AN INTRODUCTION TO THE SIMULATION DATA LANGUAGE

Charles R. Standridge, Ph.D.
Assistant Professor
Systems Engineering
The University of Iowa
Iowa City, Iowa 52242

A. Alan B. Pritsker, Ph.D., P.E.
President
Pritsker & Associates, Inc.
P.O. Box 2413
West Lafayette, Indiana 47906

ABSTRACT

In this tutorial, basic concepts for using database management capabilities in simulation studies will be discussed. The implementation of these concepts in the Simulation Data Language (SDL™) will be presented. For better understanding, the tutorial will consist of two example simulation studies, one elementary and one realistic, performed using SDL. In the course of these examples, SDL data organization constructs, expecially those for model outputs, and SDL commands will be discussed. Finally, a simulation study application performed using SDL will be overviewed.

1. INTRODUCTION

As simulation has become widely accepted as an analysis technique, the complexity of the studies performed using simulation has increased. Increased complexity has made other activities in addition to model building and implementation more significant. One such activity is the management of the data involved in a large scale simulation study. Analysts often spend a great deal of time organizing input data, determining how to analyze model outputs, and studying printed results of runs. However, there has been no simulation-specific software support for data management. Thus, an analyst must develop data management software for specific projects or expand and rerun simulation programs when different output reports become necessary, for example.

In general data processing, analysts use software tools (called database management systems) to assist with the management and organization of large amounts of data. Many such systems are widely available [Date, Martin, Whinston and Haseman]. Typically, these systems do not provide all the specific capabilities needed in simulation. A database management system for simulation must help the user organize, store and retrieve, and report input parameter values, model outputs, and analyses of model outputs. Furthermore, the system must be usable from within programs written in a simulation language. Of course, the system must still provide traditional database functions such as loading, editing, deleting and locating data that satisfies particular relationships.

We have developed SDL™, the Simulation Data Language, to meet the specific data management requirements of simulation analysts. In particular, SDL provides a simulation-specific framework for organizing data and a set of commands for manipulating data in ways typically needed in simulation.

2. SDL DATA ORGANIZATION

SDL provides both a general data organization and a data organization for model outputs and analyses of model outputs. The general data organization is relational [Codd]. A relation is a matrix of rows and columns. A database consists of one or more relations. The data organization for model outputs and analyses of model outputs is an extension of the relational data organization. Special structures for organizing, in the same database, outputs from different runs of the same model or from runs of different models are provided. Furthermore, special structures for organizing all observations of variables which are generated by a model, statistics and histograms are supplied.

3. SDL COMMANDS

SDL consists of over 120 commands which may be organized into 16 categories:

SDL is a trademark of Pritsker & Associates, Inc.

81CH1709-5/81/0000-0635$00.75

1. Relation definition: Commands by which the user describes the characteristics of the relations comprising the database (number of columns, relation name, column names, columns containing key values);
2. Loading: Commands which transfer data from non-SDL files to an SDL database;
3. Editing: Commands which examine data for conformity to user specified conditions;
4. File creation: Commands which transfer data from an SDL database to a non-SDL file;
5. Deletion: Commands which remove data or relation definitions from an SDL database;
6. Statistics: Commands which compute and store in the database commonly used statistical measures;
7. Histograms: Commands which build histograms and store them in the database;
8. Report generation: Commands which generate columnar reports or display histograms in either an SDL defined or a user-specified format;
9. Plot generation: Commands which generate plots in either a user-specified or an SDL defined format;
10. Calculator: Commands which perform simple, arithmetic calculations on data in the database and store the results in the database;
11. Error recovery: Commands which allow the user to restore a correct database after an error occurs during a run of an SDL program;
12. Housekeeping: Miscellaneous commands such as those which open and close a database;
13. Storage: Commands which store one row of data in the database; and
14. Retrieval: Commands which retrieve one row of data from the database;
15. Aggregation: Commands which form arithmetic summaries of data;
16. Simulation statistics: Commands which compute the variance of sample means concerning model outputs.

Some commands in each category may be used to perform general database management functions. More importantly, other commands perform functions that are uniquely useful in simulation. These include:

1. definition of relations which contain model outputs, histograms or statistics;
2. computing statistics or building histograms by replication, by batches within one replication, or by regeneration cycle, deleting observations from an initial transcient period if desired;
3. generating reports showing statistics concerning the same variable from models of different system alternatives or from the actual system;
4. generating plots displaying time series of the same variable from models of different system alternatives or from the actual system;
5. computing interarrival distributions observations and observations of time delays between two events in a model;
6. storing individual observations or statistical summaries of observations of model variables;
7. aggregating statistics concerning a variable across model replications or batches within one replication;
8. computing confidence intervals concerning sample means of model variables using the method of replications, batch means or regeneration; and
9. taking a random sample from a histogram stored in the database.

In addition, a user can easily write programs to provide application-specific manipulations of model outputs, including links to graphical display software.

