

PITFALLS IN MANAGING A SIMULATION PROJECT

Edward J. Williams

131B Fairlane Center South
College of Business
University of Michigan – Dearborn
Dearborn, Michigan 48126, USA

Onur M. Ülgen

PMC
15726 Michigan Avenue
Dearborn, Michigan 48126, USA

ABSTRACT

When simulation analyses first became at least somewhat commonplace (as opposed to theoretical and research endeavors often considered esoteric or exploratory), simulation studies were usually not considered “projects” in the usual corporate-management context. When the evolution from “special research investigation” to “analytical project intended to improve corporate profitability” began in the 1970s (both authors’ career work in simulation began that decade), corporate managers naturally and sensibly expected to apply the tools and techniques of project management to the guidance and supervision of simulation projects. Intelligent application of these tools is typically a necessary but not a sufficient condition to assure simulation project success. Based on various experiences culled from several decades (sometimes the most valuable lessons come from the least successful projects), we offer advisories on the pitfalls which loom at various places on the typical simulation project path.

1 INTRODUCTION

For the last half-century, simulation using computers has been a respected analytical technique for the analysis and improvement of complex systems. Originally, and as the first simulation languages (GPSS, SIMSCRIPT, SIMULA, GASP, etc.) were being developed for mainframe use, simulation studies were rare, special events in corporations. Many such studies were first undertaken as an investigation into the ability of the then novel technology to contribute to profitability of operations. In practice, these systems are often directed toward manufacturing and production (historically the earliest and largest commercial application of simulation (Law and McComas 1999)), delivery of health care, operation of supply chains, public transport, and delivery of governmental services. As simulation proved its value and availability of simulation expertise migrated from universities to corporations and government organizations, corporate and government managers quickly realized the importance of running simulation projects under the control of project management tools which track progress, monitor resource allocation, and devote particular attention to the critical path (Kerzner 2009). Numerous authors have developed step-by-step templates, at various levels of detail, for the delivery of simulation project results. For example, (Ülgen et al. 1994a) and (Ülgen et al. 1994b) present an eight-step approach to undertaking simulation studies. More finely subdivided overviews (Banks 1998), (Banks et al. 2010) enumerate twelve steps, from problem formulation to *implementation* of the recommendations of the study; it is this overview we will follow in describing the pitfalls to avoid.

The subsequent sections of this paper will explore the various pitfalls at each step of a typical simulation study, with examples and suggestions for avoiding these hazards. As befits the authors’ experience, some of these suggestions, particularly those in sections five and six, apply primarily to discrete-event process simulation. Others apply generally to all simulation application domains. Further, we present an

explicit project management *milestone* defining the completion of each phase. We conclude with suggestions for developing a corporate “habit of success” in simulation work.

2 PROBLEM FORMULATION

First, the problem must be stated clearly; perhaps by the client, perhaps by the simulation analyst, and (best) by the client and analyst working together. The objective statement should *never* begin “To *model*...” Modeling is a means to an end, not an end in itself (Chung 2004). The best problem statements specify numeric questions using key performance indicators (KPIs). Examples: “Will the proposed design of the milling department succeed in processing 50 jobs per hour?” “Will the envisioned reallocation of personnel in the bank reduce the probability a customer waits more than 10 minutes in line to below 0.05?” “Of the two proposed warehouse configurations, will the more expensive one achieve inventory levels not exceeding 80% of the inventory levels of the cheaper alternative?” When formulating these goals for the simulation project, the client and the modeler should also reach concurrence on assumptions to be incorporated into the model. In the milling department example, such an assumption might concern the rate at which incoming work will reach the department from upstream.

Milestone: The client and the analyst jointly sign a statement of quantitative problem formulation.

3 ESTABLISHMENT OF PROJECT PLAN

With the objective of the simulation project fixed (at least for the undertaking of current work – change requests may arise later and must then be negotiated anew), this step establishes the plan to achieve it. Details within this plan must include the scenarios to be examined and compared, the deliverables of the project and their due dates, the staffing levels to be used, and the software and hardware tools to be used. Projects frequently fail (more accurately, “evaporate into nothingness”) because this plan fails to specify that the client’s personnel, such as engineers, reporting to the client, will need to make themselves available to the simulation analyst(s) for questions and discussion – no doubt over and above the normal duties of these engineers. Questions too often omitted in the project plan are:

1. Can the questions to be posed to the simulation, as formulated in the previous step, be answered soon enough to benefit the client?
2. Will personnel of the client be taught to run the simulation model (as opposed to simply receiving a report of results)?
3. Will personnel of the client be taught to modify the simulation model?
4. Will input enter the model and/or will output exit the model via spreadsheets (often *much* more convenient for the client, requiring extra effort by the analyst)?
5. Exactly how much of the real-world system (existing or proposed) will be incorporated into the model?

