Proceedings of the 2005 Winter Simulation Conference
M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.

INTRODUCTION TO MODELING AND SIMULATION

John S. Carson II

Brooks Automation
Brooks Software Division – AutoMod Group
140 Dickerson Road
Marietta, GA 30067, U.S.A.

ABSTRACT

Simulation is a powerful tool for the evaluation and analysis of new system designs, modifications to existing systems and proposed changes to control systems and operating rules. Conducting a valid simulation is both an art and a science. This paper provides an introduction to discrete-event simulation and the main concepts – system state, events, processes – underlying simulation. It discusses the major world views used by simulation software. It includes a brief discussion of a number of other important issues: the advantages and disadvantages of using a simulation model, the skills required to develop a simulation model, the key steps in conducting a simulation study, as well as some project management guidelines and pitfalls to avoid.

1 DEFINITIONS AND CONCEPTS

There are many types and kinds of simulation. In this tutorial we limit ourselves to discrete-event, stochastic process-oriented simulation. This covers almost all simulations discussed at the Winter Simulation Conference. It excludes Monte Carlo-type simulations in a spreadsheet (sampling studies, financial and risk analyses, and so on). It also excludes equation-based numerical solvers, for example, differential equation solvers and other equation-based models. Although not explicitly discussed, it may include training simulations and man-in-the-loop simulations such as many conducted by the military.

1.1 Simulation Concepts

A model is a representation of a system or process. A simulation model is a representation that incorporates time and the changes that occur over time. A discrete model is one whose state changes only at discrete points in time, not continuously.

A model may incorporate logical, mathematical and structural aspects of the system or process. A discrete-event model, the type discussed in this paper and the type represented by the great majority of papers at the Winter Simulation Conference, is one based on the concepts of state, events, activities and processes. Time is a critical component. A discrete-event model is one whose state changes only at discrete times called event times. When an event occurs, it may trigger new events, activities and processes.

In concept, a model's state is a (long) vector, that is, a list of values that are sufficient to define the complete state of the system at any point in time. In practice, a model's state is defined implicitly by the internal status of all the entities used in the simulation software package.

An event is an instantaneous occurrence that changes the model's state. Examples include an arrival event for a customer at a bank, and a service completion event for the same customer. An activity is a duration of time, such as a service time or interarrival time, that is initiated by an event in conjunction with the model being in a certain state. For example, when arrivals are defined by a probability distribution of interarrival times, then when one arrival occurs (an event), the model generates a new interarrival time (an activity) which in turn will cause the next arrival event.

Primary events are those driven by data. Examples include arrival times and service completion times. In simulation terms, the primary events are scheduled to occur at some future time, calculated from data and statistical assumptions. For example, if we assume that inter-arrival times are exponentially distributed with a mean of 10 minutes, then at the time of an arrival, the model draws a new exponential sample, adds it to the current time, and schedules the resulting future time as the time of the next arrival event.

Secondary events are those generated internally by model logic. For example, in a waiting line model, when a server becomes available and there is a waiting entity (person or product) at the front of the queue, then a "service begin" event is scheduled to occur immediately.

An entity is an object in the model. Dynamic entities are created at time zero or at other times by an arrival event. Dynamic entities usually represent some real-world object that is flowing through a system. Examples include automobiles in a manufacturing model, pallets or cases in a warehouse model, passengers in an airport model, and telephone calls in a communications model. Entities have standard and customized attributes that individualize the entity.

A resource is an entity that provides a service to dynamic entities. A resource usually has a finite capacity representing some system constraint. Examples include a worker or a team of workers doing a task, a machine, or a vehicle. In some models, resources may have user-defined resource states and special characteristics such as downtimes and availability schedules.

Almost all discrete-event models are stochastic. That is, they contain some components that are modeled as a statistical distribution. This introduces random variation into a model, making it into a statistical or sampling experiment. More precisely, when one or more components are stochastic (for example, interarrival or service times), then model outputs are stochastic, necessitating some kind of statistical analysis to draw valid conclusions.

Virtually all simulation software packages include automatic collection of performance statistics, and easy collection of custom statistics.

