

# Software Tech News

Volume 1 Number 2

## **Topic In This Issue :** **Software Measurement**

Earned Value and Automated Information Systems .....	1
Software Engineering Institute Measurement Initiative .....	2
Practical Software Measurement Conference .....	4
Analyzing Quantitative Data Via the Web .....	11
Conference Announcement: DACS at STC '98 .....	14
Software Technology Calendar of Events .....	15
DACS Product & Services Orderform .....	insert

## **Earned Value (EV) and Automated Information Systems (AIS)**

*by Fred Manzer, Defense Systems Management College*

One of the most common questions I receive today goes something like, "Does earned value management apply to an automated information system effort?" This question reflects DoD history relative to both earned value management and AIS. Historically, earned value (under the guise of CS/CSC) was commonly perceived as a required reporting process oriented toward major acquisition programs. This caused it to become a burden and added cost with little value to the management. Although this was never the intent, the perception became reality and earned value came into disfavor.

AIS efforts often consisted of "software maintenance" planned and managed as a level of effort. Annual budgets funded teams working on system problems, upgrades and sometimes completely new applications. In this environment the development team generated as much product as possible for the money available that year and moved any remaining work to the next year. Additionally, the customer often had poorly defined or constantly changing requirements. Without specific and controlled product definition there could not be a product, or project, management mentality.

*Continued on page 12*



<http://www.dacs.dtic.mil>

**Your Source for Information in  
Software Engineering Technology.**

*Formerly the DACS Newsletter*

## The Software Engineering Measurement and Analysis Initiative at the SEI

by David Zubrow, Software Engineering Institute (SEI)

Software measurement lies at the heart of the answer to questions about project control, organizational performance, and return on investment. Do you or your organization have questions like the following:

- How well are we meeting schedules and budgets?
- Has our performance really improved?
- What software practices and/or technologies should our organization invest in and what yields can we expect from this investment?
- How does my organization's performance compare to other organizations' performance?

Without measurement, none of these questions can be credibly answered. Furthermore, those who want to measure their performance have questions as well, such as:

- What is the value of a software measurement program?
- How do I get started building a comprehensive measurement program?
- How should a measurement program connect to other organizational improvement activities?
- How should my measurement program relate to what is described in the Software CMM® (Capability Maturity Model<sup>SM</sup>)?
- How are higher maturity organizations addressing their measurement issues?

Today, the Software Engineering Measurement and Analysis (SEMA) Initiative at the Software Engineering Institute helps organizations to better manage their projects, to understand their own capabilities and performance, and to measure and document the results of innovations promising improvement in software development and maintenance. The new mission of the SEMA group includes two broad goals:

- To develop and aid the transition of software measurement and analysis products and services that can be used by software organizations to improve their processes and manage their efforts
- To develop information resources that address important issues and needs confronting the software community

To accomplish our mission we produce technical reports, guide books, and training related to software measurement. We also collaborate with organizations that wish to establish or improve their measurement activities, investigate the use of innovative analytical techniques on their data, and want to document the effects of process and technology innovations on their software engineering performance.

### Products and Services for Software Measurement

Software measurement products and services address how to determine what to measure, how to define the measures completely and unambiguously, and how to identify some of the dangers and pitfalls commonly associated with implementing measurement in software organizations. At the core of these products is the *Goal-Driven Software Measurement Guidebook* (Park, Goethert, and Florac, 1996). The document details a 10-step method of aligning measurement activities with goals in an organization. This method attempts to assure that the data collected will be used to make decisions and guide action. The SEI also provides corresponding training in a workshop and public course. In the workshop and course we focus on helping attendees identify explicit indicators and use analyses that rest upon a foundation of clearly defined software measures. To arrive at these clear definitions, we utilize software measurement definition frameworks. The frameworks provide a means for explicitly describing the entities and attributes of a software measure. These frameworks are documented in a series of SEI

Continued on page 3

## SEI SEMA Initiative

*Continued from page 2*

technical reports covering size (Park et al, 1992), effort and schedule (Goethert, Bailey, and Busby, 1992), and defects (Florac et al, 1992). In addition, general guidance for implementing measurement activities within an organization is published in *Establishing a Software Measurement Process* (McAndrews, 1993).

While the above products are of most help to organizations who are just getting measurement in place, the latest work on measurement for process management and improvement (Florac, Park, and Carleton, 1997) provides guidance to higher maturity organizations seeking to understand how statistical process control might benefit them. This report reviews the principles underlying the analytical techniques used in statistical process control and shows how these principles can be applied to software processes.

To bridge the gap between the existing support for lower maturity organizations and those that are more mature, the SEI is working to develop guidance regarding the evolution of measurement in the context of CMM®-based software process improvement. The focus of this work is on how measurement

can accelerate and serve as a pulling function for improvement by providing rapid and specific feedback to projects and the organization. This is the same principle that is applied in the Personal Software Process<sup>SM</sup> and the Team Software Process<sup>SM</sup>. While much material for this work already exists, the synthesis of this material into an evolutionary path integrated with the CMM-based software process improvement (SPI) has not, to our knowledge, been done. The specific evolutionary themes we've identified can be found in a presentation at the 1997 Software Engineering Symposium (Zubrow, 1997) and is available from our web site at <http://www.sei.cmu.edu>.

Finally, we have established a data analysis testbed, or laboratory, where we investigate the application of statistical and other analytical techniques to software engineering problems. The application of quantitative techniques to software management and development is still relatively immature. We are collaborating with software organizations to explore various techniques for analysis including time series analysis, statistical process control, and lexical analysis of text.