A common cause of simulation failure appears, as identified by (Keller, Harrell, and Leavy 1991), when the first question is not asked – or a negative answer to it is airily brushed aside. If client management must make a decision within one month, a two-month modeling effort to guide that decision is as useless as “This marvelous computer program can forecast tomorrow’s weather perfectly, and the run will finish a week from Wednesday.”

Another major pitfall lurks in the fourth question. Far too often the client and the modeler jointly decide, in an initial flush of enthusiasm, to model *too much* (e.g., the entire factory versus the milling department, the entire hospital versus the emergency room, the entire bank versus the business loan department). Analyzing a smaller system thoroughly and correctly is easier, faster, and far superior in outcome than analyzing a larger system superficially. This precaution is especially germane if the client is relatively new to simulation and perhaps trying to “sell” simulation as a useful analytical tool to upper management (Williams 1996). The analyst is responsible for assuring the client that it is far easier to expand the scope of a valid model subsequently than it is to retroactively restrict the scope of a muddled, over-

ambitious model. In the health care field specifically, (Jacobson, Hall and Swisher 2006) provide many examples of intelligent restriction of project scope. Whenever the analyst is organizationally separate from the client (i.e., not internal consulting), the project plan must surely include cost and payment schedule agreements.

Milestone: The client and the analyst jointly sign a project proposal statement.

4 CONSTRUCTION OF CONCEPTUAL MODEL

The simulation analyst next becomes responsible for constructing an abstract representation of the system whose scope was defined in the previous step. This abstraction may involve discrete variables (number of customers waiting in line, status (busy, idle, down, or blocked) of a machine, number of parts in a storage buffer) and/or continuous (quantity of product in a chemical tank, concentration of pollutant in emissions gases, rate of growth of a predator species). In a discrete-event simulation model, the conceptualization must specify the arrivals to and the departures from the simulated system. It must also specify the logical flow paths of parts within the system, including queuing locations and visits to servers (Bratley, Fox, and Schrage 1987). Furthermore, the conceptual model must incorporate provision for gathering and reporting outputs sufficient to address the quantitative questions posed within the project plan.

The wise analyst avoids two deep pitfalls during the construction of the conceptual model. The first pitfall is inadequate communication with the client. The analyst should “think through the model out loud” with the client, to the extent the client feels comfortable with the approach to be taken during the actual modeling and analysis phases. Second, the modeler must avoid adding too much too soon. Details such as material-handling methods, work shift patterns, and downtimes may ultimately be unnecessary – or they may be absolutely essential to an adequate analysis. Therefore, the model should be only as complex and detailed as required – no more so. When these details are necessary, they should be added after the basic model is built, verified, and validated. It is far easier to add detail to a correct model than it is to add correctness to a detailed but faulty model. Often many of the details are unnecessary – as (Salt 2008) cautions, the “unrelenting pursuit of detail” (“trifle-worship”) characterizes the sadly mistaken conviction that detail = correctness, and therefore more detail is better than less.

Milestone: The conceptual model is described in writing, and the client and the analyst concur in the description. This description includes details of the data which will be required to drive the model when implemented in computer software.

5 DATA COLLECTION

In the Land of Oz, the needed data have already been routinely collected and archived by the client. In terrestrial practice, however, even if the client assures the analyst that needed data (as defined in the immediately previous milestone) are available, many problems typically loom. For example, the client may have summary data (e.g., arrivals at the clinic per day) when the model will need arrival rates per hour. Or, the client may have sales data but the model needs delivery data. Downtime data, if needed for the model, are typically more difficult to obtain (both technically and politically) than cycle-time or service-time data (Williams 1994). Clients, especially those new to simulation, are prone to view model construction as an esoteric, time-consuming process and data collection as a routine, easy task. However, the reality is usually the reverse. Therefore, the key pitfalls the analyst must avoid (and help the client avoid) are:

1. Underestimating the time and effort the client will need to expend in gathering data.
2. Alerting the client to subtleties in data collection.
3. Failing to exploit concurrency possible in data collection and the steps (discussed next) of model construction and verification.

