

Computer Simulation and Learning Theory, Volume 3, 1976

STRUCTURE ANALYSIS IN SIMULATION DEVELOPMENT

Hugh V. Leach

Washburn University of Topeka

Business simulation exercises are abstracted forms of real world business operations. Hoare points out that abstraction arises from the recognition of similarities between certain objects, situations, or processes in the real world, and upon concentrating our study upon these similarities. When we have recognized which similarities are relevant to the prediction and control of the future success or failure of the firm, we will tend to regard the similarities as fundamental to such study.

Primary uses of such abstracted representation are for research, teaching, and even information storage purposes. It is for use in research that abstraction will provide a more thorough understanding of the interactions involved. Teaching will convey information about the real world aspect to others. Abstracted information can be recorded for historical use by future generations.

The first stage in the abstraction process is for the person involved to gain a thorough understanding of the problem situation under study. This would include identification of the relevant variables and an understanding of the general relationships among these variables. Achievement of this understanding would be followed by the development of the rigorous logical statement--presuppositions about the real world, and a formulation of the desired objectives of the particular simulation project in mind.

The next stage would be the development of algorithm or an abstract program which is demonstrably capable of carrying out the stated tasks. Later stages would tend to be more in terms of computer programming activities beyond the scope of this paper.

Abstraction is an intellectual process that should move from the very abstract formulation of the problem to more and, more concrete aspects of a solution. Quite frequently, earlier steps in this progression of abstraction will prove to be in error and, therefore, deeper investigation of the nature of the problem and, perhaps a return to earlier stages in the process of progression will be required. Thus the abstraction process should lead to the development of some mathematical models to express the nature of relationships being studied. Or cut points out that mathematical models of economic systems consists of four elements; components, variables, parameters and functional relationships. Com-

Computer Simulation and Learning Theory, Volume 3, 1976

ponents are used. to specify segments of tile system. Variables relate one component to another. Parameters express the degree of relationship between variable and components. And functional relationships are an expression of the nature of the interrelationships.

We may summarize the development of a simulation project into the following steps: a) study the real world problem situation in detail in order to identify components and variables, to estimate parameters, and to understand the functional relationships; 2) specify the objectives of this particular simulations. This refers to such things as research objectives, if this is a research simulation, or the teaching objectives, if this is a teaching simulation; 3) build the algorithm to express the relationships among the variables identified and to express the functional form desired; 4.) getting the algorithm programmed for computer operation or set up for other operation if necessary, 5) testing the simulation exercise for completeness and. accuracy.

Nature of Objectives for Simulation Exercise

The objective should be specified for the particular research or teaching function desired. It may be desirable to study tactical decisions or to study strategic decisions. The simulation can be developed to study either of those. The simulation exercise may need to be revised for later applications with some modifications in objectives. Therefore the development of the simulation should reflect such possible revision. The simulation exercise must be correct in terms of function. Structure analysis of simulation exercises will help to achieve all three of these objectives.

Structure Analysis

The strength of structure analysis is based upon the following characteristics. The simulation problem is broken into modules. These modules have the characteristics of: 1) being sufficiently small and simple so that the researcher can understand the nature of the simulation relationships within that module; 2) the modules are linked together in sequence or within repetitive loops to achieve the basic objectives of the simulation exercise; 3) the linkage between the modules is one point of logic input from one point of logic output. The linkage being a common logic flow from one module to the next module.

Dijkstra offers several reasons for using the sequencing procedure of analysis. 1) We can understand a decomposed

Computer Simulation and Learning Theory, Volume 3, 1976

problem by enumerative reasoning provided the number of sub- actions within a segment is sufficiently small. 2) We can establish relationships between components and can attach meanings to variables. 3) We can understand repetitive subactions by mathematical induction. Li.) We can identify specific points of computational progress.

Warnier argues that "Any set of information must be subdivided into sub-sets from the highest level using appropriate criteria for subdivision. This rule applies to any problem involving the organization of data, program or results." He identifies two approaches that could be used on the construction of data sets and processing procedures. A weaker approach would be to tackle the problem at the beginning in time, develop the analysis in time sequence and write down the processing steps in space. The weakness evolved because the process leads the analyst to attach equal importance to details and logic. A more appropriate approach in the hierarchical which facilitates the focus on logic flow from the general to the particular case. Certain rules are specified by Warnier to control the development of a hierarchical organization. "Any set of information must be subdivided into sub-sets from the highest level using appropriate criteria for subdivision." Two criteria must be used in the subdivision process and must be applied in sequence. First apply the criterion: "Subdivide a set if it includes sub-sets that are present a number of times other than once only." After the information set has been subdivided according to the above criterion the second criterion is applied, "Write down, in order, the items which make up the set." The information generated by the application of these rules and criteria permit the fulfillment of yet another rule. "In a hierarchical structure there must be a mapping between each sub-set and all the sub-sets of higher levels," The mapping provides a framework for analysis of variables within components and a means of effective communication between user and the systems analyst.

