In This Issue...

- Practical Software Measurement Program Update
- The 7th Software Engineering Process Group Conference
- What's Your Opinion? Is the SEI's Capability Maturity Model (CMM) a Step Forward or Backward?
- The Center for High-Integrity Software System Assurance (CHISSA)
- Blue Ribbon Panel on Integration of Modeling and Simulation into Air Force Software Intensive Systems
- Announcing the NASA/SEL Dataset Update
- Analyst's Corner Function Points
- Software Engineering Events
- Call for Participation in Software Engineering: Ethical and Professional Standards of Practice
- DTIC Announces the PC Version of GoldenGate

Register Now for Software Tech News
Practical Software Measurement: A Guide to Objective Program Insight, was developed in response to the need for more effective techniques for managing DOD software intensive programs. The Practical Software Measurement (PSM) Program is sponsored by the Joint Logistics Commanders Joint Policy Coordinating Group on Computer Resources Management. Project participants include representatives from all of the DOD services and agencies involved in software measurement, as well as representatives from industry and academia.

The initial draft of 'Practical Software Measurement (PSM)' was distributed at the Software Technology Conference in April 1995. Since its initial release, the PSM has generated much interest within the DOD software community, and is already being implemented within many acquisition and development organizations.

The technical guidance included in the PSM was derived from actual DOD experience in applied software measurement. The PSM guidance emphasizes the integration of a practical measurement process within the program management function, and objective communication between the acquisition and development organizations. The measurement process is based upon a set of measurement principles that have been derived from successful program implementations. Rather than focus on a pre-defined set of measures, the PSM recommends the use of flexible measures and indicators driven by program-defined issues. Each measurement implementation, therefore, is tailored to the specific needs of the individual program.

PSM Version 2.0 is available and includes revisions to incorporate a revised software measurement planning model, enhanced analysis guidance, and an Automated Information Systems case study. In addition, PSM Version 2.0 addresses how to implement a measurement process on an existing program. A one-day PSM training curriculum is under development and will be available in September 1995. PSM Version 3.0 is scheduled for release in 1997 and will focus on advanced estimation and analysis techniques.

Additional information on the PSM Program and copies of the document are available from:

John McGarry
NUWC
(401) 841-3834 mcgarry@ada.npt.nuwc.navy.mil

Use these buttons to navigate between newsletter articles.
[Previous Article] [Next Article] [In This Issue]
Approximately 1,300 people attended the 7th Annual Software Engineering Process Group (SEPG) Conference on 22 through 25 May 1995 in Boston, Massachusetts. Co-sponsored by the Boston Software Process Improvement Network (SPIN) and the Software Engineering Institute (SEI), this conference, as noted by Julia Allen of the SEI, has grown in attendance from 46 people in 1988 to 1,300 today. Joseph Farinello, of the Boston SPIN and co-chair of the conference, stated that the SEPG conference has become the largest SEI-sponsored event. A major objective of the 1995 conference was to present "hard" data and results of SPI initiatives.

John D. Warner, President of Boeing Computer Services, in one of two keynote speeches, discussed "Software Development and Customer Satisfaction." Mr. Warner said the software industry needs to move toward the quality of other industry products, whereby software organizations will provide guarantees to their customers that their products will be defect-free for 25 years or they fix problems for free. He pointed out that the software industry is slow to mature. In 1988, 88% of all software organizations were at SEI Level 1 (the initial or lowest SEI level). In 1993, 85% of all software organizations were still at Level 1. Mr. Warner described the success of the Boeing 777 aircraft as "three million parts flying in close formation." The 777 contains four million source lines of code, the software systems are all functional, and the 777 is service ready and on schedule. The 777 was designed on CAD/CAM systems and involved thousands of engineers and scientists working in the U.S. and Japan.

Eileen S. Quann, President of Fastrak Training, talked about "Transforming Your Workplace Into a Learning Organization." Her major premise was that the most successful software organizations are those organizations that institutionalize learning. Technology shelf life is now five years, that change is constant, to remain competitive we must engage in professional development activities. Professional development is the process of continuing education for all employees. The learning process must be institutionalized; organizations and individuals must commit to continuous learning and training; individuals, organizations and customers should share the cost of learning; and acquisition strategies must change to provide incentives to companies that train.