1.2 Simulation Worldviews

Simulation modeling software usually takes one of three worldviews: event scheduling, process interaction, or activity scanning. As their names suggest, each puts its main focus on the events, the processes, or the activities in a simulation, respectively. When following an event scheduling perspective, a model developer must define the model logic and system state changes that occur whenever any event occurs.

A process is a sequence of events, activities and other time delays associated with one entity as it flows through a system. For example, a customer process at a bank consists of an arrival event (to the lobby, perhaps), joining and waiting in a queue (a delay), a service time by a teller , and finally a service completion event. In terms of concepts discussed earlier, the service time is an activity and the teller is a resource. Simulation software based on the process interaction perspective, or worldview, provide a way for a user to define a process for each entity in the system.

Activity scanning provides a way to define model logic by focusing on activities from the point of view of a resource, defining resource state changes depending on various events. For example, in the bank, the teller serves one customer until completion, then looks at the queue. If the queue is not empty, the teller “takes” the first entity out of the queue, changes its own state to “busy” and begins a new service activity. If the queue is empty, the teller changes its own state to idle.

For a more in-depth look at simulation concepts and worldviews, see Carson (1993) and Banks et al. (2005).

2 HOW IS SIMULATION USED?

A simulation model is a descriptive model of a process or system, and usually includes parameters that allow the model to be configurable, that is, to represent a number of somewhat different system or process configurations. Simple examples include parameters that allow a user to vary the number of workers at a workstation, the speed of a machine or vehicle, the timing characteristics of a conveyor control system, and so on.

As a descriptive model, you can use a simulation model to experiment with, and evaluate and compare, any number of system alternatives. Evaluation, comparison and analysis are the key reasons for doing simulation. Prediction of system performance and identification of system problems and their causes are the key results.

3 WHEN SHOULD SIMULATION BE USED?

Simulation is most useful in the following situations:

1. There is no simple analytic model, spreadsheet model or “back of the envelope” calculation that is sufficiently accurate to analyze the situation.
2. The real system is regularized; that is, it is not chaotic and out of control. System components can be defined and characterized and their interaction defined.
3. The real system has some level of complexity, interaction or interdependence between various components, or pure size that makes it difficult to grasp in its entirety. In particular, it is difficult or impossible to predict the effect of proposed changes.
4. You are designing a new system, considering major changes in physical layout or operating rules in an existing system, or being faced with new and different demand.
5. You are considering a large investment in a new or existing system, and it represents a system modification of a type for which you have little or no experience and hence face considerable risk.
6. You need a tool where all the people involved can agree on a set of assumptions, and then see (both statistically and with animation) the results and effects of those assumptions. That is, the simulation process as well as the simulation model can be used to get all members of a team onto a (more) common understanding.

7. Simulation with animation is an excellent training and educational device, for managers, supervisors, engineers and labor. (Don't tell me, show me.) In fact, in systems of large physical scale, the simulation animation may be the only way in which most participants can visualize how their work contributes to overall system success or creates problems for others.

4 THE ADVANTAGES AND VALUE OF SIMULATION, AND THE DISADVANTAGES

Simulation allows experimentation with a model of a system. Without a model, you either experiment with a real system (if it exists) – probably causing major disruptions – or proceed without such experimentation and analysis – at some potential risk. Simulation allows the identification of problems, bottlenecks and design shortfalls before building or modifying a system. It allows comparison of many alternative designs and rules of operation. Evaluation and comparisons can take place before committing resources and investment to a project.

Simulation allows study of the dynamics of a system, how it changes over time and how subsystems and components interact. A simulation model provides about the only method to study new, non-existent complex dynamic systems for which analytic or static (spreadsheet) models provide at best a low fidelity model with correspondingly low accuracy.

On the other hand, often simulations are time-consuming, data is not available or costly to obtain, and the time available before decisions must be made is not sufficient for a reliable study. In some companies, an early success with simulation has evolved into simulation becoming a "checklist" item on every project whether it is justified or not for the project at hand. In some situations, the animations and other visual displays, combined with the time pressure present on all projects, may mislead decision makers into premature conclusions based on insufficient evidence. In addition, inexperienced simulation analysts, or those too focused on (and in love with) the simulation software and technology may add too much detail to a model and spend too much time in model development, resulting in the original goals and project timelines being forgotten. This often leads management to conclude that simulation, while a promising and interesting technology, is too costly and time-consuming for most projects.

