STRUCTURED MODELING

Jay W. Cohen, Anthony M. Fiore, Roger H. Larson

International Business Machines Corporation,
Kingston, New York 12401

Abstract

Today, since late design changes are extremely costly, early performance information is required for the evaluation and support of design alternatives. This information may be generated by an accurate simulation model. A process for model development may be defined in accordance to a product's individual components. This will result in a structured or modular model that is more adaptable, and more easily understood than many performance models in use today. At each development phase the product is examined and decomposed. Each component, be it hardware or software, is defined, and structurally broken down into a segment that identifies a specific queue. Each queue resulting from this product decomposition is then modeled and called a module. Modules are then independently evaluated for accuracy. The modules are then combined, allowing for product level model verification or validation.

INTRODUCTION

Structured modeling is a process whereby a product's architecture is decomposed into discrete product components (characterized by queues) which can be independently modeled and validated. Figure 1 depicts this process for the structured development of a model. It represents a methodology that may be divided into three general steps:

- Decomposition of Product
- Model Development
- Product Level Validation and/or Verification

Step 1, Decomposition of Product, describes the breakdown of a product into its basic independent components characterized by queues. When these queues are modeled, they become modules and are independently validated as presented in Step 2. The modules are then combined to create the product (system) model. Step 3 addresses the validation and/or verification of this model. For the sake of organization, the format of the text presented in this paper corresponds to the logical flow represented in Figure 1. Each block in the diagram is described by a section of this paper and both are identically labeled. The reader is encouraged to periodically review Figure 1 as an overview of the methodology.

Proceedings of the 1982
Winter Simulation Conference
Highland * Chao * Madrigal, Editors

82CH1844-0/82/0000-0253 $00.75 © 1982 IEEE

DECOMPOSITION OF PRODUCT (STEP 1)

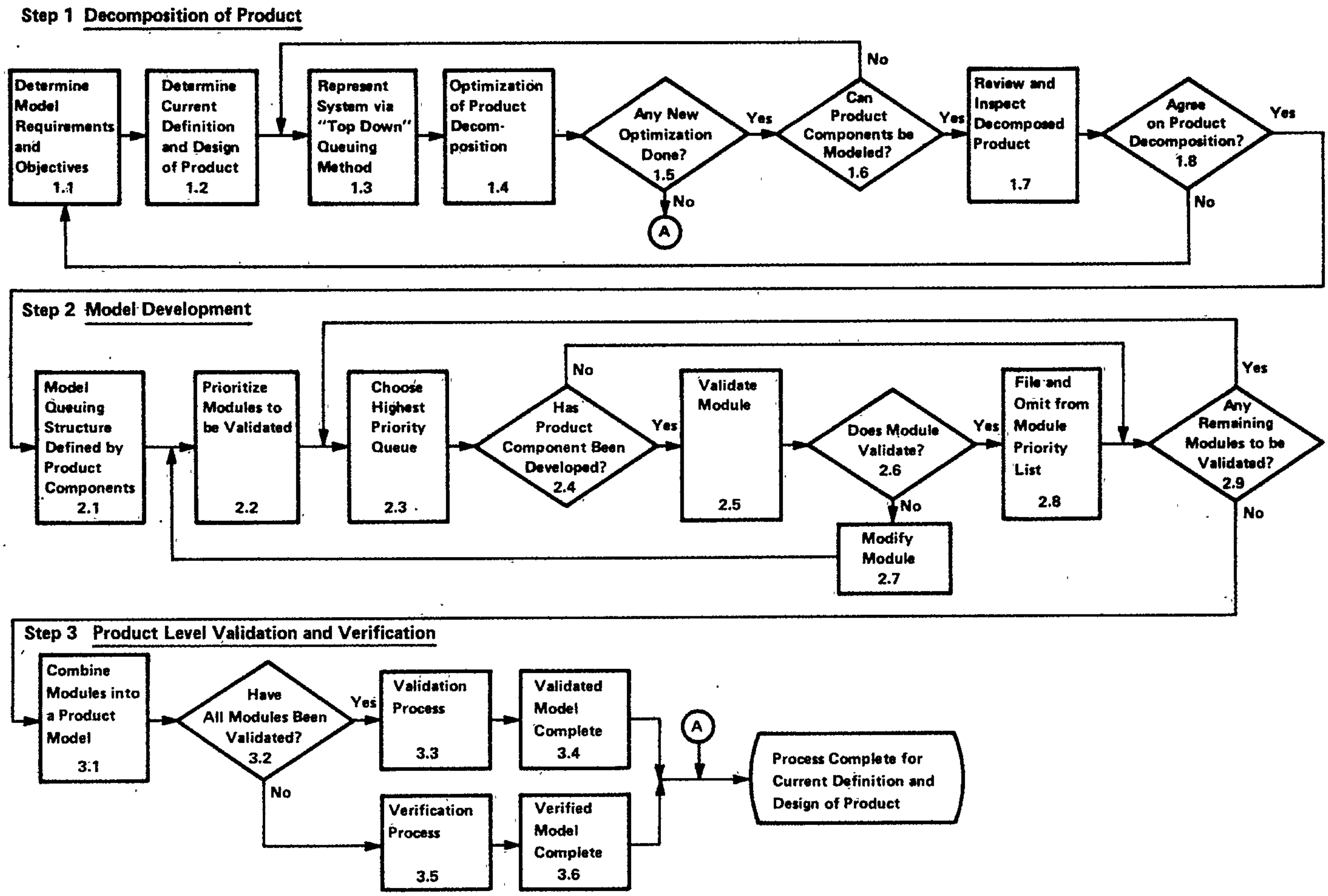
Product decomposition methodically breaks down a product into smaller units called components. It is an evolutionary process which continuously refines these components in terms of queues as the development phases progress. Whereas early phases require only a general overview of the queueing structure, later phases demand a more granular definition of the queueing characteristics of the product. The method outlined in Figure 1 should be adhered to during each phase with emphasis on specific items as required by the development cycle.

