Does Calibration Improve Predictive Accuracy?

There are many sophisticated parametric models for estimating the size, cost, and schedule of software projects; however, the predictive accuracy of these models is questionable. Several authors assert that a model's predictive accuracy can be improved by calibrating (adjusting) its default parameters to a specific environment. This article reports the results of a three-year, 10-study project that tests this assertion. Results show that calibration did not improve the predictive accuracy of most of the models we tested. In general, the accuracy was no better than within 25 percent of actual development cost or schedule, about one half of the time.

Software costs continue to rise in the Department of Defense (DoD) and other government agencies. To better understand and control these costs, agencies often use parametric cost models for software development cost and schedule estimation. However, the accuracy of these models is poor when the default values embedded in the models are used [1]. Even after the software cost models are calibrated to DoD databases, most have been shown to be accurate to within only 25 percent of actual cost or schedule about half the time. For example, Robert Thibodeau [2] reported accuracy of early versions of the PRICE-S and SLIM models to be within 25 and 30 percent, respectively, on military ground programs. The IIT Research Institute [3] reported similar results on eight Ada programs, with the most accurate model at only 30 percent of actual cost or schedule, 62 percent of the time.

Furthermore, the level of accuracy reported by these studies is likely overstated because most studies have failed to use holdout samples to validate the calibrated models. Instead of reserving a sample of the database for validation, the same data used to calibrate the models were used to assess accuracy [4]. In a study using 28 military ground software data points, Gerald Ourada [5] showed that failure to use a holdout sample overstates a model's accuracy. Half of the data was used to calibrate the Air Force's REVIC model. The remaining half was used to validate the calibrated model. REVIC was accurate to within 30 percent, 57 percent of the time on the calibration subset, but only 28 percent of the time on the validation subset.

Validating on a holdout sample is clearly more relevant because new programs being estimated are, by definition, not in the calibration database. The purpose of this project was to calibrate and properly evaluate the accuracy of selected software cost estimation models using holdout samples. The expectation is that calibration improves the estimating accuracy of a model [6].

The Decalogue Project

This paper describes the results of a long-term project at the Air Force Institute of Technology to calibrate and validate selected software cost estimation models. Two Air Force product centers provided software databases: the Space and Missile Systems Center (SMC), and the Electronic Systems Center (ESC). The project has been nicknamed the "Decalogue project" because 10 masters' theses extensively document the procedures and results of calibrating each software cost estimation model.

The Decalogue project is organized into three phases, corresponding to when the theses were completed. Five theses were completed in 1995; two theses were completed in 1996; and three theses were completed in 1997. Lessons learned during each phase were applied to the next phase. A brief description of each phase and its results follows.

Phase 1

Each student calibrated a specific software cost model using the SMC software database. The models were the Revised Enhanced Intermediate Version of the Constructive Cost Model (COCOMO) (REVIC), the Software Architecture Sizing and Estimating Tool (SASET), PRICE-S, SEER-SEM, and SLIM. The government owns REVIC and SASET. The other models are privately owned.

Management Consulting and Research developed the SMC database, and it contains detailed historical data for more than 2,500 software programs. The database includes inputs for REVIC, SASET, PRICE-S, and SEER-SEM for some of the 2,500 projects, but none specifically for SLIM.

The details of each thesis project are described in the separate thesis reports [7, 8, 9, 10, 11]. Each is available from the Defense Technical Information Center. Additional detail is also available from the authors of this article [12, 13]. Here, only the highlights of the results of the five studies are provided.

Calibration rules. The five models were calibrated to a portion of the SMC database. The database was divided into subsets: military ground, avionics, unmanned space, missiles, and military mobile. The military ground subset was further divided into command and control programs and signal processing programs. Each subset was divided into calibration and holdout samples using three rules:

1. If there were less than nine data points, the subset was considered too small for a holdout sample and could not be validated.
2. If there were between nine and 11 data points, eight were randomly selected for calibration and the rest were used for validation.
3. If there were 12 or more data points, two-thirds were randomly selected for calibration and the rest were used for validation.