A good simulation model provides not only numerical measures of system performance, but provides insight into system performance. Insight comes from a tacit understanding of system behavior, an understanding that can be developed by intelligent use of animation and other visual aids, and an intelligent set of valid experiments together with a good statistical analysis.

5 THE SIMULATION TEAM

Simulations are conducted by in-house specialists at many companies as well as by many engineering, consulting and services companies. A few companies specialize in offering simulation services. These groups or companies may supply the simulation expertise and model development experience, but the whole team must be broader based.

5.1 The Team

Simulations are almost always conducted by a simulation team, not an isolated individual. Sometimes one individual plays several roles. The various roles include:

- The customer's executives and managers who "own" the problem, the decision-makers,
- The customer's engineers, staff, plant and line managers, and others who are involved in the problem, know key portions of the day-to-day operations and will live with implemented solutions,
- In-house or outside systems designers, who are designing a new system or changes to the existing system, and
- The simulation analyst.

People who know and understand the actual system are a key resource for project success. Even if the system itself is new and not yet built or operational, there will be people who understand the business, the processes and the end product or service; their expertise is absolutely needed. It is quite infrequent that one person alone understands the whole system in sufficient detail to provide all the information needed; rather, a number of different people are needed, each providing a bit of expertise for one small part of the system.

The project team must include all those with questions that they expect the model to address. These questions need to be identified and specified up front at project initiation, and not sprung upon the team at a final presentation where they will most likely go unanswered leading to misunderstanding and failure in the minds of some participants.

5.2 The Simulation Analyst: Skills and Software

For a simulation analyst, simulation is both an art and a science. As with any art, one learns by training and education but more importantly by practice and mentoring. Good communication skills are a necessity; a willingness not to assume anything, not to be afraid to ask "stupid" or "obvious" questions, and a willingness to ask the same question of many team members is a key to understanding and making accurate assumptions. For the science portion, programming, modeling, and a working knowledge of probability and statistics are important skills to attain. Knowledge of a simulation package, while necessary, by itself is not sufficient to be a good simulation analyst.

Models should be developed in a suitable, commercially available and supported simulation package. There are many simulation packages, some more or less general purpose, and some that specialize in either one or a few application domains (such as manufacturing, material handling, call centers, medical, transportation, logistics or other limited area of applicability). Packages offer differing levels of detail, ease of use, and skill required for effective use, as well as differing levels of user customization capability. Some have minimal or no programming, emphasizing ease of use and quickness of model development for small to medium-size models with minimal complexity. Other packages offer total customization but usually at the cost of programming skill, time and effort as well as gaining a knowledge of the selected package. Large complex models with unique rules and algorithms cannot be developed in most (or possibly any) "no-programming" packages. On the other hand, many simpler simulation models can be developed most efficiently and quickly in the simpler "point & click" environments based on flow charting.

The actual choice of software used is beyond the scope of this introduction, but is often heavily influenced by what the simulation analyst has used in the past. This is a valid consideration, as learning a new simulation package can be time-consuming, and becoming an expert in it takes a number of projects and an openness for self-education.

A new simulation analyst with a programming background may think, on seeing the price range of simulation packages, that a model can be developed in a general purpose non-simulation programming language, such as C or C++ or Visual Basic, with less expense and in about the same time. This judgment is based on inexperience; in the author's experience, using a general purpose language, even with a library of simulation routines, generally takes from 4 to 10 times the amount of analyst time for model development as using a good simulation package; maintaining or extending such a model usually requires the original developer, and can be a challenge (to put it nicely).

To be successful with most simulation packages, a new analyst needs training from an expert and ongoing mentoring for a period of time. Self-education is sometimes possible, but usually results in a "spotty" knowledge with learning gaps if used as a total solution.

6 STEPS IN A SOUND SIMULATION STUDY

Every simulation project proceeds through a set of phases and steps whose goal is a successful project. Here are some guidelines.