At the general session on the second day, a lively debate occurred between panelists Bill Curtis, VP and Chief Scientist of Teraquest Metrics, and James Bach, Senior Staff QA Engineer for Borland, around "The CMM Controversy: Is it a Step Forward or a Step Backward?" This was a continuation of a debate that first began in the September 1994 issue of American Programmer magazine. Mr. Curtis, who supported the CMM, stated the CMM is Total Quality Management-based, which results in repeatability and discipline from organizations that employ CMM, and the empirical data supports the CMM approach. The benefits of the CMM include: professional standards in software engineering; improved schedules, costs, and product quality; agility in fast paced markets; and improved skills from software professionals. Mr. Bach contends that the CMM is a step backward, and criticizes the CMM because he believes there is no formal theoretical...
basis to the CMM, vague empirical data supports the CMM, the CMM ignores people, and institutionalizing the CMM reduces software professionals' creativity. He stated that the CMM implies that Level 1 organizations are "barbaric" when in fact good work occurs in Level 1 companies, dissenters are assumed to be "afraid of change," higher levels are assumed to be better, and that KPAs (Key Process Areas) are always applicable regardless of industry segment, people or any other factor.

Will Hayes and Dave Zubrow of the SEI, in a presentation titled "Moving On Up: Data and experience Doing CMM-Based SPI", presented some preliminary results on new data about organizations that have been reassessed versus first time assessments. Of the organizations having a first time assessment, commercial organizations (39%) outnumber DoD Contractors (36%). 73% of all organizations having their initial assessment are at Level 1; 25% of those being reassessed remain at Level 1. Figure 1 graphically depicts the process maturity path for 47 organizations that have been reassessed at least once. For example, of the 14 organizations that started at Level 2 in the first assessment, 13 of them moved to Level 3 in the second assessment. The presenters stated that the average time for organizations to move up one maturity level was 18-30 months.

Dennis Goldenson of the SEI presented the results of the first systematic survey of the benefits of higher maturity levels in his presentation entitled "What Happens After the Appraisal?" Goldenson stated that higher maturity levels resulted in improved product quality, staff morale, customer satisfaction, ability to meet schedule, staff productivity, and the ability to meet budget commitments.

Other presentations at the conference included:

Bill Curtis in his tutorial on "Building a Cost-Benefit Case for Software Process Improvement" stated that many companies are achieving cost benefits of 6 to 1 and higher with process improvement, a 200-300 percent increase in productivity, and a 100 percent decrease in delivered defects. However, two-thirds of all process improvement programs fail. The cause of failure was due to executives not being behind the program, resistance from middle management, and ineffective SEPG leadership.

Donna Dunaway of the SEI provided information concerning the new CMM Based Appraisal (CBA) Internal Process Improvement (IPI) method. The CMM Appraisal Framework (CAF) Maturity Questionnaire is used as a starting point for the IPI assessments. Free form interviews are used along with the questionnaire to obtain improvement information. The new CBA IPI focuses on the goals of the CMM, rating them consistent with the CMM Appraisal Framework. From an IPI standpoint, the activities of the CMM are not prescriptive, but are indicative of fulfillment of the goals of the CMM. Any KPAs that are omitted must be justified in writing, but the major emphasis is goal satisfaction. Ms. Dunaway noted that IPI assessments can be tailored to specific projects or focus only on specified KPAs. Twenty-seven CBA IPIs have been conducted since June 1994 and there are currently 116 Lead Assessors in 51 organizations. The SEI continues to gather data, review and examine potential revisions for the CBA IPI method.

Ray Madachy of Litton Data Systems, in discussing the benefits of inspections, reported that for every person-hour of effort applied to inspections, 2.3 person-hours are saved in testing, and inspections reduced by 2/3 the number of defects detected during integration.