Through these efforts we are working to improve the organic capability of software organizations to apply software measurement. Note, however, that it is not measurement for its own sake, but rather for the insight and support that it can provide for making decisions at both the project and organizational levels.

### Information Resources for the Software Engineering Community

Many organizations also seek information regarding the experiences of others to assist them in deciding to adopt new technologies or to embark on process improvement initiatives. They ask questions such as the following:

- What software practices or technologies should our company invest in and what yields can we expect from this investment?
- What are the potential risks facing a project and how can they be mitigated?
- What practices have led to reductions in post-release defects? And by how much?
- How does my organization meet its business goals compared to other organizations?

## Practical Software Measurement (PSM) 1997 Users' Group Conference

review by Robert Vienneau and Jack McGarry, Naval Undersea Warfare Center

The first annual PSM Users' Group Conference was held on 21-24 July 1997 in Vail, Colorado. After a morning tutorial and a welcome by Ms. Cheryl Jones of the Naval Undersea Warfare Center (NUWC), Mr. John (Jack) McGarry (NUWC) provided a PSM Project Update.

PSM is a software measurement approach which began by emphasizing program management. PSM includes an Issues/Measurement/Categories hierarchy and a process that provides flexibility for tailoring and integration into the developer's software process. PSM began in parallel with the development of MIL-STD-498, but has expanded in scope to include non-DoD government and commercial systems. PSM is currently sponsored by the Joint Logistics Commanders Joint Group on Systems Engineering and the Office of the Under Secretary of Defense for Acquisition and Technology. PSM was developed with an open process and the participation of a wide range of organizations, including other DoD software measurement programs including the U. S. Army Software Test and Evaluation Panel (STEP) Metrics Program, the Software Engineering Institute (SEI) Core

Measures Program, and U. S. Air Force programs. All members of the PSM project team have an equal voice, but Mr. McGarry stated the adoption of PSM by some organizations, such as the Federal Aviation Authority, Lockheed-Martin, and an ongoing International Standards Organization (ISO) standard development, increases PSM credibility. PSM products and services consist of:

- PSM: A Guide to Objective Program Insight, Version 2.1, (March 27, 1996)
- Training and support
- Insight, a beta version of a software tool for a measurement analyst's workstation.

A related product is the SEI report, *Practical Software Measurement: Measuring for Management and Process Improvement* by Bill Florack, Robert Park and Anita Carleton. The draft Version 3.0 of PSM will be issued by October. A systems engineering PSM effort has just begun, and product engineering issues will be included in Version 4.0.

Dr. George Stark, of MITRE, presented *Return on Investment from a Software Measurement Program*. In Fiscal Year 1994, MITRE began supporting the maintenance of a Missile

Warning and Space Surveillance System (MWSSS) fielded at 8 sites and consisting of 12 million lines of code. The maintenance process was reorganized from several level-of-effort programs to a single organization tasked by release. PSM was used to introduce measures addressing issues of cost, schedule, requirements volatility, product quality, and workload. Dr. Stark presented data showing the cost of the measurement program was \$146 K over the first two years. He also stated the average cost per Standard Change Form was lowered by \$21 K between FY1995 and FY1996. The Return on Investment (ROI) of the software measurement program, based on this data, is 187%. Dr. Stark described two decisions that were made based on measurement results and avoided a total cost of \$610 K. The ROI of the measurement program based on this cost avoidance data was \$418 K. Dr. Stark agreed with a questioner that ROI was not the most important aspect of a successful measurement program. ROI attracts attention and is needed to justify high-level management support. In his application, objective communication, increased

Continued on page 5

## PSM Conference

*Continued from page 4*

process understanding, and model-building to support answering questions were just as important.

Major Thomas Neff of the U. S. Air Force, presented “Implementing PSM at USSTRATCOM”. Major Neff is the Chief of Metrics for the Software Support Division. The objective of his talk was to suggest a method to implement the PSM in one’s organization in about six months. He contrasted the situation at USSTRATCOM last year and now. Last year management was by guess and feelings with no consistency between branches or projects. Today PSM guides the Software Support Division’s metrics program. Metrics are driven by business goals, and business goals are clearly prioritized. Major Neff recommends that one form a corporate-wide Metrics Management Advisory Group (MetMAG) to initiate PSM. One must sell senior management and mid-management to support metrics. Senior management should prioritize issues. MetMAG and senior management should develop a timetable for implementing metrics. Major Neff also discussed training.

Ms. Carol Dekkers, of Quality Plus, Incorporated, discussed

“What Functions Points Are and Are Not.” Ms. Dekkers is also a Vice President of the International Function Points Users Group (IFPUG). She explained that FPs are a measure of application or project size based on quantification of user requirements independent of technology, tools, and other physical project attributes. FPs are not equal to work effort, productivity, or the internal complexity of applications. Ms. Dekkers stated that FPs can be used in conjunction with other measures to produce software metrics such as Productivity, Delivery Rates, and Support Ratios.