6.1 Project Initiation

In the first phase, projects begin with a kickoff meeting, problem formulation, objectives setting, determination of measures of performance, and details of modeling assumptions and data requirements, followed by a project plan with time and cost estimates and project timelines. The end results of this phase are the Assumptions Document and a project plan.

6.1.1 Problem Formulation and Setting of Objectives

All modeling activities should be focused on the objective. Often, the actual problem may be unknown or little understood, and problem formulation may initially be stated in terms of observed symptoms (for example, product throughput less than desired or expected). During the study, as the nature of the problem becomes clearer, problem formulation may be restated and clarified with the project team.

During this phase, the simulation team should develop a list of specific questions that the model should address, and develop a list of measures of performance that will be used to evaluate or compare the alternatives being modeled. Often, the customer has a goal in mind; for example, that the new system under a certain level of resources and manning will achieve an expected throughput. This means that if the study finds that the proposed system design or set of operating rules does not achieve the expected throughput, then the model is expected to provide information and insight into the causes, so that the simulation analyst and team can develop intelligent alternatives that have a better chance of achieving desired goals.

At this phase, the simulation analyst (or project leader) needs to ask questions of all participants and develop a set of working assumptions that will form the basis for model development. Three important overall considerations are:

- Model boundary and scope,
- Level of detail,
- Project scope.

The model boundary or scope determines what is in the model, and what is out. The model level of detail specifies how in-depth one component or entity is modeled; it is determined by the questions being asked and data availability. Think of model boundary as "width" and level of detail as "depth". Overall project scope deals with the breadth of the questions that the model will be used to address; that is, it deals more broadly with how the model will be used during the experimentation and analysis phase. As more and more questions can be asked of a given model (especially a parameterized one), the team needs a common understanding of project scope to avoid scope creep and a project with no end.

6.1.2 Overall Project Plan

With the information developed during problem formulation, the simulation analyst should develop time estimates and project timelines for model development, verification and validation, and experimentation and analysis – all the steps in a simulation.

With these time (and cost) estimates in hand, management can decide whether to proceed with the simulation study, or possibly to expand or limit its scope.

6.1.3 Conceptual Model and Assumptions Document

The set of agreed-upon assumptions and data is, in essence, the conceptual model. These assumptions and data requirements should be detailed in an Assumptions Document or Functional Specifications Document.

The Assumptions Document should be written in the language of the real system and the people who work in that system. It should not use modeling language or jargon peculiar to any particular simulation software or language. After all, its purpose is to communicate a set of assumptions and data requirements among all members of the simulation team, not all of whom will be, or even need to be, simulation experts. With this common document, the team can revise the assumptions until all members agree to a common set of working assumptions, or at least to note disagreement until agreement can be reached.

In summary, project initiation has these essential activities:

- Get all interested parties involved in project kick-off, initial problem formulation and meetings discussing model assumptions. If a person on the customer or client side will be present at any review meetings or final presentations, that person must be present at these initial meetings. If a person expects the model to address certain questions, that person must put the questions on the table at project initiation.
- Put all assumptions and data requirements into writing. Include objectives, specific questions to address, and measures of system performance. A written Assumptions Documents is essential. A reviewed, and signed-off, Assumptions Document is critical.

6.2 Project Work

The project "work" consists of model development and data collection. The end result is a working model with customer-provided and validated data. The working model is subjected to verification and validation in the next phase.

6.2.1 Model Development

Model development consists, in a nutshell, of two major activities: (1) development of data structures to represent the data needed by the model, and (2) translation of the modeling assumptions in the Assumptions Document into the language or representation required by the simulation package. The simulation analyst must design data structures that represent the data and its inter-relationships as well as fit into those allowed by the simulation software. For example, almost all packages allow variable arrays, most allow tabular displays of data (and referencing of that data by model entities and processes), and some allow lists of objects and data.

6.2.2 Data Collection, Cleansing and Analysis

The customer or client usually collects the agreed-upon data. If data is not available or (subjective) estimates are being used, then the customer should review these data assumptions internally with people knowledgeable in the processes involved.

