLACE THE CUSTOMER IN THE WAITING.QUEUE
 111          RECORD THE CUSTOMER'S QUEUE.ENTRY.TIME
 112       CASE:  BARBER IS IDLE
 113          SCHEDULE HAIRCUT_COMPLETION ON THE EVENT.SCHEDULE-------------->(  6)
 114          * GIVING SERVICE.RATE OF THE BARBER
 115       OTHER CASES:  PROGRAM FAULT
 116          PROVIDE PROGRAM DIAGNOSTIC
           ENDSELECT - STMT SUPPLIED BY PROCESSOR
 117    ENDIF
 118 ENDEVENT FOR CUSTOMER_ARRIVAL ACTIONS
```

```
LINE                                                                    PAGE     6
 119 EVENT FOR HAIRCUT_COMPLETION ACTIONS
 120    * GIVEN BARBER AND CUSTOMER
 121
 122    ACCUMULATE AVERAGE.SERVICE.TIME AND CUSTOMERS.SERVICED
 123
 124    IF THE BARBER HAS WORKED MORE THAN 5 CONSECUTIVE HOURS
 125       SEARCH THE WAITING.QUEUE FOR A CUSTOMER WAITING FOR THIS BARBER
 126       IF A CUSTOMER IS WAITING
 127          REMOVE THE CUSTOMER FROM THE WAITING.QUEUE
 128          ACCUMULATE AVERAGE.WAITING.TIME
 129          SCHEDULE HAIRCUT_COMPLETION EVENT ON THE EVENT.SCHEDULE----->(  6)
 130          * GIVEN BARBER, CUSTOMER, AND SERVICE TIME
 131       ELSE
 132          IF THE BARBER HAS WORKED MORE THAN 3 CONSECUTIVE HOURS
 133             SET THE BARBER'S ACTIVITY.STATUS TO "ON BREAK"
 134             SCHEDULE END_OF_BREAK ON THE EVENT.SCHEDULE---------->(   )
 135             * GIVEN BARBER, BREAK TIME = 30 MINUTES
 136          ELSE
 137             SET BARBER'S ACTIVITY STATUS TO "IDLE"
 138          ENDIF
 139       ENDIF
 140    ENDIF
 141 ENDEVENT


LINE                                                                    PAGE     7
 142 EVENT FOR ARRIVAL_RATE_CHANGE ACTIONS
 143    READ IN A NEW VALUE FOR MEAN.TIME.BETWEEN.CUSTOMERS
 144    READ IN THE DURATION TIME FOR THIS ARRIVAL RATE
 145    IF THE NEXT RATE CHANGE IS DUE BEFORE CLOSING.TIME
 146       SCHEDULE ARRIVAL_RATE_CHANGE ON THE EVENT.SCHEDULE---------->(  7)
 147       * GIVING EVENT TIME = DURATION
 148    ENDIF
 149 ENDEVENT


   ********** MODULE REFERENCE TREE ******                              PAGE     8
  LN  PAGE
   1     1  MEMORANDUM

   2     2  OBJECTIVES

   3     3  MAIN
   4     *  .  INITIALIZATION
   5     5  .  CUSTOMER_ARRIVAL
   6     5  .  .  CUSTOMER_ARRIVAL
   7              ** EXPANDED ON LINE    5
   8     6  .  .  HAIRCUT_COMPLETION
   9     6  .  .  .  HAIRCUT_COMPLETION
  10                 ** EXPANDED ON LINE    8
  11     *  .  .  .  END_OF_BREAK
  12     7  .  ARRIVAL_RATE_CHANGE
  13     7  .  .  ARRIVAL_RATE_CHANGE
  14              ** EXPANDED ON LINE   12
  15     *  .  END_OF_SIMULATION
  16     4  .  EVENT_SELECTOR
  17     5  .  .  CUSTOMER_ARRIVAL
  18              ** EXPANDED ON LINE    5
  19     6  .  .  HAIRCUT_COMPLETION
  20              ** EXPANDED ON LINE    8
  21     *  .  .  END_OF_BREAK
  22     7  .  .  ARRIVAL_RATE_CHANGE
  23              ** EXPANDED ON LINE   12
  24     *  .  SUMMARY_AND_REPORT
```

IDENTIFIER