Ms. Joan Weszka, of Lockheed, began Tuesday morning with the first keynote talk, “Measurement at Lockheed-Martin”. She discussed Lockheed-Martin’s roles in PSM transition and PSM influence on the Lockheed-Martin Corporation Software & Systems Resource Center (SSRC) Measurement Initiative. SSRC is a service organization and a source of expert resources to Lockheed-Martin companies performing software development and systems integration. The intent is to leverage PSM as a best practice on Lockheed-Martin programs. Lockheed-Martin has a number

of transition mechanisms, including an internally-developed guidebook, training, and recommended policy, processes, methods, tools, and technologies. The SSRC measurement initiative includes collaboration with key external groups, such as the SEI, PSM project, and the University of Southern California Center for Software Engineering. Ms. Weszka concluded by suggesting PSM products and activities, including:

- PSM involvement in Software Process Improvement Networks (SPINs) and maybe initiating a Measurement Improvement Network
- A PSM newsletter
- Lessons learned
- Guidance on retiring a measurement
- Greater support for analysis
- Expanding sample indicators
- Use of PSM for a team/risk management approach (as described by the SEI).

Dr. Robert Charette, of ITABHI Corporation, presented ‘Risk Management and PSM’. His talk included much tutorial information. For example, according to Dr. Charette, not all problems are risks. A risk only exists if there is some uncertainty and one has choices

*Continued on page 6*

## PSM Conference

*Continued from page 5*

that might mitigate it.

Dr. Charette summarized several process models, including a spiral model much like Dr. Barry Boehm's spiral life cycle model. He concluded by identifying opportunities for cooperation, coordination, and symbiosis between risk management and PSM.

Mr. Scott Lucero, of the U. S. Army [OPTEC](#) introduced Mr. Dave Morris of Independent Engineering Incorporated and the PSM Insight Demonstration. Insight was developed under OPTEC's sponsorship and was referred to as a measurement analyst's workbench. It runs under Windows and is very flexible. Mr. Morris demonstrated how to define issues, categories, measures, and data items. A major capability allows the user to import data in several formats. Mr. Morris concluded the demonstration by illustrating graphing capabilities. All conference attendees received a beta version of Insight.

Ms. Cheryl Jones introduced the Transition and Development Partners Panel. Development partners help develop PSM products, while transition partners implement the PSM. OPTEC, having developed Insight, is a PSM Development

Partner. Mr. Scott Lucero summarized OPTEC's history in software measurement, the Software Test and Evaluation Panel ([STEP](#)), and a unified OT&E process. OPTEC changed their metrics policy last year. Mr. Bruce Allgood represented ESIP as a development partner and the STSC as a transition partner. Ms. Tamara Chism of the FAA discussed process improvement, a relatively new initiative in the FAA. PSM was adopted because FAA managers resisted having metrics mandated and wanted the flexibility to tailor their own metrics. The FAA is a transition partner. Since Lockheed Martin was already discussed in a previous paper, Mr. David Card mentioned commercial best practices and the importance of the ISO PSM standard under development for lending credibility to commercial use of PSM. Audience discussion centered on whether commercial organizations can profitably sell PSM support, whether government organizations should charge for PSM products and services, key issues such as getting started, and the use of tools (such as Structured Analysis tools) in collecting data.

Two papers were presented on **Software Estimation Techniques**. Ms. Betsy Bailey,

of Software Metrics Inc. provided an overview of how PSM issues relate to software estimation. Size is the single most important input, but size measures will vary among developers. Estimates are essentially meaningless in the absence of historical data. Estimates are updated throughout a project. The estimator should make all assumptions and inputs visible. Mr. Douglas Putnam stepped the audience through a recent case study of a billing system. The system being estimated was being developed by a commercial client with good functional domain experience, but no experience with the language and methodology, Smalltalk and Object Oriented development. Mr. Putnam presented the sizing estimates in some detail. His original estimates included a risk analysis. The developers used a metrics "control panel" in managing their activities. The control panel was used to identify changes that required a re-estimate. The customer was fairly satisfied at the end of the development.

Dr. Arthur Pyster delivered the second keynote speech, "PSM and the FAA", at the Tuesday evening dinner. Dr. Pyster is the

*Continued on page 7*

## PSM Conference

*Continued from page 6*

Chief Scientist for Software Engineering at the Federal Aviation Agency. Dr. Pyster stated that safety is the top priority of the FAA and emphasized the scale of the FAA problem. The average American flies 2,000 miles per year, and 600 million people enplane per year. The Air Traffic Control system has four levels, and 50,000 people in the FAA operate 34,000 pieces of equipment. Dr. Pyster stated that process improvement at the FAA now has high-level management support, especially from Mr. George Donahue, the FAA Acquisition Executive. He emphasized that the FAA acquires systems, not software. Their Capability Maturity Model was developed by the FAA and integrates the SEI software engineering, software acquisition, and systems engineering CMMs. The FAA's goal is to increase the process maturity of 75% of selected major software-intensive systems to FAA CMM Level 2 by December 1999 and to Level 3 by December 2001. PSM is used in the FAA process improvement infrastructure, including the use of executive program metrics reported to Mr. Donahue.

The conference broke up into four working groups on Wednesday. Each group reported back to the floor Thursday morning. Mr. David Card and Mr. Jack McGarry moderated the PSM Guidance Planning & Feedback Group. This session reviewed the PSM Version 3.0 guidance and requirements. The session discussed integrating risk management into PSM, performance measurement, and maintenance. They developed process diagrams for the PSM Version 3.0 measurement process.

Mr. Dave Morris (IEI), Ms. Joyce Jakaitis (ASC), and Mr. Scott Lucero (OPTEC) hosted the PSM Insight Measurement Tool Beta Test Review and Feedback Group. Workshop participants presented comments on functionality, usability, scope of the tool, types of analyses needed, and technical recommendations. For example, although Insight currently is designed for analyzing data from single projects, participants wanted cross-project analyses. Requirements were identified for a number of successive releases. Security issues were heavily discussed.