As we focus on the problems of total simulation development we recognize two types, developmental and communication. User problems may reflect inadequate identification of the goals or the model may be inadequate to express the desired relationships among components and variables. Communication problems arise when the systems analyst received message A about the nature of the simulation when the user conveyed message B. All too often the two parties do not realize that there is a communication problem until a program has been written and the results of test runs differ from those expected by the user. Then

Computer Simulation and Learning Theory, Volume 3, 1976

significant amounts of time are spent to identify the reasons for the discrepancies and to make program alterations. Other communication problems arise when personnel and computer hardware change.

Objectives of Structured Analysis

I specify these objectives of structured analysis. 1) Correctness in abstracting the essential components and variables of the real world situation for simulation. 2) Correctness in understanding the interrelationships among the components and variables. 3) Ease and correctness of communication between the user and the systems analyst, 4) Assurance that the results are correct, 5) Ease of revision of the simulation model at some future date.

Functions of Structured Analysis

Structure analysis provides the simulation developer with formal structure and rules to aid creative thought in the abstraction and development process. Creative thought is not wasted in brain storming activity of an aimless and nonrecorded manner. Yet, the rules and criteria should enhance, not stifle, creative thought. Structure analyses can be presented in highly communicative visual forms which enhance both the analytical and the communication process.

Example--Craftco Manufacturing Company

Tables I-IV present examples of abstraction and structure development of parts of a very simple simulation exercise. This simulation has the objective of permitting students to study the impact of managerial decisions on profits. Each quarter the students exercise managerial decisions on five variables; price, advertising, sales force, administrative overhead, and production schedules. At the very elementary level of abstraction I wanted the students to operate the simulation for ten quarters during a semester. For output I wanted each student to receive an inventory record, a production cost record, a balance sheet, and an income statement reflecting results of their operating decisions for each quarter.

At a more operational level structure is used for the development of a balance sheet, Tables II-IV. My problem was to carry the balance sheet abstraction to that point where the form and operation would be simple, yet meet the accounting standards of form and accuracy. At best the balance sheet requires much data which are developed from various sources. Thus identification of appropriate components, variables and interrelationships

Computer Simulation and Learning Theory, Volume 3, 1976

must be carried out with care. Table V shows a list of variables used. in the balance sheet component. For purposes of illustration Table VI exhibits the significant characteristics of cash. This listing of characteristics fulfills the subdivision of the hierarchical set according to Warniers' second criterion.

Conclusions

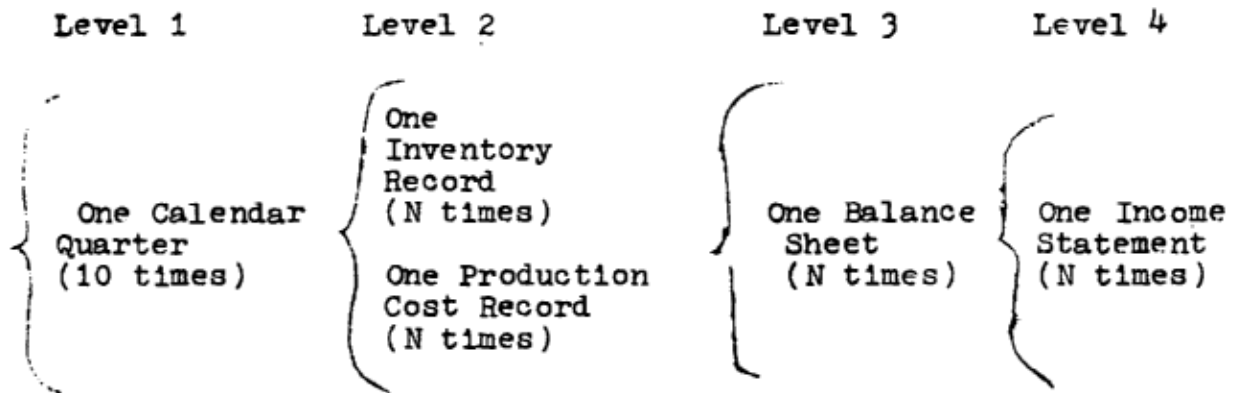
I find, structured analysis to be of significant help in the simulation development process. It provides the mechanism to insure completeness in logic flow from the beginning to the end of the project. In addition it utilizes the visual reference and the detail of components, variables, and interactions to make communication between the simulation developer and. the systems analyst clear and efficient.