```
ARRIVAL_RATE_CHANGE
    PAGE   3  PROGRAM MAIN
     LINES  67
    PAGE   4  PROCEDURE EVENT_SELECTOR
     LINES  93
    PAGE   7  EVENT FOR ARRIVAL_RATE_CHANGE
     LINES  142, 146
CUSTOMER_ARRIVAL
    PAGE   3  PROGRAM MAIN
     LINES  65
    PAGE   4  PROCEDURE EVENT_SELECTOR
     LINES  87
    PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
     LINES  96, 100, 118
END_OF_BREAK
    PAGE   4  PROCEDURE EVENT_SELECTOR
     LINES  91
    PAGE   6  EVENT FOR HAIRCUT_COMPLETION
     LINES  134
END_OF_SIMULATION
    PAGE   3  PROGRAM MAIN
     LINES  69
EVENT_SELECTOR
    PAGE   3  PROGRAM MAIN
     LINES  72
    PAGE   4  PROCEDURE EVENT_SELECTOR
     LINES  77
HAIRCUT_COMPLETION
    PAGE   4  PROCEDURE EVENT_SELECTOR
     LINES  89
    PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
     LINES  113
    PAGE   6  EVENT FOR HAIRCUT_COMPLETION
     LINES  119, 129
INITIALIZATION
    PAGE   3  PROGRAM MAIN
     LINES  63
MAIN
    PAGE   3  PROGRAM MAIN
     LINES  62, 76
MEMORANDUM
    PAGE   1  PROGRAM MEMORANDUM
     LINES  16
OBJECTIVES
    PAGE   2  PROGRAM OBJECTIVES
     LINES  32, 60
SUMMARY_AND_REPORT
    PAGE   3  PROGRAM MAIN
     LINES  74
```

```
                                MODEL PARAMETERS
                            CROSS REFERENCE LISTING                    PAGE   10
IDENTIFIER++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

ACTIVITY.STATUS
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  105
     PAGE   6  EVENT FOR HAIRCUT_COMPLETION
      LINES  133
AVERAGE.SERVICE.TIME
     PAGE   6  EVENT FOR HAIRCUT_COMPLETION
      LINES  122
AVERAGE.WAITING.TIME
     PAGE   6  EVENT FOR HAIRCUT_COMPLETION
      LINES  128
BARBER.PREFERENCE
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  103
CLOSING.TIME
     PAGE   3  PROGRAM MAIN
      LINES  70
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  97
     PAGE   7  EVENT FOR ARRIVAL_RATE_CHANGE
      LINES  145
CUSTOMERS.SERVICED
     PAGE   6  EVENT FOR HAIRCUT_COMPLETION
      LINES  122
EVENT.SCHEDULE
     PAGE   3  PROGRAM MAIN
      LINES  65, 67, 69
     PAGE   4  PROCEDURE EVENT_SELECTOR
      LINES  84
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  100, 113
     PAGE   6  EVENT FOR HAIRCUT_COMPLETION
      LINES  129, 134
     PAGE   7  EVENT FOR ARRIVAL_RATE_CHANGE
      LINES  146
MEAN.TIME.BETWEEN.CUSTOMERS
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  101
     PAGE   7  EVENT FOR ARRIVAL_RATE_CHANGE
      LINES  143
QUEUE.ENTRY.TIME
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  111
SERVICE.RATE
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  114
WAITING.QUEUE
     PAGE   5  EVENT FOR CUSTOMER_ARRIVAL
      LINES  110
     PAGE   6  EVENT FOR HAIRCUT_COMPLETION
      LINES  125, 127
```

SOURCE INPUT FOR THE BARBER SHOP SIMULATION SDD