Dr. James "Sean" Arthur of Virginia Polytechnic Institute, and Mr. Edward Dudash of

Naval Surface Warfare Center, hosted the Software Product Engineering Measurement Group. Their objectives were to provide requirements for the PSM team that will incorporate product measures into PSM Version 4.0. The group discussed "context" and measurement issues. The impact of the life cycle model on measurement is an example of a context issue.

Mr. Garry Roedler of Lockheed-Martin and Dr. William Farr of Naval Surface Warfare Center, moderated the Systems Engineering Measurement Group. The group's objectives were to identify initial user requirements for Practical System Measurement. The group compared and contrasted PSM and PSystem project objectives, scope, and concepts. PSystem will have the same look and feel as PSM. Some issues were identified for systems/program measurement that differ from software issues.

Major Thomas Neff briefly told us "How to get involved in a SPIN". A SPIN is a Software Process Improvement Network, and Dawna Baird [dbaird@sei.cmu.edu](mailto:dbaird@sei.cmu.edu) can provide information about joining a local SPIN.

*Continued on page 8*

## PSM Conference

*Continued from page 7*

Major Neff summarized some of the presentations provided by the Omaha SPIN. Mr. Perry DeWeese, of Lockheed Martin, discussed Software Standards. This discussion focused on the transition from MIL-STD-498 to commercial standards, namely EIA IS 640/IEEE 1498 (EIA/IEEE/J-STD-016 - Trial Use) for software development and IEEE/EIA 12207 (ISO/IEC 12207) for life cycle processes. Mr. DeWeese discussed Performance Based Business Environment (PBBE), as defined by a Joint Service Guide Specification. PBBE is a result of acquisition reform and became policy on 1 October 1997. Mr. DeWeese stated that PBBE can be implemented in a framework compatible with 12207. PSM fits well with PBBE.

Ms. Cheryl Jones of NUWC presented "Lessons Learned Adapting the PSM Process". She summarized PSM Support Center services including training, tailoring workshops, surveys, and Web sites. She listed over a dozen programs the PSMSC has recently supported. Programs included all services, the FAA, and even an Australian program. About 1300 people have received the half-day PSM

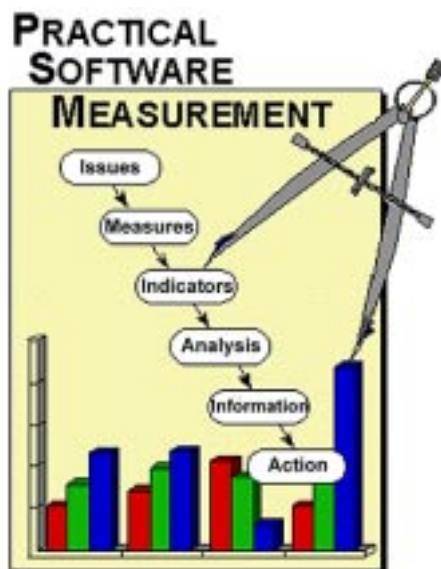
course. One hundred thirty three people have received the one-day PSM course. After having overviewed Support Center services, Ms. Jones presented a case study. She began the discussion of the case study with the tailoring approach. Issues consisted of development and integration progress, functional system performance, product characteristics and quality, and development and support resources. Lessons learned included the observations that:

- The program manager, staff, and the contractor should collaborate in identifying program issues
- Measurement data was available, although some issues were not addressed
- Tailoring is a learning process

- The priority of software issues changes
- Measurement is effective even without perfect or complete data
- Analysis feedback to management is essential.

Mr. Jack McGarry concluded the workshop and thanked us for attending. More information about PSM can be obtained from the PSM Support Center:

**Ms. Cheryl Jones**  
 PSM Support Center  
 Naval Undersea Warfare Center  
 Code 2252  
 1176 Howell Street  
 Newport, RI 02841-1708  
 Voice: (401) 841-4581  
 Fax: (401) 841-2130  
 E-mail:  
[psm@code22.npt.nuwc.navy.mil](mailto:psm@code22.npt.nuwc.navy.mil)



Visit the PSM Web Site at:  
<http://www.psmc.com>

## SEI SEMA Initiative

*Continued from page 3*

To begin to address this need for information about the performance of various technologies and practices, the SEI is developing the Software Engineering Information Repository (SEIR), available on the web at <http://seir.sei.cmu.edu/>. It is a web-based repository of information on software engineering practices that have been demonstrated to lead to improved organizational performance. It will contain tools allowing users to customize searches and analyses to provide specific answers to their questions. The SEIR is designed to serve the software engineering community by gathering, coordinating, analyzing, and disseminating data and information on the impact of software engineering practices leading to improvement.

Currently available in the repository is information on the CMM-based software process improvement, risk management, and the Personal Software Process. The data on CMM-based SPI is primarily derived from software process assessments. This data includes the results published in the Community Maturity Profile and analyses of assessment findings that are being performed now.

We plan to provide capabilities on-line for analysis of these data including drill-down analyses. We will be adding new areas as we develop the data and information for them.