Determine Model Requirements and Objectives (1.1)

Before attempting to decompose a product which is intended to be modeled, the following eight areas must be researched and defined:

- The model's use.
- The modeling emphasis for each development phase.
- The type and detail of information required from the model.
- The level of accuracy required of the model's output.

Figure 1
STRUCTURED MODELING OVERVIEW



- The modeling experience of the group of people that will use the model.
- The independent variables (workloads, configurations, etc.)
- The level of resources required to do the job.
- The scheduling and completion requirements.

This information will aid in determining the degree of product decomposition during each phase of the development cycle.

Determine Current Definition and Design of Product (1.2)

The definition and design of the product is continuously modified as the product matures. For example, in the early phases, the product's definition is generally developed in terms of marketplace requirements. The definition is then translated, with assumptions, into a hardware and software design. This will be the basis for planning a system structure strategy. Most assumptions will eventually gain in clarity and become more accurate later in the development cycle. At later phases, the product's definition is refined in terms of the product's objectives and initial specifications. At this time, the basic definition of the product should be firm in terms of capabilities and architecture. All modifications should be carefully monitored and reviewed for their impact on the structured model.

Represent System via "Top Down" Queueing Method (1.3)

To apply the "Top Down" queueing method requires that the product's function be initially broken down into relatively large independent components. These components will become more granularized during the development process. At all phases of the development cycle, the product is represented in terms of components whose characteristics are defined by queues. The "Top Down" queueing method is an evolutionary process of decomposing coarser queues into their more refined elements. For each refinement, the queues will be transformed into model modules. At all times the structure and definition of the queues should be representative of the current product definition and strategy. It is essential that the decomposition be kept at a level where the queues can be modeled. It must also be recognized that decomposition is limited to the point where any further breakdown would add to the complexity of the model and not to its accuracy.

At each phase, the system structure is documented indicating the major independent queues and their logical connections. Entry and exit points are shown and the data flow indicated with arrows. For example, in a typical display, a keyboard (queue) communicates to an adapter (queue) which is attached to a bus (queue) that is connected to a processor (queue).