Data sources include databases, manual records, automatic data collection systems, sampling studies and time studies. Unfortunately, it seldom happens that all or even much of the needed data is readily available, or when available that it is of the desired quality. In these circumstances, much effort and expense may be required to collect the data or extract it from existing databases.

After collecting it, a further effort may be required to validate and "cleanse" the data. Even data in customer databases, surprisingly to some, may be suspect. Often simple tests or audits may show that what appears to be data availability is data garbage. For example, when simulating a distribution center and using actual customer orders to drive the model, we found that order files were indeed quite accurate (after all, the company is paid by customers who receive what they order!). In contrast, the master SKU list had many inaccuracies. For each SKU, the master SKU list was supposed to give pallet weight and pallet height, but these were inaccurate in up to 50% of the 100,000 SKUs listed. The reason was simple: in the existing manual system, the forklift drivers did not use this data to decide where to store a pallet; no one used it. In the proposed computer-controlled new building which we were simulating, this data was essential as all storages and retrievals of pallets into rack were under software control. A great deal of effort was required to cleanse the data to make it accurate; on the positive side, this effort was required before the new system could be put into operation.

When data on an activity is available, and the data exhibits random variability, that is, variability for which no immediate cause is evident, then the activity duration is usually modeled by a statistical distribution. Sometimes the empirical distribution of the data is used; sometimes a statistical package is used to fit a distribution to the data.

With some types of data, the analyst may decide to use the actual data itself as input to the simulation. This may be done at customer request, or because it is too difficult to represent the data as a statistical distribution. For example, we often use customer order files as input to a model of a distribution center or order fulfillment center. Each customer order may consist of a number of line items, and each line item has a quantity of one or more. There is almost always a correlation between number of line items and quantity, a correlation that would be difficult to characterize and represent with a statistical distribution. In this and similar situations, we have decided to use actual order files to drive models. To get representative variation, we ask the customer to provide several different samples of order files (that is, orders from several different days). If there is a need to experiment with greater demand (more orders), we can combine different order files into a single order file. If there is a need for a different order profile (perhaps more small orders and fewer large orders), we partition the order file appropriately and sample or recombine to get a desired profile.

6.3 Model Verification and Validation

In this phase the simulation analyst verifies the model, and working with the customer, validates the model. If problems are found, the model or the data, or both, are corrected. The end result of the V&V phase is a verified, validated model that is judged to be accurate enough for experimentation purposes over the range of system designs contemplated.

6.3.1 Model Verification

In model verification, the simulation analyst checks the model, using a number of different techniques, to verify that the running model agrees with the Assumptions Document. This is more than debugging in the programming sense. All model outputs should make sense and be reasonable over a range of the input parameters.

Numerous techniques should be applied, including but not limited to: (1) stress testing, or testing with a wide range of parameters and different random numbers; (2) a thorough review of all model outputs, not just the primary measures of performance, but numerous secondary measures; (3) using the software's debugger, animation and any other tools provided; (4) using selective traces, especially for complex portions of the logic; and (5) review by a more senior simulation professional (especially valuable for the relatively new practitioners).

A valuable attitude to take is the one of a true scientist. First, make an hypothesis: the model is "correct". Second, try as hard as you can to prove the hypothesis is false; that is, try to prove that the model is "bad" in some way. If only after great effort, you have only confirmations and no evidence of a faulty model, then conclude (tentatively) that the model is verified. From a scientific perspective, the best that can be achieved is a tentative verification; a future test, or a change in conditions or data, may detect a problem with the model requiring changes. Simply put, there are virtually an unlimited number of potential tests that could (theoretically) be carried out to test a model's validity; in practice, we have the time for only a certain number of them. So the best we can achieve is a "failure to reject" the hypothesis of a "correct" or valid model.

6.3.2 Model Validation

Model validation gets the customer involved. After the simulation analyst is convinced that the model is accurate and verified, the analyst should conduct a thorough model review with the customer team. It is important to have all members of the customer team who may have an interest or "investment" in the model, and who expect the model to answer their questions. If a team member will be present at meetings to present model results, that team member should be present at validation review meetings (and, indeed, at earlier project kickoff meetings).