While the SEIR provides a general means of disseminating data and information on software engineering practices, SEMA also supports other initiatives within the SEI. We provide our expertise in terms of conducting impact studies and other empirical research. We have done extensive work on the impacts of CMM-based software process improvement. This work has been documented in a series of technical reports and summarized most recently in an article in the *Communications of the ACM* (Herbsleb et al, 1997). Soon to be published is an analysis of PSP course data in which we analyze the changes experienced by individual course participants. These results were presented at the 1997 Software Engineering Symposium (Hayes, 1997). This approach to the analysis allowed us to use each individual's skills upon entering the course as a baseline. This analysis documents the improvements that participants realized in estimating accuracy, reduction in defects, and earlier removal of defects. Today, we

are just beginning a rigorous investigation of PSP in the field. And, we are working with others in the SEI to measure the performance impacts associated with the adoption of a product line approach to software development.

To keep our work aligned with the needs of the software engineering community, we seek collaborators willing to co-develop products, participate in our studies, and provide data and information. If you are interested in our work or would like to collaborate, please contact us through:

### SEI Customer Relations

[customer-relations@sei.cmu.edu](mailto:customer-relations@sei.cmu.edu)

or visit our website at:

<http://www.sei.cmu.edu/technology/measurement>.

### Notice

<sup>SM</sup> Capability Maturity Model, Personal Software Process, Team Software Process, and PSP are service marks of Carnegie Mellon University. CMM® is a registered in the U.S. Patent and Trademark Office.

## SEI Measurement Initiative

*Continued from page 9*

### References

Florac, William, et al., *Software Quality Measurement: A Framework for Counting Problems and Defects* (CMU/SEI 92-TR-22). Pittsburgh, PA.:

Software Engineering Institute, Carnegie Mellon University, September 1992.

Florac, William, Park, Robert, and Carleton, Anita, *Practical Software Measurement: Measuring for Process Management and Improvement* (CMU/SEI 97-HB-003). Pittsburgh, PA.:

Software Engineering Institute, Carnegie Mellon University, April 1997.

Goethert, Wolfhart, Bailey, Elizabeth, and Busby, Mary, *Software Effort and Schedule Measurement: A Framework for Counting Staff-Hours and Reporting Schedule Information* (CMU/SEI 92-TR-21).

Pittsburgh, PA.: Software Engineering Institute, Carnegie Mellon University, September 1992.

Hayes, Will, “Quantifying the effects of the PSP”, *Proceedings of the 1997 Software Engineering Symposium*, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, August 1997.

Herbsleb, James, Zubrow, David, Goldenson, Dennis, Hayes, Will, and Paulk, Mark “Software Quality and the Capability Maturity Model”, *Communications of the ACM*, 40, 6 (June, 1997): 30-40.

McAndrews, Donald, *Establishing a Software Measurement Process* (CMU/SEI 93-TR-16).

Pittsburgh, PA.: Software Engineering Institute, Carnegie Mellon University, July 1993.

Park, Robert, et al. *Software Size Measurement: A Framework for Counting Source Statements* (CMU/SEI 92-TR-20). Pittsburgh, PA.:

Software Engineering Institute, Carnegie Mellon University, September 1992.

Park, Robert, Goethert, Wolfhart, and Florac, William, *Goal-Driven Software Measurement—A Guidebook* (CMU/SEI 96-HB-002). Pittsburgh, PA.:

Software Engineering Institute, Carnegie Mellon University, July 1996.

Zubrow, David, “The Evolution of Measurement with CMM-Based Software Process Improvement”, *Proceedings of the 1997 Software Engineering Symposium*, Pittsburgh, PA:

Software Engineering Institute, Carnegie Mellon University, August 1997.



## Analyzing Quantitative Data Via the Web

Summary of the paper by Robert L. Vienneau and James D. DeLude

The World Wide Web (WWW) is one of the most exciting applications used today on the Internet. Created in 1992, the Web is an Internet resource discovery service supporting distributed hypertext documents. Major components of the Web include the Uniform Resource Locator (URL) addressing scheme, the Hypertext Transfer Protocol (HTTP), and the Hypertext Markup Language (HTML). The popularity of the Web is undoubtedly due to its support for multimedia and graphics, introduced in January 1993 with Mosaic, the first popular easy-to-use graphical Web browser. A recent survey found that of 24 million people aged 16 and above who had used the Internet in the last three months of summer 1995, three quarters of these Internet users used the Web.

The Internet is still evolving and future Web applications are likely to exhibit greater interactivity. Corporations are

beginning to find value in providing customers with up-to-date and customized information on products and services. Approximately 2.5 million Web users have already made purchases through the Web. The technology to provide the needed interactivity is beginning to emerge. Most notably, the programming language Java allows users to automatically download platform-independent applications ("applets") to run on the user's computer.

**The focus of this paper, however, is on server-side capabilities that provide the automatic generation of Web pages from existing databases,** including graphical displays and rudimentary analysis tools. "WebObjects", by NeXT Software, Incorporated, is an example of a recently released commercial tool that appears promising for this type of application.

But what if you want to hook a Web page up to your databases today? This paper shows how the Data & Analysis Center for Software (DACS) created Web pages to provide software engineers access to software metrics data through the Web. Users around the globe can select data to analyze. Graphical and statistical summaries are created dynamically and presented to the user. This system exploits the usability of the Web to present results in a format a manager can use in planning or monitoring a software project.

This application uses a commercial database and public domain Internet tools, including a Sybase database, Structured Query Language (SQL), SybPerl, Gnuplot, Fudgit, and ImageMagick. These tools and others can be used for a wide variety of Web applications quite different from software metrics.