Optimization of Product Decomposition (1.4)

Optimization of product components is necessary to maximize efficiency in the development of a structured model. A discrete product component represents one queue, consisting of a defined arrival rate and service time. Each queue, regardless of the level of decomposition, must be measurable and independent. As the design of the product progresses, the level of decomposition becomes more refined. That is, as the characteristics of the product's components become more definitive, the queues become more granular. The product is decomposed into optimum components. Optimum components result when the degree of decomposition is maximized into independent and measurable queues which can be modeled and represent the current product definition and design. It must be recognized that, at times, decomposition limitations may be warranted to avoid adding unnecessary complexities to the modeling effort.

Any New Optimization Done? (1.5)

During this step of the methodology, a review is required to determine if any significant updates to the optimum modular components have been made. If there are no changes, the current queues representing the product remain unmodified.

However, if there is a change in the optimum component structure, then this requires further analysis and updates to the model.

Can Product Components Be Modeled? (1.6)

The major activity of this task is to determine whether the updated or new product components can be modeled. If the new or updated component can be modeled, then the system decomposition is complete and ready for the next step. If they cannot, then the queues representing the component must be redefined to make modeling possible. This occurs when the product decomposition has become too granular to model or measure and requires a backstep in the "Top Down" queueing process.

Review and Inspect Decomposed Product (1.7)

The decomposed product is then placed through a rigorous inspection process to determine if the decomposed components are an accurate representation of the product. This inspection is conducted by the modeling group with active participation from the product's designers, planners, and assurance groups.

Agreement of Product Decomposition (1.8)

A formal agreement that the product has been accurately decomposed must be reached by all concerned parties before the modeling activities commence. If agreement cannot be reached on the level of detail and accuracy of the product's decomposition, then a reevaluation of the definition and design of the product is required. However, if agreement is reached, then the foundation is set for modeling all new components (queues) defined by the product decomposition.

MODEL DEVELOPMENT (STEP 2)

Once agreement has been reached on the product's decomposition, the development and/or enhancements to the model can commence. Model development is an ongoing evolutionary process which corresponds to the product's development. In fact, there must be a one-to-one correspondence between the modeled queues (modules) and the components resulting from the product's decomposition. It is essential that the modeling group has continuous and open communications with the designers and planners to assure that the modules are representative of the product's components.

Model Queuing Structure Defined by Product Components (2.1)

The development of the model is dependent upon accurate product decomposition. Each optimized product component (queue) becomes an independent and measurable model module (queue). The characteristics of each queue are defined by:

- Transaction arrival rate
- Service time of transaction
- Queuing discipline (i.e. FIFO, LIFO)

The modeler will select the modeling technique (e.g. analytic, simulation) that best meets the objectives of the modeling effort. This choice is independent of the structured modeling methodology presented in this paper.

The module must reflect each component's characteristics. Each module consists of an input interface, a queue or series of queues representing a task, and an output interface. Ideally, each module will have one input and one output interface and accomplish a single task. There will be various degrees of knowledge and understanding of each component's queuing characteristics. The greater the modeler's understanding of the components and their interactions, the more accurate and successful will be the modeling effort. For those decomposed units with limited information concerning their queuing behaviors, assumptions must be made by the model developers.

Prioritize Modules to be Validated (2.2)

Once all product components have been modeled, a sensitivity study is conducted to determine the modules which have the most influence on the measurements that define the product's performance (dependent variables). The model input (independent variables) used during this evaluation is defined by the most representative workload characteristics available and should consider all the essential user environments. This task will result in a prioritized list of modules which will direct the validation effort.

Choose Highest Priority Queue (2.3)

The objective of this task is to sequentially select modules from the prioritized list for the validation process. This task allows focusing attention on those modules which are most important in terms of the product's capabilities.

Has Product Component Been Developed? (2.4)

Validation of a module is a process which is defined by a statistical comparison of the real world component to the module. Therefore, if the real product component exists and can be measured, validation of that module may commence. Otherwise, the validation process of this module must be delayed and the next one on the priority list is chosen.

Validate Module (2.5)

The validation process requires two sets of data for analysis. One set is comprised of the measurements obtained from the real component under investigation and the second set from the model predictions. Both sets are collected using identical workload characteristics (independent variables). The measurements to be analyzed (dependent variables) are chosen in a manner that reflect the capabilities of the product. It is highly recommended that statistics and experimental design be utilized for validation analysis. The end result of this activity will be a determination of whether or not a model module accurately represents the real product under specific workloads and environments.