Examples of these subtleties are:

1. Is the client conflating machine down time with machine idle time (basic but common error)?

2. Is worker walk-time from station to station dependent on task (an orderly pushing a gurney or wheelchair will travel more slowly than one not doing so)?
3. Is forklift travel distance from bay A to bay B equal to the travel distance from bay B to bay A (not if one-way aisles are designated to enhance safety)?
4. Do cycle times of manually operated machines differ by shift (e.g., the less desirable night shift has less experienced workers whose cycle times are longer)?

Milestone: Data collected match the data requirements specification developed at the previous milestone, and the data are therefore ready to support model validation.

6 MODEL CONSTRUCTION, VERIFICATION, AND VALIDATION

Model construction and verification can and should proceed in parallel with data collection. Even more importantly, the analyst should avoid the pitfall of building the entire model and then beginning verification. Rather, the model should be built in small pieces, and each piece verified prior to its inclusion in the overall model (Mehta 2000). This “stepwise development and refinement” approach, whose merits have long been recognized in the software development industry (Marakas 2006), permits faster isolation, identification, and elimination of software defects. Such model defects, for example, may involve inadequate attention to whether items rejected as defective will be rejected or reworked (e.g., will a defective part be reworked once and then rejected if still defective upon retest?). As another example, does the model encompass customer behavior such as balking (refusing to join a long queue), reneging (leaving a queue after spending “too much time” waiting), or jockeying (leaving one queue to join another parallel and ostensibly faster-moving queue)? Furthermore, the analyst building the model should seek every opportunity to ask other people knowledgeable in the computer tool of choice to search for problems, using “fresh eyes.” Modern software tools for simulation model development include tracers, “watch windows,” animation capabilities, and other tools which are a great help to verification – if the analyst is wise enough to use them (Swain 2007). As a magnification of the vitally important verification and validation steps, (Rabe, Spieckermann, and Wenzel 2008) have constructed a detailed and rigorous definition of all steps and milestones pertinent to V&V.

Thorough verification and validation also requires intimate familiarity and careful attention to the internal operation of the simulation tool in use. A highly pertinent example from (Schriber and Brunner 2004) illustrates this necessity: Zero units of resource R are available; entity 1 (first in a software-maintained linked list) needs two units; entity 2 needs one unit. At time t , one unit of R becomes available. What happens?

1. Neither entity appropriates the free unit.
2. Entity 1 appropriates it and awaits the second unit of R it needs.
3. Entity 2 “passes” entity 1, appropriates the one unit of R it needs, and proceeds.

For various software tools, all these answers are correct. Several of these tools provide options which permit the modeler to choose whether (1), (2), or (3) happens.

The next step of model validation requires vigorous interaction between the analyst and the client. During validation, the analyst must inquire into subtle statistical patterns which may exist in the real system. For example, do defective parts tend to be produced in clusters? If so, a straightforward sampling of a binomial distribution for “Next part defective?” in the model will be inexact. As another example, validation of an oil-change depot simulation model exposed the fact – obvious in retrospect – that time to drain oil and time to drain transmission fluid are correlated, since both tend to increase with size of the vehicle’s engine (Williams et al. 2005). In the simulation of a health care facility, time to prepare the patient for certain procedures and time to perform those procedures may both be positively correlated with the patient’s age and/or weight, hence correlated with each other.

As a much more prosaic example of “what can go wrong” in statistical analysis of model input data, the analyst may overlook that the computer software used to build the model uses parameter λ for the exponential distribution, whereas the computer software used to analyze the input data and fit a distribution

uses parameter $1/\theta$ (or vice versa), so that each parameter is the reciprocal of the other. Such a “trivial” oversight would surely set the simulation project on the road to ruin.

Milestone: The client confirms that the model has achieved credibility and is ready to simulate the various scenarios already specified in the quantitative problem formulation.