This Report is Available in its Entirety Through  
The DACS Web Site At:

<http://www.dacs.dtic.mil/techs/tr.shtml>

## EV and AIS

*Continued from page 1*

Today many believe earned value represents a natural extension of any structured software development effort, including automated information systems and information technology. Earned value provides a useful tool for any effort that results in products, the normal situation for software.

Current significant changes in both disciplines (earned value and automated information systems) make the melding of the two a natural result. Recognition of earned value as a program management tool moved it from a burden to a benefit. Pressures for improved effectiveness and accountability on the AIS community resulted in applying disciplined software development and resource management processes. This move toward more structure in the software development process encourages metrics including use of earned value management, while the need for resource management makes earned value a critical tool. By its nature the earned value process creates a disciplined approach that ties directly to other program activities such as configuration management and software metrics.

The concept of earned value relies on three principles: integration, planning, and evaluation. Integration refers to always considering the three parameters associated with project performance: technical objectives (scope), schedule (time), and resources (cost). The planning principle requires developing integrated plans for all work prior to performing. The evaluation effort relies on the objective assessment provided by earned value to force reconciliation of the plans with the performance against those plans.

The integration principle requires identifying the work effort, schedule, and resources such that the three parameters can be integrated at the day to day management level. It also requires a control process to assure the three elements remain consistent and tie to the overall program requirements. For our example the integration would be defined as follows:

Develop 5 modules of software:  
Plans project completing 2 in first month, 2 in second month, 1 in third month

### **Resource plan:**

1000 hours per month (diamonds)

The planning principle requires accounting for all work to be accomplished, identification and allowance for risk, and determining objective methods of measuring work completion. The planning forces the integration process down to the point at which work accomplishment can be traced. Planning also implies a continuous process to reflect improved knowledge over time due to the typical growth of knowledge over time or the evaluation process. In our case the planning process allocates the resources to specific modules to allow evaluation of performance.

Module Effort: Module 1= 500, Module 2=500, Module 3= 600, Module 4=400, Module 5=1,000

Finally, the evaluation principle requires examining the performance to date, identifying the cause and effect of deviations to the plan, and reflecting this knowledge back to the planning. The evaluation process thus integrates the status of the three elements in an objective manner.

## EV and AIS

*Continued from page 12*

### Status after 2 months:

1,600 hours spent.

Modules 1, 2 and 4 complete -

Earned Value = 1,400 (triangles)

This simplified example addresses the point that earned value allows differentiating cost and schedule problems through assigning values to the work accomplished. Spending less than the original plan only defines a positive situation when also accomplishing all the planned work. In this case the spending less than planned results from falling behind schedule. This evaluation compares the value of the work accomplished (1,400) against the value of the work planned (2,000) to show under accomplishment. By comparing the value of the work accomplished (1,400) against the cost spent (1,600) we recognize that the effort to accomplish the finished work exceeded plan. Thus, the effort shows as behind in schedule and exceeding planned cost.

Earned value effectively supports software development management since the structured planning and control required by earned value ties closely with the currently recommended software development processes and metrics. The major reluctance in applying earned value often

comes from the concern that successful completion of interim steps may not predict successful performance. True! However, unsuccessful completion of current efforts definitely predicts unsuccessful performance. The key lies in the planning itself.

The discipline of a detailed plan and control of changes moves decisions about the software ahead of the development process. This identifies risks and forces evaluation of schedule and cost impact of changes.

Thus, earned value cannot assure success in a software effort. It will, however, provide insight to the probability of success, force management of change, and allow objective evaluation of status.

For more information on earned value and software development contact:

**Fred Manzer** - Chair

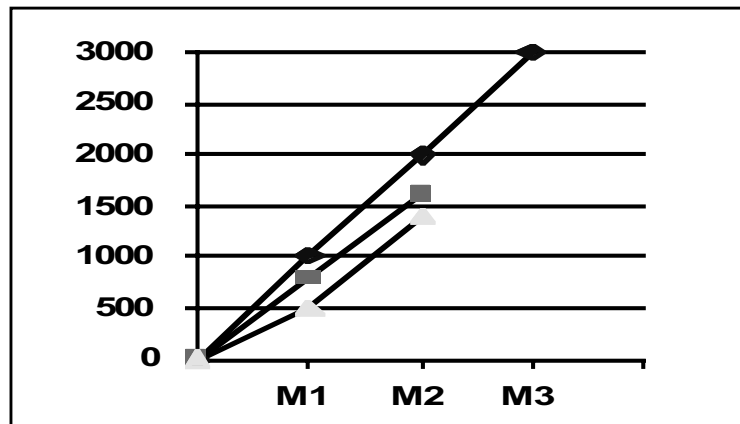
Earned Value Management Dept.

Defense Systems

Management College

(703) 805-3705

[manzerf@dsmc.dsm.mil](mailto:manzerf@dsmc.dsm.mil)



### Earned Value Related Web Sites:

**Undersecretary of Defense for Acquisition & Technology**

**Earned Value Management Web Page**

<http://www.acq.osd.mil/pm/>

**NNH Enterprise - Earned Value Page**

<http://www.nnh.com/>

**Air Force Acquisition - Earned Value Page**

[http://www.safaq.hq.af.mil/acq\\_pol/evaluate.html](http://www.safaq.hq.af.mil/acq_pol/evaluate.html)

**Earned Value Web Site**

hosted by Cost Management Systems, Inc.

