1981 Winter Simulation Conference Proceedings
T.I. Oren, C.M. Delfosse, C.M. Shub (Eds.)

SIMSCRIPT II.5 TUTORIAL

Edward C. Russell
CACI
12011 San Vicente Boulevard
Los Angeles, CA 90049

ABSTRACT

This tutorial will present the highlights of the SIMSCRIPT II.5 approach to building discrete event simulation models. The approach will be to construct a small example problem, implement the program in SIMSCRIPT II.5, and then to display the modularity which is possible with SIMSCRIPT by adding several "real-world" complexities to the model.

1. BACKGROUND

Simulation languages are (or should be) more than extensions of general-purpose programming languages designed to ease the burden of programming simulation problems. The influence of a good simulation language should be felt during the specification and model design stages of simulation as well as during computer implementation. If the "world-view" of the simulation language is well understood, and if that world-view is appropriate for a given problem, then the language should aid immeasurably in reducing the effort (and consequently elapsed time) in transforming model from concept to realization.

2. SIMSCRIPT II.5 WORLD-VIEW

SIMSCRIPT is a discrete-event language. Actions are modelled in terms of events. Sequences of events describing actions of a single object (or entity) are modelled as processes. Many important relationships are described staticly in terms of entities-attributes-sets. This very powerful data structuring is one of the unique features of the language. The implementation in SIMSCRIPT of the classical simulation problems, such as small queueing models or job-shop simulations, have been described elsewhere (see Literature).

3. SIMSCRIPT II.5 IMPLEMENTATIONS

SIMSCRIPT II.5 has been implemented on seven different manufacturer's computers including IBM, Honeywell, CDC, Univac, Digital Equipment, NCR, and PR1ME. A major emphasis is to achieve portability of models from one machine to another since the type of models typically written in SIMSCRIPT have a fairly long lifetime and are often shared among a community of users with different computers.

4. SIMSCRIPT II.5 AVAILABILITY

SIMSCRIPT II.5 is the proprietary product of CACI and is available from them for lease or purchase.

SIMSCRIPT is also available on Boeing Computer Services, Control Data-Cybernet, Datacrown, Litton Data Systems, McDonnell-Douglas' Mac Auto, National CSS, Service Bureau Corporation Tymshare, and United Computing Services.

A university program is supported by CACI in which SIMSCRIPT is supplied to educational institutions for the cost of distribution.

5. THE TUTORIAL EXAMPLE

The tutorial is built around a simple queueing problem first presented in (SCHRIBER) and later revised in (FISHMAN 1978). An African Port consists of three docks serviced by a single tug. Ships which arrive to be loaded must use the tug to enter and leave the docks. The ships arrive according to an exponential distribution of inter-arrival times with a mean of eleven hours. The ships are of three sizes distributed as follows:

55% have a mean loading time of 18 hours
25% have a mean loading time of 24 hours
20% have a mean loading time of 36 hours

The times are all exponentially distributed about these means. The time for moving a ship into or out of a dock is also exponentially distributed with a mean of one hour. Ships queue for the tug and/or the dock on a first-come-first-served basis.

The first task is to model this existing situation and measure the utilization factors for the tug and the dock. In addition, the congestion in the harbor (queueing statistics for the tug and dock) are to be measured. Finally, statistics on the in-port time for the ships should be collected.

After this simple model is developed, the next phase is to superimpose a new category of tanker. These new tankers belong to a fixed size fleet. They have a mean loading time of 21 hours and require the same tug and dock services as the other tankers. When they leave the port, they make a round-trip to their destination in a mean time of ten days (again exponentially distributed). The same statistics should be reported as above in order to determine the impact of the new tankers on the port.

The next complication to be added to the model is the effect of external interruptions such as storms and smog. A storm serves to delay the arrival of ships and to detain them when ready to depart. Smog alerts cause the unloading of ships to be interrupted until the smog alert is lifted. Since storms and smog are fairly rare occurrences, these phenomena are to be represented as they actually occurred over some historical period.

Finally, in order to verify the correct execution of the model, the SIMSCRIPT tools for debugging will be added to the model.

Fig. 1 contains the simple SIMSCRIPT program for this problem. The tutorial shall be comprised of a detailed walkthrough of the original model and a discussion of the evolution to the completed model in the following steps:

1. original simple model with only old tankers;
2. additions for statistical output;
3. additions for the new tankers;
4. additions for storms and smog;
5. additions for debug.

REFERENCES

Fishman, G.S., Principles of Discrete Event Simulation, Wiley-Interscience, New York, 1978.

Kiviat, P.J., H.M. Markowitz, and R. Villanueva, SIMSCRIPT II.5 Programming Language (edited by E.C. Russell), CACI, Los Angeles, 1973.

Russell, E.C. and J.S. Annino, A Quick Look at SIMSCRIPT II.5, CACI, Los Angeles, 1979.

Russell, E.C., Building Simulation Models with SIMSCRIPT II.5, CACI, Los Angeles, 1981.

Schriber, T.J., Simulation Using GPSS, Wiley, New York, 1974.

```
"                  SIMSCRIPT II.5 TUTORIAL MODEL
"                      AN AFRICAN PORT STUDY

PREAMBLE
    PROCESSES INCLUDE GENERATOR AND SHIP
    RESOURCES INCLUDE DOCK AND TUG
    PERMANENT ENTITIES
        EVERY TANKER.TYPE HAS A LOADING.TIME
        DEFINE LOADING.TIME AS A REAL VARIABLE
    THE SYSTEM HAS A TYPE.DISTRIBUTION RANDOM STEP VARIABLE
    DEFINE TYPE.DISTRIBUTION AS AN INTEGER VARIABLE
    DEFINE .END.OF.SIMULATION TO MEAN TIME.V >= 365.
END "PREAMBLE

MAIN
    READ TYPE.DISTRIBUTION
    CREATE EACH TANKER.TYPE(4)
    LET LOADING.TIME(1) = 18
    LET LOADING.TIME(2) = 24
    LET LOADING.TIME(3) = 36
    CREATE EVERY DOCK(1)
    LET U.DOCK(1) = 3
    CREATE EVERY TUG(1)
    LET U.TUG(1) = 1
    ACTIVATE A GENERATOR NOW
    START SIMULATION
END "MAIN

PROCESS GENERATOR
    UNTIL .END.OF.SIMULATION,
    DO
        ACTIVATE A SHIP NOW
        WAIT EXPONENTIAL.F(11.0, 2) HOURS
    LOOP
    STOP
END "PROCESS GENERATOR

PROCESS SHIP
    DEFINE TANKER.TYPE AS AN INTEGER VARIABLE
        LET TANKER.TYPE = TYPE.DISTRIBUTION
        REQUEST 1 DOCK(1)
        REQUEST 1 TUG(1)
        WAIT EXPONENTIAL.F(1.0,3) HOURS
        RELINQUISH 1 TUG(1)
        WORK EXPONENTIAL.F(LOADING.TIME(TANKER.TYPE),4) HOURS
        REQUEST 1 TUG(1)
        WAIT EXPONENTIAL.F(1.0,5) HOURS
        RELINQUISH 1 TUG(1)
        RELINQUISH 1 DOCK(1)
END "PROCESS SHIP
```

Figure 1. AFRICAN PORT MODEL