7 EXPERIMENTAL DESIGN AND EXECUTION

For each scenario which will be examined (from “establishment of project plan,” the analyst bears responsibility for deciding the number of replications to run for each scenario, the length of simulated time to run each of these replications, and whether the simulation analysis will be terminating (warm-up time zero) or steady-state (warm-up time non-zero and sufficient to overcome transient initial conditions). Perhaps shockingly, the most common pitfall here is the “sample of size one” (Nakayama 2003). A simulation of practical value will surely incorporate randomness, and this randomness in turns means that each replication (run) is an instance of a statistical experiment. Therefore, multiple replications must be run. The analyst must keep in mind (and remind the client) that the width of confidence intervals for key performance metrics varies inversely as the *square root* of the number of replications. Halving the width of a confidence interval therefore requires quadrupling the number of replications. Furthermore, the analyst must avoid the temptation of treating successive observations within one simulation replication as statistically independent, which they rarely are. For example, the time patient n waits for the doctor and the time patient $n+1$ waits for the doctor are almost surely highly positively correlated.

Next, the analyst, in consultation with the client, must decide how the various scenarios will be compared. The most common approach is the use of two-sample Student- t hypothesis tests among pairs of scenarios. Too often, when even a moderate number of scenarios are to be compared and contrasted, analysts overlook the greater power and convenience of design-of-experiments (DOE) methods. These methods, which include one-way and two-way analyses of variance, factorial and fractional factorial design, Latin and Graeco-Latin, and nested designs (Montgomery 2012), have three advantages over pairwise Student- t comparisons:

1. A larger number of alternatives can be compared collectively, especially with the use of fractional factorial designs.
2. The presence of interactions among the differences between the scenarios can be readily detected when present.
3. Qualitative input variables (e.g., queuing discipline to use) and quantitative input variables (e.g., running speed of a crucial machine) can be intermixed within a design.

Milestone: The client and the analyst agree on the experimental design – only then do computer runs of the scenarios begin.

8 DOCUMENTATION, REPORTING, AND IMPLEMENTATION

As the computer runs, as specified in the previous step, proceed, both the client and the modeler will learn and understand more subtleties of the system. As the output accumulates, both the modeler and the client should mentally challenge it. Are results reasonable; do they make intuitive sense? (Musselman 1992). As Swedish army veterans tell their recruits, “If the map you’re reading and the ground you’re walking on don’t match, believe the ground, not the map.” Results of these runs may spawn significant further investigations; the client and analyst must agree on whether to extend the project scope or (very likely preferable) define a follow-up project. Furthermore, it is the analyst’s responsibility to ensure that project documentation is correct, clear, and complete. This documentation includes both that external to the model (project scope, assumptions made, data collections methods used, etc.) and that internal to the model (comments within the computer model explaining the details of its construction and functioning). The pitfall to avoid here has been explicitly stated in (Musselman 1993): “Poor communication is the single biggest reason [simulation] projects fail.” When thoroughly and properly documented, a simulation mod-

el can and should become a “living document,” which can evolve with the system over a period of months or years. From management’s viewpoint, such ongoing usefulness of a simulation model enormously increases the benefit-to-cost ratio of a simulation project.

Milestone: The client concurs that the model documentation is valid and complete.

9 IMPLEMENTATION

Implementation of the recommendations in the report prepared in the previous step is the province and prerogative of the client. The analyst’s work must *earn* this implementation – that is, the model must achieve *credibility*. The simulation analyst must avoid the pitfall of acting as an advocate, and instead act as a neutral reporter of implications and recommendations available from the simulation study.

10 SUMMARY AND CONCLUSIONS; NEXT STEPS

Using a commonly recognized “road map” through the chronology of a typical simulation project, we have identified significant pitfalls to avoid and milestones marking successful avoidance of them. In summary, these pitfalls include:

1. Missing, vague, or nebulous (non-quantitative) problem statement and questions the model must address.
2. Project plan absent, overambitious (in scope and/or time), or lacking specification of roles and responsibilities among consultant and client personnel.
3. Construction of a computer model inadequately supported, or not supported at all, by a conceptual model acceptable to both analyst and client.
4. Data collection truncated and/or inadequate due to omitting to ask important questions about the system being modeled.
5. Model inadequately verified and validated, both with respect to its internal logic and its use of the input data supplied to it.
6. Experimental design fails to acknowledge statistical requirements or makes inadequate use of statistical analysis methods.
7. Documentation and communication (both written and oral) within the project team missing or inadequate.

Significantly, at a time when the “point-&-click” ease of desktop (or laptop!) software use and the enticing animations such software readily now provides were still in the misty future beyond mainframe computers (“two turnarounds a day on a good day”), the admonitions of the seminal paper (Annino and Russell 1979) are at least as pertinent now as they were then.