Does Module Validate? (2.6)

If a module validates, then it is filed for future use and removed from the prioritized module list. If it does not validate, then the module must be modified to more accurately represent the product's component.

Modify Module (2.7)

When a module does not validate, then the module does not accurately represent the component. Hence, that module must be updated to more accurately reflect this component. The updated version is then returned to the list of modules for subsequent validation.

File and Omit from Module Priority List (2.8)

Once a module has been validated, it is omitted from the module list and filed for later use. This will allow other models, with similar functions, to use these modules and take full advantage of past model development and research.

Any Remaining Modules to be Validated? (2.9)

The module validation process should continue until there are no additional modules in the priority validation list that significantly affect the performance of the product. Once complete, the model is ready for system level validation.

PRODUCT LEVEL VALIDATION AND VERIFICATION (STEP 3)

This section presents a process whereby the validity of the model is quantified during the various phases of the development cycle. The success of the model, as a prediction tool, is dependent on the degree of confidence resulting from this process. One of two analysis procedures (validation or verification) is used to assess the accuracy of the model:

- Model validation occurs when all significant components can be compared to their corresponding modules.
- Model verification occurs when there exists at least one significant component that has not yet been developed and assumptions are made concerning its queuing characteristics.

The product level validation or verification process starts by combining all current modules, regardless of their validation status, into a product model. By determining the proportion of validated modules and their importance, a degree of credibility of the product model can be obtained. The result of this initial analysis will establish a preliminary confidence factor of the model. The greater this factor, the more extensive will be the validation effort.

For the first time in the structured modeling process, higher level interactions between modules can be investigated. The evaluation of higher level interactions is a statistical process utilizing experimental design and various analysis techniques.

Combine Modules into a Product Model (3.1)

The current modules are now combined into a structured model. Once this is accomplished, the model validation of the product may commence.

Have All Modules Been Validated? (3.2)

There will be times when the validation of the module cannot be accomplished because its corresponding component has not yet been developed. If this is the case, only model verification can be accomplished. However, if all modules are validated, then the validation of the structured model may commence.

Validation Process (3.3)

Model validation is a process designed to determine whether significant differences exist between module and component interactions. It involves statistical analysis based on design of experiment techniques. Deviations between the component interactions of the product and the module interactions of the model are identified and resolved. The output of this process is a model that is validated with an assigned risk.

Validated Model Complete (3.4)

The model is now complete and validated. It can be used as a tool within the constraints defined by the model objectives with a known degree of accuracy and confidence. Because of the structured design and independent modules, there is a strong potential to utilize these modules in future models. In this manner, model development takes full advantage of past research and minimizes "reinventing the wheel" for each new product.

Verification Process (3.5)

This effort is required when a significant module cannot be validated because the corresponding component has not yet been developed. It is a two step process consisting of the following:

1. Determine a preliminary confidence factor which is a subjective measure of the model's credibility.
2. Verify model using queuing assumptions characterizing the modules not validated.

A preliminary confidence factor is a measure of the model's credibility. It may be obtained by assigning to each module, regardless of its validation status, a subjective quantifiable value based on its relative importance in determining the capabilities of the product. Various alternatives are available to calculate this factor. The simplest method is to calculate a proportion, whereby the denominator is the sum of assigned weights of all the modules and the numerator is the sum of assigned weights of the validated modules. As this proportion (confidence factor) approaches 100 percent, the credibility of the model increases. This is subjective in nature but provides an evaluative measure which aids in determining the credibility of the model's results.

Once a confidence factor has been calculated, the verification process can commence. The extent of the verification effort should correspond to this factor. When this confidence factor is relatively high, it implies that:

- A high number of modules have been validated.
- Some of the important modules have been validated.
- Queuing assumptions have been minimized.

When the factor is high, a rigorous verification process may be warranted (i.e. statistical analysis of measurements). If it is low, other methods of verification may be used (i.e. tests of reasonableness).

Verified Model Complete (3.6)

With the confidence factor calculated and verification complete the following is known about the verified model:

- The model may be used as a tool with known limitations.
- Exposures have been identified due to lack of validated modules.
- Assumptions substituted for missing validated modules have been identified.

This model, although not validated at this time, is the basis for enhancements during subsequent development phases.