```
 1:#DEFINE MODULE EVENT ENDEVENT EXITEVENT
 2:#DEFINE BLOCK SELECT, , ,OTHER
 3:#DEFINE CALL SCHEDULE
 4:#TITLE SODL EXAMPLE
 5:
 6:THE
 7:
 8:SOFTWARE DESIGN AND DEVELOPMENT LANGUAGE
 9:
10:SAMPLE DESIGN FOR
11:
12:
13:YE OLDE BARBER SHOPPE
14:
15:#END
16:PROGRAM MEMORANDUM
17:#TEXT
18:
19:    NOTE: BECAUSE OF SPACE LIMITATIONS SOME DESIGN MODULES AND DESIGN DETAIL
20:HAD TO BE EXCLUDED.  TO ACCOUNT FOR THIS INCOMPLETENESS THE SDD SHOULD BE
21:UNDERSTOOD TO REPRESENT AN IN-PROGRESS, PARTIALLY COMPLETE DESIGN.
22:
23:    SOME OF THE PROJECT CONTROL MODULES WHICH HAD TO BE OMITTED ARE:
24:       CALENDAR OF TEAM MEETINGS AND AGENDA
25:       ACKNOWLEDGEMENTS OF TEAM MEMBERS AND OTHER CONTRIBUTORS
26:       DOCUMENT READING CONVENTIONS
27:       LIST OF HIGH PRIORITY PROBLEM AREAS
28:       DATA STRUCTURE LIST AND EXPLANATIONS
29:
30:#END
31:ENDPROGRAM
32:PROGRAM OBJECTIVES
33:    SIMULATE ONE DAY'S OPERATION OF A BARBER SHOP WITH THE
34:FOLLOWING OPERATING CHARACTERISTICS:
35:
36:    THE SHOP:
37:      1. OPENING AND CLOSING TIMES ARE INPUT PARAMETERS.
38:      2. CUSTOMERS IN THE SHOP BEFORE CLOSING TIME MUST BE SERVICED.
39:
40:    THE CUSTOMERS:
41:      1. CUSTOMER ARRIVAL TIMES ARE EXPONENTIALLY DISTRIBUTED.
42:      2. MEAN TIME BETWEEN ARRIVALS IS AN INPUT PARAMETER WHICH VARIES
43:         THROUGHOUT THE DAY.
44:      3. CUSTOMERS HAVE BARBER PREFERENCES.
45:      4. EACH CUSTOMER HAS A WAITING TIME PATIENCE FACTOR.
46:
47:    THE BARBERS:
48:      1. THE NUMBER OF BARBERS IS AN INPUT PARAMETER.
49:      2. BARBERS SHOULD HAVE A LUNCH BREAK AFTER 3 AND BEFORE 5
50:         HOURS OF WORK.
51:      3. BARBERS ARE PAID 1.5 TIMES NORMAL RATE FOR TIME WORKED IN
52:         EXCESS OF 5 HOURS WITHOUT A BREAK AND 8 HOURS IN ONE DAY
53:      4. EACH BARBER HAS A SERVICE RATE.
54:      5. EACH BARBER HAS A CUSTOMER POPULARITY FACTOR.
55:
56:    SIMULATION OUTPUT:
57:      1. CUSTOMER WAITING TIME
```

```
 58:      2. QUEUE LENGTHS
 59:      3. BARBER UTILIZATION
 60:ENDPROGRAM OBJECTIVES
 61:*MARK MODEL PARAMETERS . MODULES *
 62:PROGRAM MAIN ROUTINE
 63:CALL INITIALIZATION TO READ DATA AND SET UP THE MODEL
 64:
 65:SCHEDULE CUSTOMER_ARRIVAL ON THE EVENT.SCHEDULE
 66:* GIVING EVENT TIME = IMMEDIATE
 67:SCHEDULE ARRIVAL_RATE_CHANGE ON THE EVENT.SCHEDULE
 68:* GIVING EVENT TIME = IMMEDIATE
 69:SCHEDULE END_OF_SIMULATION ON THE EVENT.SCHEDULE
 70:* GIVING EVENT TIME = CLOSING.TIME
 71:
 72:CALL EVENT_SELECTOR TO BEGIN THE SIMULATION
 73:
 74:CALL SUMMARY_AND_REPORT ROUTINE
 75:
 76:ENDPROGRAM MAIN ROUTINE
 77:PROCEDURE EVENT_SELECTOR
 78:*TEXT
 79:IN SOME PROGRAMMING LANGUAGES (E.G. SIMSCRIPT) THIS
 80:FUNCTION IS SUPPLIED AUTOMATICALLY.
 81:*END
 82:
 83:
 84:SEARCH THE EVENT.SCHEDULE FOR THE NEXT MOST IMMINENT EVENT
 85:SELECT THE APPROPRIATE EVENT ROUTINE
 86:CASE 1
 87:CALL CUSTOMER_ARRIVAL EVENT
 88:CASE 2
 89:CALL HAIRCUT_COMPLETION EVENT
 90:CASE 3
 91:CALL END_OF_BREAK EVENT
 92:CASE 4
 93:CALL ARRIVAL_RATE_CHANGE EVENT
 94:ENDSELECT
 95:ENDPROCEDURE
 96:EVENT FOR CUSTOMER_ARRIVAL ACTIONS
 97:IF IT IS PAST CLOSING.TIME
 98:EXITEVENT WITHOUT FURTHER ACTION
 99:ELSE
100:SCHEDULE CUSTOMER_ARRIVAL EVENT ON THE EVENT.SCHEDULE
101:* GIVING DELAY TIME = EXPONENTIAL DRAW (MEAN.TIME.BETWEEN.CUSTOMERS)
102:MAKE A RANDOM SELECTION OF A BARBER CHOICE
103:* USING BARBER.PREFERENCE FACTORS
104:
105:SELECT ACTION BASED ON BARBER ACTIVITY.STATUS
106:CASE:  BARBER IS ON A BREAK
107:REMOVE THE CUSTOMER FROM THE SHOP
108:TALLY  LAST CUSTOMER STATISTICS
109:CASE:  BARBER IS BUSY
110:PLACE THE CUSTOMER IN THE WAITING.QUEUE
111:RECORD THE CUSTOMER'S QUEUE.ENTRY.TIME
112:CASE:  BARBER IS IDLE
113:SCHEDULE HAIRCUT_COMPLETION ON THE EVENT.SCHEDULE
114:* GIVING SERVICE.RATE OF THE BARBER
```