Simulation projects are rapidly becoming larger and longer – in that they often have multiple analysts in charge, and they are more likely to extend over many months (even years), providing opportunities for later phases to learn from mistakes or omissions in prior phases. Best practices for exploiting these opportunities for institutional learning and for effective collaborations among project leaders are promising targets for future research and investigation.

REFERENCES

- Annino, Joseph S. and Edward C. Russell. 1979. The Ten Most Frequent Causes of Simulation Analysis Failure – and How to Avoid Them! *Simulation* (32,6):137-140.
- Banks, Jerry. 1998. Principles of Simulation. In *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, ed. Jerry Banks. New York, New York: John Wiley & Sons, Incorporated, 3-30.
- Banks, Jerry, John S. Carson, II, Barry L. Nelson, and David M. Nicol. 2010. *Discrete-Event System Simulation*, 5th ed. Upper Saddle River, New Jersey: Prentice-Hall, Incorporated.

- Bratley, Paul, Bennett L. Fox, and Linus E. Schrage. 1987. *A Guide to Simulation*, 2nd edition. New York, New York: Springer Verlag.
- Chung, Christopher A. 2004. *Simulation Modeling Handbook: A Practical Approach*. Boca Raton, Florida: CRC Press.
- Jacobson, Sheldon H., Shane N. Hall, and James R. Swisher. 2006. Discrete-event Simulation of Health Care Systems. In *Patient Flow: Reducing Delay in Healthcare Delivery*, ed. Randolph W. Hall. New York, New York: Springer Verlag, 211-252.
- Keller, Lucien, Charles Harrell, and Jeff Leavy. 1991. The Three Reasons Why Simulation Fails. *Industrial Engineer* (23,4):27-31.
- Kerzner, Harold. 2009. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 10th edition. New York, New York: John Wiley & Sons, Incorporated.
- Law, Averill M. and Michael G. McComas. 1999. Manufacturing Simulation. In *Proceedings of the 1999 Winter Simulation Conference*, Volume 1, edited by Phillip A. Farrington, Harriet Black Nembhard, David T. Sturrock, and Gerald W. Evans, 56-59.
- Marakas, George M. 2006. *Systems Analysis & Design: An Active Approach*. Boston, Massachusetts: The McGraw-Hill Companies, Incorporated.
- Mehta, Arvind. 2000. Smart Modeling – Basic Methodology and Advanced Tools. In *Proceedings of the 2000 Winter Simulation Conference*, Volume 1, edited by Jeffrey A. Joines, Russell R. Barton, Keebom Kang, and Paul A. Fishwick, 241-245.
- Montgomery, Douglas C. 2012. *Design and Analysis of Experiments*, 8th edition. New York, New York: John Wiley & Sons, Incorporated.
- Musselman, Kenneth J. 1992. Conducting a Successful Simulation Project. In *Proceedings of the 1992 Winter Simulation Conference*, edited by James J. Swain, David Goldsman, Robert C. Crain, and James R. Wilson, 115-121.
- Musselman, Kenneth J. 1993. Guidelines for Simulation Project Success. In *Proceedings of the 1993 Winter Simulation Conference*, edited by Gerald W. Evans, Mansooreh Mollaghasemi, Edward C. Russell, and William E. Biles, 58-64.
- Nakayama, Marvin K. 2003. Analysis of Simulation Output. In *Proceedings of the 2003 Winter Simulation Conference*, Volume 1, edited by Stephen E. Chick, Paul J. Sánchez, David Ferrin, and Douglas J. Morrice, 49-58.
- Rabe, Markus, Sven Spieckermann, and Sigrid Wenzel. 2008. A New Procedure Model for Verification and Validation in Production and Logistics Simulation. In *Proceedings of the 2008 Winter Simulation Conference*, edited by Scott J. Mason, Ray R. Hill, Lars Mönch, Oliver Rose, T. Jefferson, and John W. Fowler, 1717-1726.
- Salt, J. D. 2008. The Seven Habits of Highly Defective Simulation Projects. *Journal of Simulation* (2,3):155-161.
- Schriber, Thomas J. and Daniel T. Brunner. 2004. Inside Discrete-Event Simulation Software: How It Works and Why It Matters. In *Proceedings of the 2004 Winter Simulation Conference*, Volume 1, edited by Ricki G. Ingalls, Manuel D. Rossetti, Jeffrey S. Smith, and Brett A. Peters, 142-152.
- Swain, James J. 2007. New Frontiers in Simulation. In *OR/MS Today* (34,5):32-35.
- Ülgen, Onur M., John J. Black, Betty Johnsonbaugh, and Roger Klungle. Simulation Methodology in Practice – Part I: Planning for the Study. In *International Journal of Industrial Engineering: Applications and Practice* (1,2): 119-128.
- Ülgen, Onur M., John J. Black, Betty Johnsonbaugh, and Roger Klungle. Simulation Methodology in Practice – Part II: Selling the Results. In *International Journal of Industrial Engineering: Applications and Practice* (1,2): 129-137.
- Williams, Edward J. 1994. Downtime Data -- its Collection, Analysis, and Importance. In *Proceedings of the 1994 Winter Simulation Conference*, edited by Jeffrey D. Tew, Mani S. Manivannan, Deborah A. Sadowski, and Andrew F. Seila, 1040-1043.