The accuracy of each model was evaluated using criteria proposed by Samuel Conte, et al. [14] based on the following statistics:

(1) Magnitude of Relative Error (MRE) = $|\text{Estimate} - \text{Actual}| / \text{Actual}$

(2) Mean Magnitude of Relative Error (MMRE) = (MRE) / n

(3) Root Mean Square (RMS) = $[(1/n)(\text{Estimate} - \text{Actual})^2]^{1/2}$

(4) Relative Root Mean Square (RRMS) = RMS / [(Actual) / n]

(5) Prediction Level (Pred (.25)) = k/n

For Equation No. 5, n is the number of data points in the subset and k is the number of data points with MRE # 0.25. According to Conte, et al. [14], a model's estimate is accurate when MMRE < 0.25, RRMS < 0.25, and Pred (.25) > .75.

Results. Table 1 summarizes the results of Phase 1. Due to an oversight, not all five theses reported RRMS. Thus, only MMRE and PRED (.25) are shown. Validation sample size is the number of data points in the holdout sample used for validation. For some models, the military ground subsets (signal processing and command and control) were combined into an overall military ground subset to obtain a sufficiently large sample size for validation.

As shown in Table 1, most of the calibrated models were inaccurate. In the two instances where the calibrated models met Conte's criteria, only one data point was used for validation. Thus, these results are not compelling evidence that calibration improves accuracy. In some cases the calibrated model was less accurate than the model before calibration.

These results may be due in part to the nature of the databases available to DoD agencies. In the SMC database, the developing contractors are not identified. Therefore, the data may represent an amalgamation of many different development processes, programming styles, etc., which are consistent within contracting organizations, but vary widely across contractors. Also, because of inconsistencies in software data collection among different DoD efforts, actual cost data and other data may be inconsistent and unreliable.[1]

Phase 2

In 1996 two additional models, SoftCost-OO and CHECKPOINT, were calibrated by two master's students. CHECKPOINT is unique among the models calibrated in this study because the internal algorithms are based on function points instead of lines of code.[2] Details are provided in their thesis reports [15,16]. A brief description of each model, the calibration procedures, and the results of Phase 2 follow.

Calibration rules. With a few exceptions related to the subsets to calibrate and the holdout sample rules, the two models were calibrated and validated using the same methods that were used in Phase 1. A seventh subset of the SMC database, ground in-support-of-space ("Ground Support" in Tables 2 and 3) was used for both models. For SoftCost-OO, three additional subsets for European Space Agency programs were added, since SoftCost-OO is used extensively in Europe.

Table 1 **REVIC, SASET, PRICE-S, SEER-SEM, AND SLIM CALIBRATION RESULTS (1995)**

Model	Data Set	Validation Sample Size	Pre-Calibration MMRE	Pre-Calibration PRED (.25)	Post-Calibration MMRE	Post-Calibration PRED (.25)
REVIC	Military Ground	5	1.21	0	0.86	0
	Unmanned Space	4	0.43	0.50	0.31	0.50
SASET	Avionics	1	1.76	0	0.22*	1.00*
	Military Ground	24	10.04	0	0.58	0
PRICE-S	Military Ground	11	0.30	0.36	0.29	0.36
	Unmanned Space	4	0.34	0.50	0.34	0.50
SEER-SEM	Avionics	1	0.46	0	0.24*	1.00*
	Command and Control	7	0.31	0.43	0.31	0.29
	Signal Processing	7	1.54	0.29	2.10	0.43
	Military Mobile	4	0.39	0.25	0.46	0.25
SLIM	Command and Control	3	0.62	0	0.67	0