Version 2 of the CMM is planned for the 1996-1998 time frame.
Use these buttons to navigate between newsletter articles.

[Previous Article] [Next Article] [In This Issue]

DoD DACS Home Page
The Software Engineering Institute (SEI) Capability Maturity Model (CMM) is a framework for demonstrating the key process areas of an effective software process. The CMM is intended to describe an evolutionary improvement path for software development from an ad hoc, immature process to a mature, disciplined process, in five Levels.

At the Seventh Software Engineering Process Group Conference and in the September 1994 issue of American Programmer, Dr. Bill Curtis and Mr. James Bach debated whether the CMM is a step forward or a step backward. Mr. Bach criticizes the CMM for having no formal theoretical and only a vague empirical basis, revering institutionalization of process for its own sake, and encouraging a displacement of goals from improving an organization’s process to one of achieving a higher maturity level. In defense of the CMM, Dr. Curtis states that the CMM establishes professional standards in software engineering, were derived from inputs of 500 individual opinions of best practices, and results in improved skills and performance of both individuals and companies. We would like you to E-Mail your opinions and comments on the CMM to the DACS. Your individual opinions will be kept confidential. If you have a positive opinion of the CMM, please E-mail your opinions to CMM-pro@dacs.dtic.mil; if you have a negative opinion, please E-Mail your opinions to CMM-con@dacs.dtic.mil.

In developing your response, please consider the following questions:

1. Does the CMM help improve a software organization's development process? If so, how? If not, why not?

2. What problems arise in implementing the CMM, if any?

3. What issues has the CMM ignored?

4. In your opinion, do the CMM levels accurately and fairly represent the strengths and weaknesses of a software organization?

5. Will implementing the CMM improve the morale of your software organization? Will it lower employee turnover rate?

6. If your organization has implemented a CMM Software Process Improvement Program, has it been a success or failure? Why?

Please provide your opinions by 15 November 1995. We will tabulate the results and provide them in the Spring 1996 issue of the DACS Newsletter.

In your response, please identify the following:

1. Are you a member of the DoD, other Government, Academia, a DoD Contractor or other Commercial Organization, or other type of organization.

2. What is your current job title.

3. How did you learn about this survey.

4. What is your level of awareness of the CMM? Are you part of a Software Engineering Process Group?
involved in a CMM based project team?
The National Institute of Standards and Technology (NIST) High Integrity Software System Assurance program is currently involved in several high integrity software activities. This includes work performed as a result of contracts with other agencies, developing technical products in software engineering, organizing the NIST High Integrity System Lecture series, organizing the annual COMPASS conference, and other tasks in the high integrity area.

High-Integrity software is necessary in order for United States industries and government to function properly. NIST created CHISSA to identify criteria for software assurance for use by organizations that build or evaluate high-integrity software systems. CHISSA was organized to enable industry to build high integrity software systems using technology with defined benefits. To achieve this goal, CHISSA has these objectives:

- Collaborate with industry to determine high integrity software technology requirements,
- Identify high leverage research topics and potentially beneficial research results,
- Identify technology issues between software and other system components,
- Provide a mechanism for linking research, measurement, and transfer of software technology related to similar efforts for other system components,
- Provide for measurement and assessment of technology in real application projects,
- Identify mechanisms for integration of technology,
- Promote continuous training for engineers and scientists,
- Promote development of guidance and standards, and
- Provide results that will be made available to those organizations developing rules, policies, or contracting requirements to help them ensure that the rules, policies, and requirements are economically and technically feasible.

An outside Steering Committee of industrial, government, and academic experts has been organized to provide an independent assessment of the role for CHISSA and for providing program guidance on CHISSA plans. A series of Calls for White Papers provided the opportunity for industry, government, and university experts to make their needs known to CHISSA. In late 1994 a call for white papers resulted in 94 submissions to CHISSA. The papers were reviewed and formed the basis for CHISSA's activities.