These commands, along with the data organization for model outputs and analyses of model outputs, provide simulation-specific capabilities which make SDL unique among database management systems.

In addition to these simulation-specific capabilities, a user may make use of the general database management capabilities of SDL to enhance the modeling and analysis process. For example, a time history of actual arrivals to a system could be loaded into the database and edited. This historical arrival pattern could be employed within the model by using SDL retrieval commands to obtain the next arrival from the database whenever it was required.

4. ACCESS TO AN SDL DATABASE

Access to an SDL database is gained by invoking SDL commands. SDL is written entirely in ANSI standard FORTRAN. Thus, an SDL command may be used by invoking the appropriate FORTRAN subprogram. Typically, this method of access is used to store and retrieve data, within a simulation program, for example. Alternatively, the SDL Operation Invocation Language (OIL) may be used. The OIL is a high level programming language designed to allow a user-friendly interface for specifying values for the parameters of SDL commands. Typically, OIL programs are written to define the data organization, load data, edit data, delete data, create sequential files, compute statistics, build histograms, generate reports, and generate plots. In addition, application-specific programs for using an SDL database interactively have been written [Standridge, 1981b] and a general interactive interface for SDL is being developed [Chang].

5. SDL APPLICATIONS

To date, eight SDL applications have been performed. In these applications, SDL has been used:

1. in the analysis of model outputs;
2. to allow data gathered from the system being studied to be employed in the simulation model of that system;
3. to decompose large-scale, complex models into sets of smaller, simpler models; and
4. in the development of automatic, user-transparent data collection in special purpose simulation languages.

The applications include the assessment of operational policies and rates used by a capitation reimbursement system; the assessment of a hospital clinic and related laboratory operations; the projection of the availability of primary care physicians in Indiana; the assessment of the ability of a maintenance facility to meet scheduled maintenance requirements for a fleet of aircraft; the assessment of the ability of a job shop to meet its current production requirements; the development of a simulation language tailored for modeling steel operations; the development of a transition path analysis language for economic, technological and societal problem assessment; and the development of network simulation language for modeling river systems. In these application SDL was helpful when:

1. comparing data generated by models of different alternatives;
2. all analyses of the output data cannot be defined before runs are made;
3. using data as input to graphics displays;
4. models are large and dividing them into independent components is helpful;
5. data gathered from the system must be managed and analyzed; and/or
6. comparisons of data gathered from the system with model-generated data must be made.

6. SUMMARY

In this paper, SDL concepts for using database management capabilities in a simulation study have been overviewed. The SDL data organization, SDL commands, ways of accessing an SDL database, and applications of SDL have been surveyed.

Additional information concerning SDL can be obtained from Pritsker & Associates, Inc.; P.O. Box 2413; West Lafayette, Indiana 47906 or by consulting material listed in the bibliography. Standridge [1981b] discusses the eight SDL applications; Rao, Rundgren and Standridge, Standridge and Marshall, and Standridge and Wortman discuss individual applications. Standridge [1980] and Standridge [1981a] present basic SDL concepts in some detail.

BIBLIOGRAPHY

Chang, Pao-Fang (1981), "An Interactive Interface for the Simulation Data Language," unpublished masters thesis, The University of Iowa, Iowa City, Iowa.

Codd, E. F. (1970), "A Relational Model of Data for Large Shared Data Banks," Communications of the ACM, Volume 13, Number 6.

Date, C. J. (1975), An Introduction to Database Systems, Addison-Wesley, Reading, MA.

Martin, James (1977), Computer Data-Base Organization, Prentice-Hall, Englewood Cliffs, NJ.

Rao, Narasinga B. (1981), "A River System Modeling and Simulation Methodology," unpublished Ph.D. dissertation, The University of Iowa, Iowa City, Iowa.

Rundgren, William P. and Charles R. Standridge (1981), "A Database Supported Discrete Parts Manufacturing Simulation," Proceedings of the 1981 Winter Simulation Conference.

Standridge, Charles R. (1980), The Simulation Data Language (SDL) Language Reference Manual, Pritsker & Associates, Inc. West Lafayette, IN.

Standridge, Charles R. (1981a), "The Simulation Data Language: Fundamental Concepts for Its Use in Simulation Studies," Simulation, September 1981.

Standridge, Charles R. (1981b), "The Simulation Data Language: Applications and Examples of Its Use in Simulation Studies," Simulation, October 1981.

Standridge, Charles R. and Susan R. Marshall (1981), "Enhancing Simulation Analyses of Health Care Delivery Policies Using SDL Database Capabilities," Proceedings of the 1981 Winter Simulation Conference.

Standridge, Charles R. and David B. Wortman (1981), "SDL: The Simulation Data Language: A Database Management System for Modelers," Simulation Today in Simulation, August 1981.

Whinston, A. and W. Haseman (1977), Introduction to Data Management, Richard D. Irwin, Inc., Homestead, IL.