* Met Conte's criteria

Table 2 **SOFTCOST CALIBRATION RESULTS (1996)**

Data Set	Validation Sample Size	Pre-Calibration MMRE	Pre-Calibration RRMS	Pre-Calibration PRED (.25)	Post-Calibration MMRE	Post-Calibration RRMS	Post-Calibration PRED (.25)
Ground Support	15	2.73	3.13	0.13	1.80	1.96	0.20
Ground Support (Europe)	25	3.05	3.61	0.08	0.67	0.84	0.36
Unmanned Space	5	0.56	1.05	0.20	0.48	0.92	0.20
Unmanned Space (Europe)	7	1.79	0.79	0.14	1.27	0.84	0.14
Avionics	5	0.71	0.76	0.20	0.85	0.56	0.20
Command and Control	6	1.90	3.43	0.17	0.52	0.87	0.50
Signal Processing	9	0.43	0.61	0.11	0.28	0.64	0.44
Military Mobile	5	0.63	0.51	0.20	0.42	0.40	0.20

Table 3 **CHECKPOINT CALIBRATION RESULTS (EFFORT, 1996)**

Data Set	Validation Sample Size	Pre-Calibration MMRE	Pre-Calibration RRMS	Pre-Calibration PRED (.25)	Post-Calibration MMRE	Post-Calibration RRMS	Post-Calibration PRED (.25)
Effort – Function Points							
MIS – COBOL	6	0.54	0.10	0.67	0.02*	0.01*	1.00*
Military Mobile - Ada	4	1.38	0.41	0.25	0.19*	0.06*	0.75*
Avionics	4	0.82	0.68	0.50	0.16*	0.11*	0.75*
Effort – SLOC							
Command and Control	6	0.19*	0.14*	0.50	0.16*	0.16*	0.50
Signal Processing	10	0.09*	0.08*	1.00*	0.09*	0.08*	1.00*
Unmanned Space	5	0.05*	0.05*	1.00*	0.04*	0.06*	1.00*
Ground Support	4	0.05*	0.06*	1.00*	0.05*	0.06*	1.00*
COBOL Programs	4	0.05*	0.05*	1.00*	0.05*	0.05*	1.00*

* Met Conte's Criteria

Table 4 **CHECKPOINT CALIBRATION RESULTS (1997)**

Data Set	Validation Sample Size	Pre-Calibration MMRE	Pre-Calibration RRMS	Pre-Calibration PRED (.25)	Post-Calibration MMRE	Post-Calibration RRMS	Post-Calibration PRED (.25)
Ada Language	8	1.21	1.34	0.00	1.70	2.54	0.50
Assembly Language	11	0.83	1.44	0.09	2.05	1.20	0.18
FORTRAN Language	12	0.73	1.12	0.17	0.70	2.31	0.17
JOVIAL Language	7	0.71	1.22	0.00	0.44	0.68	0.43
Contractor B	4**	0.60	0.74	0.13	0.64	0.49	0.25
Contractor J	11	0.69	0.91	0.18	1.33	1.43	0.18
Ada and Contractor R	5**	0.59	0.57	0.05	0.39	0.72	0.45
CMS2 and Contractor M	5**	0.91	1.13	0.00	0.69	0.64	0.10
FORTRAN and Contractor A	7	0.82	0.84	0.00	0.44	0.88	0.29
JOVIAL and Contractor J	6	0.80	1.42	0.00	0.37	0.70	0.33

** Resampling Used for This Set

Table 5 **COCOMO II CALIBRATION RESULTS (1997)**

Data Set	Total Sample Size	Pre-Calibration MMRE	Pre-Calibration RRMS	Pre-Calibration PRED (.25)	Post-Calibration MMRE	Post-Calibration RRMS	Post-Calibration PRED (.25)
Command and Control	12	0.39	0.49	0.30	0.33	0.53	0.40
Signal Processing	19	0.45	0.63	0.33	0.38	0.53	0.40
Ground Support	15	0.71	1.16	0.07	0.66	0.95	0.20
Military Mobile	12	0.79	0.95	0.10	0.68	0.74	0.00