- Williams, Edward J. 1996. Making Simulation a Corporate Norm. In *Proceedings of the 1996 Summer Computer Simulation Conference*, edited by V. Wayne Ingalls, Joseph Cynamon, and Annie Saylor, 627-632.
- Williams, Edward J., Justin A. Clark, Jory D. Bales, Jr., and Renee M. Amodeo. 2005. Simulation Improves Staffing Procedure at an Oil Change Center. In *Proceedings of the 19th European Conference on Modelling and Simulation*, edited by Yuri Merkurjev, Richard Zobel, and Eugène Kerckoffs, 309-314.

ACKNOWLEDGMENTS

Suggestions, comments, and criticisms from five anonymous referees have helped the authors greatly improve the content, clarity, and presentation of this paper.

AUTHOR BIOGRAPHIES

EDWARD J. WILLIAMS holds bachelor's and master's degrees in mathematics (Michigan State University, 1967; University of Wisconsin, 1968). From 1969 to 1971, he did statistical programming and analysis of biomedical data at Walter Reed Army Hospital, Washington, D.C. He joined Ford Motor Company in 1972, where he worked until retirement in December 2001 as a computer software analyst supporting statistical and simulation software. After retirement from Ford, he joined PMC, Dearborn, Michigan, as a senior simulation analyst. Also, since 1980, he has taught classes at the University of Michigan, including both undergraduate and graduate simulation classes using GPSS/H™, SLAM II™, SIMAN™, ProModel®, SIMUL8®, or Arena®. He is a member of the Institute of Industrial Engineers [IIE], the Society for Computer Simulation International [SCS], and the Michigan Simulation Users Group [MSUG]. He serves on the editorial board of the *International Journal of Industrial Engineering – Applications and Practice*. During the last several years, he has given invited plenary addresses on simulation and statistics at conferences in Monterrey, México; İstanbul, Turkey; Genova, Italy; Rīga, Latvia; and Jyväskylä, Finland. He served as a co-editor of *Proceedings of the International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation 2003*, a conference held in Rīga, Latvia. Likewise, he served the Summer Computer Simulation Conferences of 2004, 2005, and 2006 as Proceedings co-editor. He is the Simulation Applications track co-ordinator for the 2011 Winter Simulation Conference. His email address is williams@umd.umich.edu.

ONUR M. ÜLGEN is the president and founder of Production Modeling Corporation (PMC), a Dearborn, Michigan, based industrial engineering and software services company as well as a Professor of Industrial and Manufacturing Systems Engineering at the University of Michigan-Dearborn. He received his Ph.D. degree in Industrial Engineering from Texas Tech University in 1979. His present consulting and research interests include simulation and scheduling applications, applications of lean techniques in manufacturing and service industries, supply chain optimization, and product portfolio management. He has published or presented more than 100 papers in his consulting and research areas.

Under his leadership PMC has grown to be the largest independent productivity services company in North America in the use of industrial and operations engineering tools in an integrated fashion. PMC has successfully completed more than 3000 productivity improvement projects for different size companies including General Motors, Ford, DaimlerChrysler, Sara Lee, Johnson Controls, and Whirlpool. The scientific and professional societies of which he is a member include American Production and Inventory Control Society (APICS) and Institute of Industrial Engineers (IIE). He is also a founding member of the MSUG (Michigan Simulation User Group). His email address is ulgen@pmcorp.com.