CHISSA will provide focus for research organizations interested in working on specific problems with other companies and government agencies sharing those problems. CHISSA will act as a clearinghouse for reports to be shared among all participants, for data to be made available to the community, and for making tools available for others to evaluate. CHISSA will work with other funding agencies (e.g., National Science Foundation) in order to identify and propose programs that address mutual concerns in high integrity software.

CHISSA must address several large issues immediately to have a positive impact on industry. The white papers influenced CHISSA short and long term goals:
- Many U.S. industries are unaware of usable technology,
- Many research experiments are not being conducted with appropriate analysis of the collected data,
- There is no current service providing guided access to information on software methods or industry needs, and
- Research results are not packaged in a manner conducive to industry adopting the technology.

CHISSA's primary short term goal is to address the role of measurement within software engineering. CHISSA’s long term goal is to understand effective technologies for producing high integrity system software.

CHISSA will provide a mechanism for disseminating results by creating a Demonstration Facility, which will provide a resource at NIST with links to other relevant technology. The demonstration facility will be built on the Internet to make the results of CHISSA activities available to a large audience. Initial plans are to create a World Wide Web (WWW) site for browsing among available data. A longer-range goal is to populate the Demonstration Facility with other tools that support the production of high integrity software. Of particular concern is the requirement to make research and prototype tools available for study and evaluation without implying NIST endorsement of a particular product.

CHISSA will provide the clearinghouse, through workshops, conferences, and publications, to facilitate industry and universities to meet and to develop projects of mutual benefit. Industry must understand that its problems are amenable to academic study, and the academic world needs to know what the industrial problems are as a source of interesting research.

For additional CHISSA information, please contact:

Dolores Wallace
NIST
(301)975-334
dwallace@nist.gov

Note: This article was prepared from NIST Report, "Center for High Integrity Software Systems Assurance - Initial Goals and Activities, NISTR 5677, " by Dolores Wallace and Marvin Zelkowitz.
On 1 and 2 August a blue ribbon panel met on the University of California campus to investigate technologies for integrating modeling and simulation (M&S) into the acquisition of software intensive systems of the USAF. The workshop was hosted by Barry Boehm and Walt Scacchi of the University of California (USC), and sponsored by Lloyd Mosemann, Deputy Assistant Secretary of the Air Force for Communications, Computer and Support Systems. The attendees came from a variety of government, academic, and industrial organizations.

The workshop was instigated by Mr. Sam DiNitto of the USAF's Rome Laboratory. The seminal idea of the workshop stemmed from a paper that Walt Scacchi had prepared on "Understanding Software Productivity". The objective of the workshop was to propose research and development initiatives that should be considered by the USAF, focused on modeling and simulation technologies, that could be employed to improve the predictability for the acquisition of software intensive systems.

The first part of the workshop was a panel during which Barry Boehm and Walt Scacchi presented ideas for the program from a seminal white paper that they had prepared entitled Simulation and Modeling for Software Acquisition (SAMSA): Plans and White Paper. Dr. Ed Feigenbaum, AF Chief Scientist, presented his views of the concept. Bob Kent, of the USAF's Electronic Systems Center, made a presentation of the Electronic Systems Command (ESC) PRISM program, a component based program to establish a product line acquisition infrastructure for command and control systems acquired by the ESC. That was followed by a series of brief presentations from the blue ribbon participants which responded to the points of the white paper. Subsequently, three working groups examined different aspects of the initiative: Architecture/Product M&S Technology, Process M&S Technology, and M&S Technology/Acquisition Integration.

The working groups reported on their findings at the conclusion of the two day workshop. While these recommendations have to be considered preliminary the top level concepts included:

- the development of a Feasibility Analysis Model. This model would be used by acquisition planners to determine the early feasibility of system implementation based on a given set of requirements. The model would be enhanced, in incremental fashion, to develop a robust model that would include advanced capabilities.

- the development of an Architecture Feasibility Workbench. This workbench would be used by acquisition planners to plan the acquisition of systems. Its capabilities would include knowledge bases for different application domains, analysis basis's supporting tradeoff decisions, and the integration of various supporting models.