Table 6 **SAGE CALIBRATION RESULTS (1997)**

Data Set	Total Sample Size	Pre-Calibration MMRE	Pre-Calibration RRMS	Pre-Calibration PRED (.25)	Post-Calibration MMRE	Post-Calibration RRMS	Post-Calibration PRED (.25)
SMC – Avionics	9	0.45	0.54	0.21	0.39	0.52	0.24
Command and Control	10	0.23*	0.23*	0.70	0.29	0.30	0.45
Signal Processing	16	0.39	0.43	0.44	0.50	0.54	0.20
Unmanned Space	7	0.66	0.69	0.14	0.59	0.88	0.30
Ground Support	14	0.32	0.44	0.43	0.32	0.44	0.43
Military Mobile	10	0.37	0.47	0.29	0.41	0.52	0.36
Missile	4	0.66	0.89	0.00	0.67	0.44	0.24
ESC – Contractor A	17	0.48	0.57	0.17	0.41	0.40	0.31
Contractor J	17	0.37	0.47	0.33	0.47	0.57	0.14
Contractor R	6	0.32	0.36	0.32	0.21*	0.23*	0.54

* Met Conte's Criteria

For CHECKPOINT, the missile subset was not used, and no European programs were used. In addition, data were obtained on Management Information System programs written in Common Business-Oriented Language (COBOL) from a local contractor, and a subset for COBOL programs was added to determine if stratification by language would provide better results. Finally, the rules to determine the sizes of the calibration and holdout samples were changed to avoid the problem of single-point validations experienced in Phase 1. If there were eight or more data points in a subset, half were used for calibration, and the other half for validation. If there were fewer than eight data points, that subset was not used.

Results. Table 2 and Table 3 show the results of calibrating each model for development effort. For SoftCost-00 (Table 2), calibration almost always improved the accuracy of the model, although none of the subsets met Conte's criteria. For CHECKPOINT, all but one subset met the criteria when predicting development effort (Table 3).

Since CHECKPOINT uses function points as a measure of size, they were used when sufficient data points were available for the subsets; otherwise, source lines of code (SLOC) were used. For three function point effort subsets, there was substantial improvement in accuracy after the model was calibrated for other programs in these subsets, especially for the Management Information System COBOL subset. Except for the Command and Control subset, the SLOC effort subsets met Conte's criteria before and after calibration. Although calibration did not significantly improve accuracy for these subsets (primarily because SLOC are an output, not an input, to CHECKPOINT), the accuracy was good even without calibration. CHECKPOINT results for effort estimation are especially noteworthy as inputs for this model were not considered when the SMC database was developed.

Although these results were promising, it should not be assumed that CHECKPOINT will do as well in other environments. The best results for the CHECKPOINT model were for the Management Information System COBOL data set, which was obtained from a single contractor. Data from multiple contractors, which often characterize DoD databases, are more difficult to calibrate accurately. Furthermore, CHECKPOINT is a function point model. If the user wants to input size in SLOC, which is usually the case, the user or model must first convert the SLOC to function points. Unfortunately, the conversion ratios are sometimes subject to significant variations. Thus, the SLOC effort results for CHECKPOINT may not work out as well elsewhere.

Phase 3

In 1997 three models (COCOMO II, SAGE, and CHECKPOINT) were calibrated. COCOMO II, the successor to Boehm's COCOMO model [1], was calibrated to the SMC database. The SAGE model, a commercial model developed by Randy Jensen, was calibrated to the SMC and ESC databases. Finally, CHECKPOINT was calibrated to the ESC database to determine whether the unusually high accuracy reported by Karen Mertes [15] could be achieved on a different database. As before, the details are documented in the 1997 thesis reports [17,18,19]. Only the highlights are described here.

The ESC database contains information on 52 projects and 312 computer software configuration items [18], and contractor identifiers and language, but does not contain information on application type. It also contains inputs for the SEER-SEM model for which it was originally developed. The ESC database was initially stratified by a contractor as it was thought that a model could be more accurate when calibrated for a specific developer [6]. For CHECKPOINT, the ESC database was also stratified by language, and by contractor and language.