<http://www.earnedvalue.com/>



## **The Data & Analysis Center for Software is sponsoring a half-day panel track at STC 98.**

### **Process Improvement via Rapid Prototyping Lessons Learned**

Four presenters have agreed to be part of the panel. Their info and abstract follows:

**1. Mr. Larry Bernstein** - President,  
National Software Council

#### **“Process Improvement via Rapid Prototyping”**

Prototyping the user interface screens based on the requirements and reviewing them with customers uncovers a number of gaps and misunderstandings that can be remedied. Finding the disconnects early, at the requirements phase of the project, saves about 30 times the effort of finding them after delivery to the customer. This prototyping approach is a best practice and can be extended throughout software system development.

Coupling prototyping with function point analysis measuring software size can control software complexity and feature-creep. The goal is to reduce the function point count by 40% (from initial requirements to delivery) while still delivering the required functionality. Prototypes allow designers to examine alternate approaches and simplify the product. The mantra is "Make it Work, Make it Work Right and, only then, Make it Work Better."

**2. Mr. Steve Cross** - Director,  
[Software Engineering Institute \(SEI\)](#)

#### **"Enterprise Improvement - Making it happen smarter, faster, cheaper"**

Many organizations now recognize the importance of process improvement and have witnessed benefits from using such industry standards as the Software CMM(r), ISO 9001, and other Capability Maturity Models(sm). But looking long term and to the larger picture; the secret to future industrial competitiveness will lie, in large part, on an organization's capability to more smartly and rapidly adopt promising new methodologies and technologies to improve business performance and achieve their long-term business goals.

**3. MAJ. David A. Dampier, Ph.D.** - Professor  
Information Resources Management College  
National Defense University

#### **"Rapid Prototyping and Incremental Evolution"**

Software development is no longer an enterprise where the traditional waterfall method of system construction is acceptable. Information technology is changing at a pace that requires complete system development and fielding in less than 18 months. This is due in part to faster technology insertion, and in part by increased user expectations. Both reasons provide justification for changing the way software is built and fielded. Increased user expectations require that we involve the user more in the requirements engineering process, and deliver the software to the user much more quickly. Faster technology insertion requires that we incorporate new technology into existing products much faster and with less rework.

A new software evolution process is needed to accomplish these goals, along with the automated tools to realize the benefits. Computer-Aided Prototyping is one such method that incorporates the goals and opinions of the user from the beginning of the software evolution process, throughout the lifecycle, and into retirement.

**4. Mr. Erik Mettala** - Vice President,  
Microelectronics and Computer Technology Corporation

Topic: To Be Decided

**The DACS Track at STC'98 will be  
24 April 1998 in Salt Lake City, Utah**

For more information visit the STC'98 Web Site at: <http://www.stsc.af.hill.mil/STC/>



## Software Technology Calendar of Events

---

### November 1997

9-13 [Conference]  
Tri-Ada '97

Sponsored by: ACM's Special Interest Group Ada (SIGADA)  
Adam's Mark Hotel  
St. Louis, MO USA  
POC: David Harrison, Conference Chair;  
dharrison@acm.org

10-14 [Course]

Personal Software Process (PSP Instructor Training)  
Sponsored by: Software Engineering Institute (SEI)  
Carnegie Mellon University  
Pittsburgh, PA USA  
POC: SEI Customer Relations: (412) 268-5800,  
Fax: (412) 268-5758, customer-relations@sei.cmu.edu

10-14 [Seminar]

Fall 1997 Professional Development Seminars  
Sponsored by: Washington, D.C. Chapter of the Association  
for Computing Machinery (ACM)  
University of Maryland University College  
College Park, Maryland USA  
POC: dcseminars@acm.org

15-16 [Call for Papers]

5th ACM Workshop on Geographic Information Systems  
Sponsored by: Association for Computing Machinery  
(ACM)  
Las Vegas, NV USA  
POC: Program Chair, Prof. Robert Laurini;  
Fax: +33 4 72 43 88 99; Robert.Laurini@if.insa-lyon.fr

24-27 [Forum]

Object Component Forum & Object Management Group  
Forum '97  
Sponsored by: International Council on Systems Engineering  
(INCOSE)  
Vienna, Austria  
POC: Mr. Manfred Zeitlhofer, Forum Administration;  
(431) 505-0900; (431) 505-0912; admin@oc-f.com

### December 1997

4-5 [Conference]

The Third Annual Software Metrics Conference  
Sponsored by: Education Foundation of the Association  
for Information Technology Professionals (EFAITP)  
and American Institute of Engineers (AIE)  
Washington, D. C. USA  
POC: Mark Mitchell, Conference Manager;  
(541) 484-4174; Fax: (541) 484-4174;  
msm4174@aol.com

### February 1998

22-25 [Conference]

11th Conference on Software Engineering Education  
and Training (CSEE&T)  
Sponsored by: Software Engineering Institute and IEEE  
Computer Society (pending)  
Atlanta, Georgia USA  
POC: W. Michael McCracken, Chair;  
mike@cc.gatech.edu

### March 1998

9-11 [Conference]

2nd Euromicro Working Conference on Software  
Maintenance and Reengineering (CSMRE '98)  
Palazzo degli Affari  
Florence, Italy  
POC: Paolo Nesi, Program Chair;  
Tel: +39-55-4796523; Fax: +39-55-4796363;  
nesi@ingfi1.ing.unifi.it

9-12 [Conference]

10th Software Engineering Process Group Conference:  
(SEPG '98)  
Sponsored by: Software Engineering Institute and  
Chicago Software Process Improvement Network  
(C-SPIN)  
Hyatt Regency Hotel  
Chicago, Illinois USA  
POC: SEI Customer Relations; (412) 268-5800;  
Fax: (412) 268-5758

For A Complete Listing of Software Technology Events On-line:  
<http://www.dacs.dtic.mil/awareness/coe/cal.shtml>



**The DACS is:**

A United States Department of Defense  
Information Analysis Center (IAC),

Administered by the Defense Technical  
Information Center, Ft. Belvoir VA,

Technically managed by Rome Laboratory,  
Rome NY, and

Operated by ITT Industries Systmes &  
Sciences Corporation, - Rome NY.