Numerous techniques, similar to those used during verification, may be used during model validation, including (1) use of animations and other visual displays to communicate model assumptions, (2) output measures of performance for a model configuration representing an existing system or an initial design, so that team members may judge model reasonableness. If sufficient data has been collected on a real-world system that matches one of the model's possible configurations, more formal tests may be conducted comparing the real system to the model.

For more discussion of V&V, see Carson (2002); in addition, a subsequent talk on model verification and validation in the introductory tutorial track will provide more detail on appropriate techniques and issues.

6.4 Experimentation, Analysis and Reporting

The purpose of this phase is to meet initial project objectives: to evaluate and compare system performance, and to gain insight into the system's dynamic behavior and, in particular, into any problems or bottlenecks identified by the analysis.

6.4.1 Experimental Design

Before conducting simulation experiments, the analyst must decide a number of issues:

1. The input parameters to be varied, their range and legitimate combinations,

2. Model runlength (how long to run the simulation),
3. For steady-state analyses, the model warm-up period,
4. Number of statistical replications.

Earlier informal experimentation during model development and the V&V phase should assist the analyst in making intelligent decisions regarding these questions: For steady-state simulations, what is a reasonable warm-up or transient period? What is a reasonable runlength? How many statistical replications are needed? In earlier phases, the analyst should explore inherent model variability – the range of short-term behavior – which should provide at least initial insight into appropriate model runlength and number of replications needed for later experiments.

Model runlength may be dictated by the nature of the system or the available data, such as when simulating one day's operation of a distribution center, one sort tour for a sortation hub for overnight packages, a one-shift ramp-up of a manufacturing line, or any other data-driven model where the data represents a fixed period of time. In one project we had a production schedule for a packaging line for one week, and hence the model runlength was one week based on this customer-supplied data. In contrast, inherent and high system variability together with a desire for a certain level of statistical accuracy (width of confidence intervals) combined to require upwards of 100 statistical replications for each point in the experimental design (each system configuration). Other models with less inherent variability have required only 3 to 5 replications. In other models, model runlength may be under the analyst's control, for example, for the future operation of a new port design or a 24/7 manufacturing system.

There is no rule of thumb for runlength or number of replications. Each is model dependent. The number of replications affects statistical accuracy of performance measures; specifically, it affects the width of any confidence interval estimators. Other talks in the introductory track address these and other statistical issues.

6.4.2 Experimentation

The project plan developed during project kickoff and initiation provides the initial guidelines for a set of experiments. Usually simulation models are used to compare a large number of alternatives, and perhaps to evaluate in greater detail a small number (1 or 2) of recommended alternatives. The Assumptions Document should include a description of expected model variations, including the range of each model input parameter, to be simulated to represent the alternatives of interest.

In practice, initial model experiments often raise new questions and may change the direction of the study after initial experiments are run and analyzed. For example, initial experimentation may establish that a proposed new design or set of operating rules leads to major bottlenecks or other problems, and some major re-thinking of system design is required. At the very least, initial experimentation may change the direction of subsequent experiments.

In each phase of the experimentation, actual model configurations should be guided by an experimental design that lays out the model parameters being varied, the range of each parameter, and the parameter combinations that make sense.

6.4.3 Analysis

Analysis is based on the agreed-upon measures of system performance. Typically in manufacturing and logistics applications, there are measures of throughput, resource utilization, queuing and bottlenecks.

It often happens that initial experiments produce outputs that identify a problem, or symptoms of a problem, but do not readily provide the causes of the problem or provide enough information to give insight into the nature of the problem. Such insight is critical if the team is to develop suggestions for design or operating improvements that have a good chance of solving the identified problem.

In this perplexing situation, the experienced simulation analyst will use the model as a basis for forming hypotheses regarding the causes of any identified problems. Then the analyst may need to add auxiliary measures of performance to further pinpoint the cause, and most importantly, to confirm that the hypothesis is correct. In any number of simulation projects over many years, the author has seen the need to use a model to dig into problem causes that are not obvious at first sight and to devise new measures of performance to confirm hypotheses regarding the causes of system failure. The insight gained is invaluable when it comes to suggesting changes to improve system performance in order to meet a goal.