Calibration rules. The techniques used to calibrate the models were significantly improved over those used in the earlier phases. In the past, small data sets reduced the meaningfulness of the calibration. Making statistically valid inferences from small data sets of completed software projects is a common limitation of any calibration study. To overcome this limitation, each model was calibrated multiple times by drawing random samples from the data set. The remaining holdout samples were used for validation. Averages of the validation results became the measure of accuracy. This "resampling" technique is becoming an increasingly popular and acceptable substitute for more conventional statistical techniques [20].

The resampling technique is flexible. For CHECKPOINT, resampling was used on only the small data sets (eight to 12 data points). Four random samples from the small sets were used to calibrate and validate the model. For COCOMO II, only data sets of 12 or more data points were used, and resampling was accomplished on all sets by using 80 percent of the randomly selected points for calibration, and the remaining 20 percent for validation. The process was repeated five times, and the results were averaged. For the SAGE model, all data sets having four or more points were used with an even more comprehensive resampling procedure. Simulation software, Crystal Ball, was used to select two data points for validation, and the rest for calibration. Instead of limiting the number of runs to four or five, all possible subsets were run.

Results. Table 4 shows the results of the CHECKPOINT calibration using the ESC database. Unlike the results reported by Mertes [15], none of the data sets met any of Conte's criteria, even those for a single contractor. This may be due in part to the lack of function point counts in the ESC database; only SLOC are provided for all data points. However, since Mertes' results using CHECKPOINT for SLOC were also good, it is difficult to account for differences between the results of Mertes [15] and Thomas Shrum [19].

Table 5 shows the results for COCOMO II, where calibration slightly improved the model's predictive accuracy, but none of the subsets met Conte's criteria. It is possible that better results may be attained when the online calibration capability is incorporated into the model.

Table 6 shows the results for SAGE on both databases. Although calibration sometimes resulted in improved accuracy, only a few sets met Conte's criteria. This is somewhat surprising for the ESC data sets, where individual contractors are identified by a code letter, and the model should be consistent for a company. It may be that even within a single company software programs are developed differently. Also, it is possible that if the simultaneous effort and schedule calibration capability that is now integrated into SAGE was used, the results would be better.

Conclusion

Calibration does not always improve a model's predictive accuracy. Most of the calibrated models evaluated in this project failed to meet Conte's criteria. According to Mertes, the one exception was the calibration of CHECKPOINT to the SMC database, where almost all of the calibrated data sets met Conte's criteria for function point and SLOC applications. Unfortunately, Shrum could not replicate this result on the ESC database using a superior validation technique. Overall, none of the models was shown to be more accurate than within 25 percent of actual cost or effort, one half of the time.

This does not mean the Decalogue project was done in vain. Much was learned about the models, their strengths and weaknesses, and the challenges in calibrating them to DoD databases. One major insight is that the use of a holdout sample is essential for meaningful model calibration. Without a holdout sample, the predictive accuracy of the model is probably overstated. Since all new projects are outside of the historical database(s), validation is much more meaningful than the more common practice of analyzing within-database performance. The calibrations performed in 1997 also developed and applied resampling as a superior technique to use in validating small samples. It is better than just using one subset of data for a holdout, and can be done easily with modern software, such as Excel and Crystal Ball. Hopefully, the findings of the Decalogue project will inspire additional effort in the area of model calibration, and more promising results will be obtained.