```
115:OTHER CASES:  PROGRAM FAULT
116:PROVIDE PROGRAM DIAGNOSTIC
117:ENDIF
118:ENDEVENT FOR CUSTOMER_ARRIVAL ACTIONS
119:EVENT FOR HAIRCUT_COMPLETION ACTIONS
120:* GIVEN BARBER AND CUSTOMER
121:
122:ACCUMULATE AVERAGE.SERVICE.TIME AND CUSTOMERS.SERVICED
123:
124:IF THE BARBER HAS WORKED MORE THAN 5 CONSECUTIVE HOURS
125:SEARCH THE WAITING.QUEUE FOR A CUSTOMER WAITING FOR THIS BARBER
126:IF A CUSTOMER IS WAITING
127:REMOVE THE CUSTOMER FROM THE WAITING.QUEUE
128:ACCUMULATE AVERAGE.WAITING.TIME
129:SCHEDULE HAIRCUT_COMPLETION EVENT ON THE EVENT.SCHEDULE
130:* GIVEN BARBER, CUSTOMER, AND SERVICE TIME
131:ELSE
132:IF THE BARBER HAS WORKED MORE THAN 3 CONSECUTIVE HOURS
133:SET THE BARBER'S ACTIVITY.STATUS TO "ON BREAK"
134:SCHEDULE END_OF_BREAK ON THE EVENT.SCHEDULE
135:* GIVEN BARBER, BREAK TIME = 30 MINUTES
136:ELSE
137:SET BARBER'S ACTIVITY STATUS TO "IDLE"
138:ENDIF
139:ENDIF
140:ENDIF
141:ENDEVENT
142:EVENT FOR ARRIVAL_RATE_CHANGE ACTIONS
143:READ IN A NEW VALUE FOR MEAN.TIME.BETWEEN.CUSTOMERS
144:READ IN THE DURATION TIME FOR THIS ARRIVAL RATE
145:IF THE NEXT RATE CHANGE IS DUE BEFORE CLOSING.TIME
146:SCHEDULE ARRIVAL_RATE_CHANGE ON THE EVENT.SCHEDULE
147:* GIVING EVENT TIME = DURATION
148:ENDIF
149:ENDEVENT
```

BIBLIOGRAPHY

1. Dijkstra, E. W., "Notes on Structured Programming," in Structured Programming, Academic Press, New York, 1972.
2. Kleine, H., Software Design and Documentation Language, Publication 77-24, Jet Propulsion Laboratory, Pasadena, Calif., July 1, 1977.
3. Caine, S. H., and Gordon, E. K., "PDL-A Tool for Software," Program Design Language Reference Guide, Caine, Farber, and Gordon, Inc., Pasadena, Calif., Sept. 18, 1974.
4. Heimburger, D. A., and Metcalfe, M. A., "VEEP: Vehicle Economy, Emissions and Performance Program" (this Proceedings).