6.4.4 Reporting

Reporting of the results of experimentation and analyses usually includes one or more presentations and a written report. It is wise to have both.

Presentations allow question and answer and expansion of explanations. The response to the presentation should be used to finalize the report and to address issues and questions that arose during the presentation.

The final report should include the Assumptions Document, appropriately revised to include any changes that arose during the course of the project, as well as, of course, the key results and recommendations of the study.

7 MANAGING A SIMULATION PROJECT

When managing a simulation study through its various phases, a good manager should follow these guidelines and watch for a number of potential pitfalls.

- Have clearly defined and achievable goals. Keep the goals in mind during the whole project.
- Allocate adequate resources. Be sure that key personnel have proper skills.
- Get upper management support and buy-in.
- Have periodic review meetings with all key people present. Keep communications open.
- Don't be afraid to ask obvious or "stupid" questions. It's always better to confirm than to assume.
- Assume nothing. Confirm everything.
- Develop a common understanding on project scope and goals, questions to be addressed, and just as importantly, questions not to be addressed.
- Documents assumptions and all changes to assumptions.

Potential pitfalls and causes of project failure include:

- Scope creep,
- Time slippage/project overrun,
- Too much detail,
- Wrong skill sets,
- Key people showing up for the first time at the final presentation and asking questions that have not been addressed.

The pitfalls can be avoided by following the guidelines presented here and in more detail in Musselman (1994).

8 FURTHER STUDY

Banks (1998) provides a comprehensive, up-to-date overview of simulation. Standard textbooks that are not software-specific include Banks et al. (2005) and Law and Kelton (2000).

There are numerous texts and references for general simulation education as well as software-specific information and training. See, for example, the papers in the introductory tutorials and software tracks in this and past year's proceedings of the Winter Simulation Conference, available via <www.wintersim.org/pastprog.htm>.

ACKNOWLEDGMENTS

The author has benefited from many papers and presentations in previous proceedings of this conference, especially the papers in the introductory tutorial track. In addition, the author has benefited from many discussions with Jerry Banks on these and related topics. This paper is based on a paper of the same title in the *Proceedings of the 2004 Winter Simulation Conference* (Carson 2004), with revisions and additions.

REFERENCES

Banks, J., ed. 1998. *Handbook of simulation: Principles, methodology, advances, applications, and practice.* New York: John Wiley.

Banks, J., J. S. Carson II, B. L. Nelson, and D. M. Nicol. 2005. *Discrete-event system simulation*, 4th Ed. Upper Saddle River, New Jersey: Prentice-Hall.

Carson, J. S. 1993. Modeling and simulation world views. In *Proceedings of the 1993 Winter Simulation Conference*, ed. G. W. Evans, M. Mollaghasemi, E. C. Russell, and W. E. Biles, 18-23. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Carson, J. S. 2002. Model verification and validation. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 52-58. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Carson, J. S. 2004. Introduction to modeling and simulation. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, 9-16. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Law, A. M. and W. D. Kelton. 2000. *Simulation modeling and analysis*, 3rd Ed. New York: McGraw-Hill.

Musselman, K. J. 1994. Guidelines for simulation project success. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila, 88-95. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

AUTHOR BIOGRAPHY

JOHN S. CARSON II is the Consulting Technical Manager for the AutoMod Product Group of Brooks Software. He has many years experience in simulation in a wide range of application areas, including manufacturing, material handling, warehousing and distribution, transportation, ports and shipping, and medical and health care systems. Before joining the AutoMod Group, he was President and founder of Carson/Banks & Associates, an independent simulation consulting firm. Before that, he taught simulation and operations research at Georgia Tech, the University of Florida and the University of Wisconsin. He is the co-author of two university level textbooks including the widely used Discrete-Event Systems Simulation (fourth edition, 2005). He holds a Ph.D. in Industrial Engineering and Operations Research from the University of Wisconsin-Madison, and is a member of INFORMS. He can be reached at <john.carson@brooks.com>.