References

1. Boehm, Barry W. *Software Engineering Economics.* Englewood Cliffs, N.J., Prentice-Hall (1981).
2. Thibodeau, Robert. *An Evaluation of Software Cost Estimating Models.* Huntsville, Ala., General Research Corp. (1981).
3. IITRI. *Test Case Study: Estimating the Cost of Ada Software Development.* Lanham, Md., IIT Research Institute (1989).
4. Ourada, Gerald L. and Daniel V. Ferens. Software Cost Estimating Models: A Calibration, Evaluation, and Comparison. *Cost Estimating and Analysis: Balancing Technology and Declining Budgets.* N.Y., Springer-Verlag, pp. 83-101 (1992).
5 Ourada, Gerald L. Software Cost Estimating Models: A Calibration, Evaluation, and Comparison. Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1991).
6. Kemerer, Chris F. An Empirical Validation of Software Cost Estimation Models. *Communications of the ACM*, pp. 416-429 (May 1997).
7. Galonsky, James C. *Calibration of the PRICE-S Model.* Master's thesis. Dayton, Ohio: Air Force Institute of Technology (1995).
8. Kressin, Robert K. *Calibration of the Software Life Cycle Model (SLIM) to the Space and Missile Systems Center Software Database (SWDB).* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1995).
9. Rathmann, Kolin D. *Calibration of the System Evaluation and Estimation of Resources Software Estimation Model (SEER-SEM) for the Space and Missile Systems Center.* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1995).
10. Vegas, Carl D. *Calibration of the Software Architecture Sizing and Estimation Tool (SASET).* Master's thesis. Dayton, Ohio,: Air Force Institute of Technology (1995).
11. Weber, Betty G. *A Calibration of the REVIC Software Cost Estimating Model.* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1995).
12. Ferens, Daniel V., and David S. Christensen. Software Cost Model Calibration, An Air Force Case Study. *Journal of Cost Analysis and Management,* pp. 43-46 (Fall 1997).
13. Ferens, Daniel V., and David S. Christensen. Calibrating Software Cost Models to DOD Databases—A Review of 10 Studies. *Journal of Parametrics* 14: 33-52 (Spring 1999).
14. Conte, S.D., Dunsmore, H.E., and Shen, V.Y. *Software Engineering Metrics and Models.* Menlo Park, Calif.: Benjamin-Cummings (1986).
15. Mertes, Karen R. *Calibration of the CHECKPOINT Model to the Space and Missile Systems Center (SMC) Software Database (SWDB).* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1996).
16. Southwell, Steven V. *Calibration of the SoftCost-R Software Cost Model to the Space and Missile Systems Center (SMC) Software Database (SWDB).* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1996).
17. Bernheisel, Wayne A. *Calibration and Validation of the COCOMO II Cost/ Schedule Estimating Model to the Space and Missile Systems Center Database.* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1977)
18. Marzo, David B. *Calibration and Validation of the SAGE Cost/Schedule Estimating System to United States Air Force Databases.* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1997).
19. Shrum, Thomas C. *Calibration and Validation of the CHECKPOINT Model to the Air Force Electronic Systems Center Software Database.* Master's thesis. Dayton, Ohio, Air Force Institute of Technology (1997).
20. University of Maryland. *The Resampling Project.* College Park, Md. (1997).
21. Albrecht, A.J. and J.E. Gaffney. Software Function, Source Lines of Code, and Development Effort Production: A Software Science Validation. *IEEE Transactions on Software Engineering.* Volume SE-9 (November 1983).

Notes

1. This problem was addressed in Phase 3 of the Decalogue project, where the ESC database was used. The ESC database contains an identifier for each contributing contractor.
2. Function points are weighted sums of five attributes or functions of a software program (inputs, outputs, inquiries, interfaces, and master files). Based on their analysis of more than 30 data processing programs, Allan J. Albrecht and John Gaffney[21] report that function points may be superior to SLOC as predictors of software development cost or effort.

See p. 24 for author information.

Conclusion

Human biases influence and generally have a negative impact on the development of task-level estimates. Although it is impossible to obviate these biases, awareness, understanding, and the incorporation of bias-reduction strategies can help mitigate their negative impact.

We have taken a step back to discuss what we feel to be the root cause of poor task-level estimates using the expert judgment approach during bottom-up estimating. The expert judgment method is viable, and likely to remain one of the most popular methods of developing software project estimates for some time. The next step will be determining to what extent these biases impact software project estimates, and where information technology project managers should focus their efforts to reduce the negative consequences of bias in the software estimating process. Our hope is that the suggestions we have provided here can help you and your team develop better task-level software project estimates.