TABLE I

CRAFTCO MANUFACTURING COMPANY, INCORPORATED

Structure of Results Desired for Student Management Teams

Semester assignment of ten calendar quarters operation



N = Student teams

TABLE II

Balance Sheet Structure

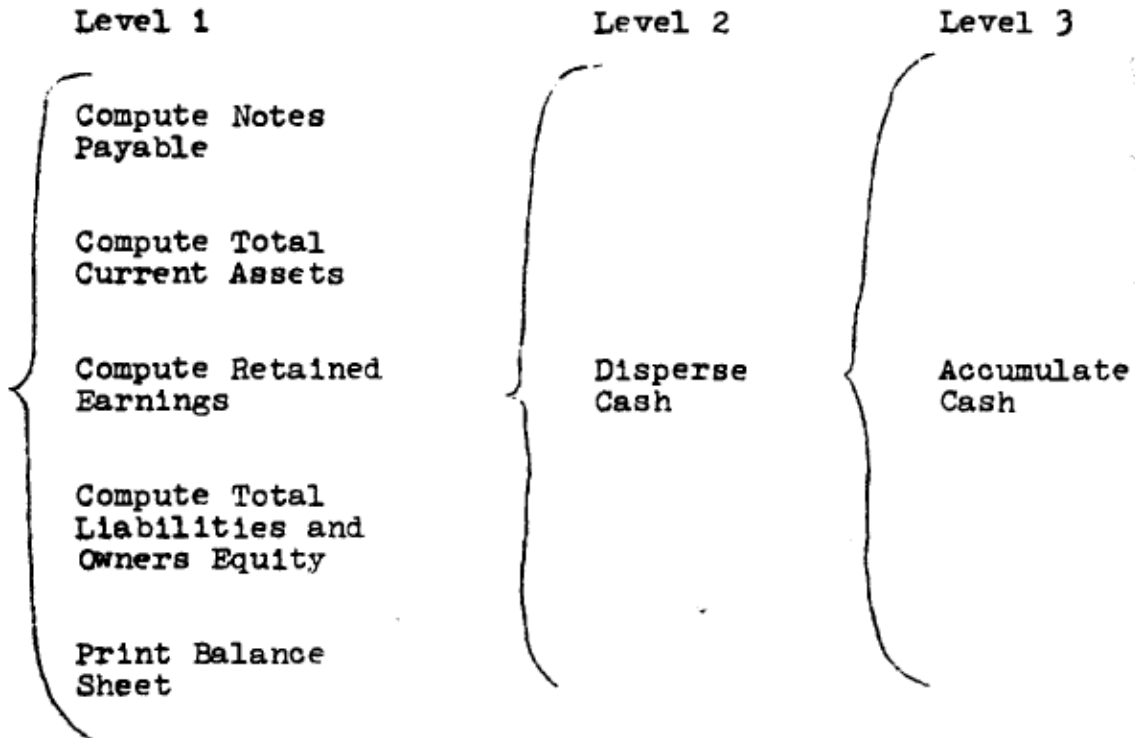


TABLE III

Structure to Disperse Cash

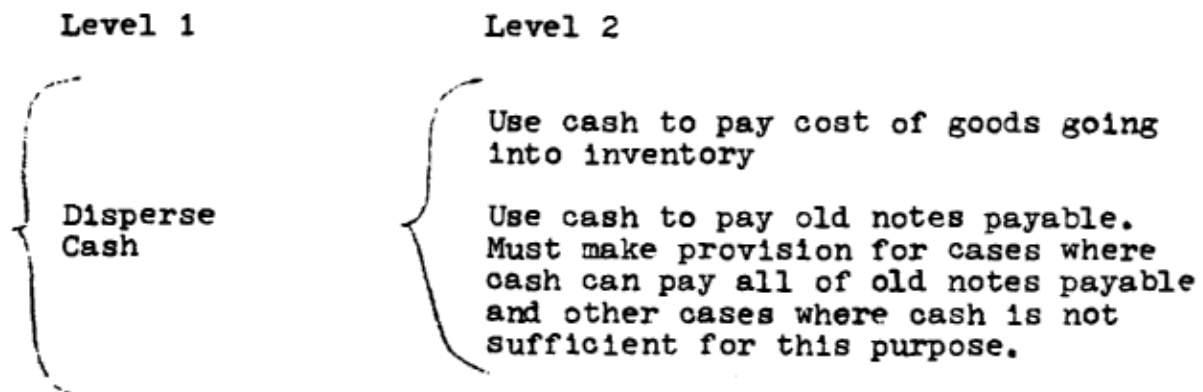


TABLE IV

Structure to Accumulate Cash

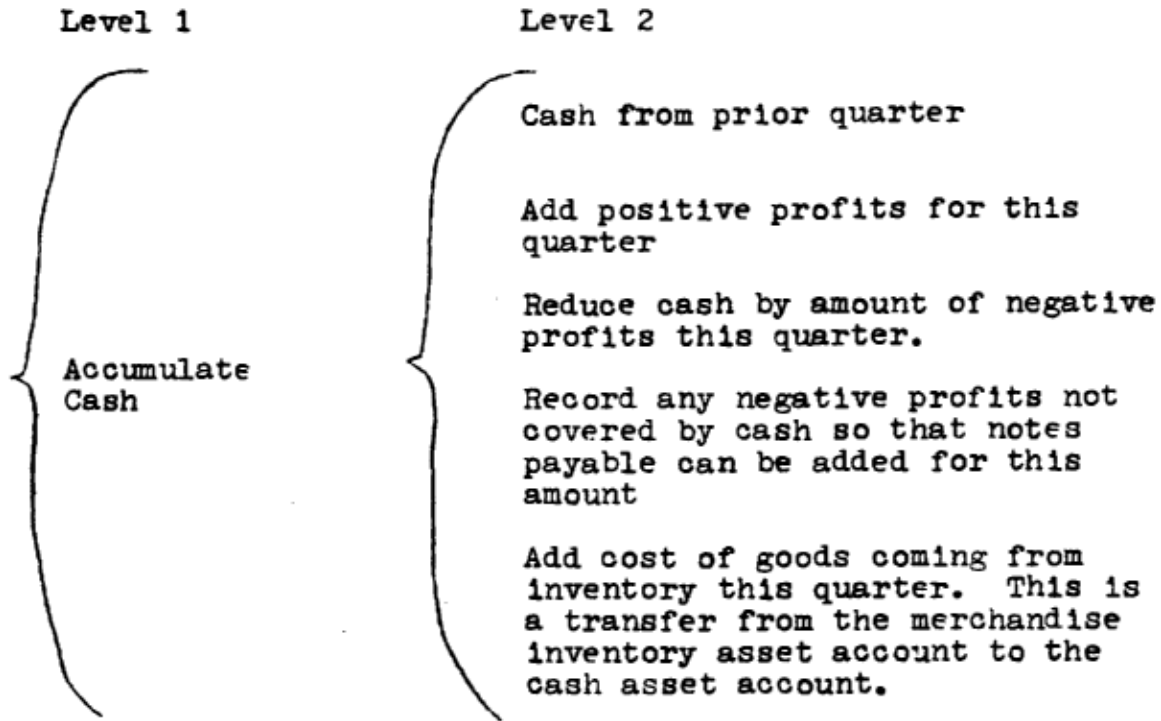


TABLE V

List of Balance Sheet Variables

Cash

Merchandise Inventory

Total Current Assets

Notes Payable

Total Current Liabilities

Stock (constant \$100,000)

Retained Earnings

Total Liabilities and. Owners Equity

TABLE VI

SIGNIFICANT CHARACTERISTICS OF THE CASH VARIABLE

Cash ≥ 0

Cash is derived from:

1. cash carryover from past quarters.
2. current quarter operations which is reflected through positive net profits and cost of goods coming from inventory.

Cash is dispersed to:

1. pay operating expenses and cost of goods going into inventory which is reflected in net profits, positive or negative.
2. pay notes payable carried over from previous quarters.

References

1. Armstrong, Russell M., Modular Programming in COBOL, John Wiley & Sons, New York, N.Y., 1973.
2. Dahl, Ole Johan, Edsger W. Dijkstra and. C.A.R. Hoare, Structured Programming, Academic Press, London and New York 1972.
3. Dijkstra, Edsger W., "Notes on Structured Programming," Technological University Eindhoven, The Netherlands, 1970.
4. McGowan, Clement L. and. John H. Kelly, Top-Down Structured Programming Techniques, Petrocelli/Charter, New York, 1975.
5. Warnier, Jean Dominique, Logical Construction of Programs, H.E. Stenfert Kroese B.V.-Leiden, 1974.