**To Contact the DACS  
you may either:**

**Contact the DACS Customer Liaison:**

E-mail: [cust-liasn@dacs.dtic.mil](mailto:cust-liasn@dacs.dtic.mil)

Telephone: (315) 334-4905

Fax: (315) 334-4965

or

Visit the DACS World Wide Web Site at;

<http://www.dacs.dtic.mil/>

**Arranging a  
Conference or Seminar?**

**Let the DACS help.**

The DACS will provide  
solutions to all your  
conference and seminar  
needs. Our experienced  
conference management team  
will provide:



- **Pre-conference Support** (brochures, graphic design, mailings, agenda development, booth displays, Internet postings)
- **Identification** (speakers and attendees)
- **Logistical Support** (audio/visual, catering, site management)
- **Technical Support** (computing, proceedings, telecommunications)

**For additional information contact:**

**Nancy L. Sunderhaft**

**DACS Conference Manager**

**Voice: (315) 334-4949**

**Fax: (315) 334-4965**

**E-mail: [dacs-coremgr@dacs.dtic.mil](mailto:dacs-coremgr@dacs.dtic.mil)**

**ITT Industries Systmes & Sciences Corporation  
Data & Analysis Center for Software  
P.O. Box 1400  
Rome, NY 13442-1400**



# DACCS Products & Services Order Form

Name:	Position/Title:
Organization:	Acronym:
Address:	
City:	State:
Zip Code:	
Country:	
Telephone:	Fax:
E-mail:	

<i>Product Description</i>	<i>Format</i>	<i>Quantity</i>	<i>Price</i>	<i>Total</i>
<b>The DACCS Information Package</b>				
<input type="checkbox"/> Including: Software Tech News newsletter, and a Special Studies Brochure	Document		FREE	FREE
<b>Empirical Data</b>				
<input type="checkbox"/> NASA / Software Eng Lab (SEL) Dataset	CD-ROM		\$150	
<input type="checkbox"/> NASA / AMES Dataset	CD-ROM		\$50	
<input type="checkbox"/> Software Reusability Dataset	Disk		\$50	
<input type="checkbox"/> DACS Productivity Dataset	Disk		\$50	
<b>Technical Reports</b>				
<input type="checkbox"/> A Business Case for Software Process Improvement	Document		FREE	FREE
<input type="checkbox"/> ROI from Software Process Improvement Spreadsheet	Diskette		\$ 40	
<input type="checkbox"/> An Overview of Object-Oriented Design	Document		\$ 25	
<input type="checkbox"/> An Analysis of 2 Formal Methods: VDM and Z	Document		\$ 30	
<input type="checkbox"/> A History of Software Measurement at Rome Laboratory	Document		FREE	FREE
<input type="checkbox"/> A Review of Formal Methods	Document		\$ 30	
<input type="checkbox"/> A Review of Non-Ada to Ada Conversion	Document		\$ 30	
<input type="checkbox"/> Artificial Neural Networks Technology	Document		\$ 30	
<input type="checkbox"/> A State of the Art Report: Software Design Methods	Document		\$ 30	
<input type="checkbox"/> A State of the Art Review of Distributable Database Technology	Document		\$ 30	
<input type="checkbox"/> Electronic Publishing on the World Wide Web: An Engineering Approach	Document		\$ 50	
<input type="checkbox"/> Object Oriented Database Management Systems	Document		\$ 30	
<input type="checkbox"/> Rome Laboratory Research in Software Measurement	Document		\$ 30	
<input type="checkbox"/> Software Analysis and Testing Technologies	Document		\$ 30	
<input type="checkbox"/> Software Prototyping and Requirements Engineering	Document		\$ 30	
<input type="checkbox"/> Software Interoperability	Document		\$ 30	
<input type="checkbox"/> Software Reusability	Document		\$ 25	
<b>Bibliographic Services</b>				
<input type="checkbox"/> DACS Custom Bibliographic Search	Diskette		\$ 50	
<input type="checkbox"/> DACS Software Engineering Bibliographic Database (SEBD)	CD-ROM		\$ 50	

**Method of Payment:**

- Check     Mastercard     Visa

*Number of  
Items Ordered*

*Total  
Cost*

Credit Card # \_\_\_\_\_

Name on Credit Card \_\_\_\_\_

Expiration Date \_\_\_\_\_

Mail this form to: DACS Customer Liaison  
Data & Analysis Center for Software  
P.O. Box 1400, Rome, NY 13442-1400

Telephone: (315) 334-4905  
Fax: (315) 334-4965  
E-mail: [cust-liasn@dacs.dtic.mil](mailto:cust-liasn@dacs.dtic.mil)