References

1. Godson, Philip. To Err is Human, to Estimate Divine. *InformationWeek,* January 18 ,1999, pp. 65-72.
2. Kahneman, Daniel, and Tversky, Amos The Framing of Decisions and the Psychology of Choice. *Science,* Vol. 211 No. 30, January 1981, pp. 453-458.
3. Hughes, Robert T. Expert Judgement as an Estimating Method. *Information and Software Technology,* Vol. 38, 1996 pp. 67-75.
4. Jones, Morgan D. *The Thinker's Toolkit,* 1998, Random House, Inc.
5. Simon, Hervert A. *Psychology Review,* Vol. 63, p. 129, 1956.
6 Lim, Lai-Huat., and Benbasat, Izak. A Framework for Addressing Group Judgment Biases with Group Technology. *Journal of Management Information Systems,* Vol. 13, No. 3, Winter 96-97, pp. 7-24.
7. Bazerman, Max. H. *Managerial Decision Making,* 2nd Ed., 1990. John Wiley & Sons, Inc.
8. Hammond, John. S. Keeney, Ralph. L., Raiffa, Howard. The Hidden Traps in Decision Making. *Harvard Business Review,* Sept.-Oct. 1998, pp. 47-58.
9. Demarco, Tom *Controlling Software Projects,* Yourdon Press Computing Series, 1982.

About the Authors

David Peeters is a project manager in the Information Services Division at American Family Insurance in Madison, Wis. He has more than 10 years experience in application development, project planning, and project management. Peeters has a bachelor's degree in computer information systems from Southwest State University and a master's of business administration from Illinois State University.

David Peeters
American Family Insurance
6000 American Parkway
Madison, Wis. 53783
Voice: 608-242-4100 ext. 31348
Fax: 608-243-6558
E-mail: DPeeters@AmFam.com

George Dewey is President of Pathfinder Global Group Inc. He is a certified Project Management Professional, with more than 20 years of experience in corporate planning, project scheduling, cost control, estimating, and management with numerous consulting firms and client organizations. Dewey holds a bachelor's degree in industrial engineering from North Dakota State University and a master's of business administration from Virginia Polytechnic Institute.

George Dewey
Pathfinder Global Group Inc.
5358B N. Lovers Lane #113
Milwaukee, Wis. 5225
Voice: 713-827-4481
Fax: 414-464-3005
E-mail: GDewey_PGGI@MSN.com

About the Authors of Does Calibration Improve Predictive Accuracy?, continued from page 17

Daniel V. Ferens is a corporate affordability officer at the Air Force Research Laboratory at Wright-Patterson AFB in Dayton, Ohio. He is also an adjunct associate professor of software systems management at the Air Force Institute of Technology (AFIT), Graduate School of Logistics and Acquisition Management, at Wright-Patterson AFB, where he teaches courses in software estimation and software management in general. He was the advisor for the 10 theses described in this paper, is an active member of the Society of Cost Estimating and Analysis, and a lifetime member of the International Society of Parametric Analysts. Ferens has a master's degree in electrical engineering from Rensselaer Polytechnic Institute, and a master's degree in business from the University of Northern Colorado.

AFRL/IFSD, Bldg 620
2241 Avionics Circle, Room S1Z19
Wright-Patterson AFB, Ohio 45433-7334
Voice: 937-255-4429, ext. 3558
FAX 937-255-4511
E-mail: daniel.ferens@sn.wpafb.af.mil

David S. Christensen is an associate professor of accounting at Southern Utah University. He was the reader for the 10 theses described in this paper. He received his doctorate degree in accounting from the University of Nebraska-Lincoln in 1987, and has published more than 50 articles in the area of defense cost management. He taught cost management topics at the Air Force Institute of Technology from 1987-97. He is a member of the American Accounting Association, the Institute of Management Accountants, and the Society of Cost Estimating and Analysis.

Southern Utah University
College of Business
351 West Center St.
Cedar City, Utah 84720
Voice: 435-865-8058
Fax: 435-586-5493
E-mail: ChristensenD@suu.edu