- the development of an advanced COCOMO style model that would afford consideration for a different basis other than size (e.g., source lines of code), the provision for creeping requirements, dealing with hardware/software architectural choices, COTS integration and reuse, sensitivity and confidence analysis, and multi-site and life-cycle considerations.

The blue ribbon panel plans on meeting October 16 and 17 to refine the recommendations of the workshop into a coherent program for the initiative. It is expected to focus on the Feasibility Analysis Workbench, integrating into the workbench the various technology suggestions that emanated from the other recommendations of the working groups.
While it would be entirely premature to predict the result of the blue ribbon panel's efforts, it certainly is well conceived and will could result in a significant program and capability software acquisition. Over the next year it is expected that efforts will be focused on articulating the initiative to the affected and interested parties across the Air Force in order to develop acceptance for the concept. The results of the initiative are expected to be fully documented in a report delivered to Mr. Mosemann's office in December of this year.
The Software Engineering Laboratory (SEL) was created in 1976 by National Aeronautics and Space Administration/Goddard Space Flight Center (NASA/GSFC) to continuously improve both GSFC software products and processes. The SEL is operated by the Software Engineering Branch at NASA/GSFC, the Department of Computer Science at the University of Maryland, and Computer Sciences Corporation.

The SEL supports process improvement with an extensive data collection program on software developed or maintained by GSFC. The SEL dataset includes high level schedule information for each project, such as major milestone dates; low level schedule and effort data, such as weekly effort data; detailed size and metrics data, such as Source Lines of Code for each module in a project; and detailed change and problem report data. The SEL dataset also contains data from controlled experiments on new technologies such as code reading, cleanroom development, Ada, and Object Oriented Design. In short, although the data characterizes one environment, no more detailed, thorough, nor complete source of empirical software data is publicly available.

NASA/SEL distributes its dataset through the DACS. Periodic updates are provided by NASA/SEL and announced by the DACS. The NASA/SEL update (March 1995) is now available. The dataset contains data on 158 GSFC projects. To illustrate analyses supported by the data, Table 1 presents an extremely small portion of the data, the type of changes in the small number of Ada projects in the dataset. Educators and those organizations beginning to adopt Ada might use this data in deciding on emphases in Ada training.

### Table 1: Type of Changes in Ada Projects

<table>
<thead>
<tr>
<th>Change Type</th>
<th>Number of Changes</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subprograms</td>
<td>1255</td>
<td>29.8%</td>
</tr>
<tr>
<td>Other</td>
<td>1143</td>
<td>27.1%</td>
</tr>
<tr>
<td>Data Typing</td>
<td>718</td>
<td>17.0%</td>
</tr>
<tr>
<td>Program Structure/Packages</td>
<td>532</td>
<td>12.6%</td>
</tr>
<tr>
<td>Generics</td>
<td>275</td>
<td>6.5%</td>
</tr>
<tr>
<td>Exceptions</td>
<td>209</td>
<td>5.0%</td>
</tr>
<tr>
<td>Tasking</td>
<td>54</td>
<td>1.3%</td>
</tr>
<tr>
<td>System Dependent Features</td>
<td>26</td>
<td>0.6%</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>4212</strong></td>
<td></td>
</tr>
</tbody>
</table>

The NASA/SEL dataset is distributed on a CD-ROM for $150. For more information about the NASA/SEL dataset, contact our Data Analyst at data-analyst@dacs.dtic.mil or our Customer Liaison.
You can contact the DACS directly by phone at (315)334-4905 or by Fax at (315)334-4965.
This article briefly introduces Function Points (FPs), a software size or an output measure. Adequate monitoring of process improvement and productivity requires measurement of output. Measures of output are also useful in estimating budgets and schedules during project planning. Ideally, a software size or output measure should be easily estimated in the requirements phase of a software project.

Allan Albrecht (Reference 1), in collaboration with John Gaffney, Jr. (Reference 2), designed FPs as a direct measure of functionality. FPs are a weighted sum of the number of inputs, outputs, user inquiries, files, and interfaces to a system. The latest counting rules are defined in Release 3.0 (1990) of "Function Point Counting Practices Manual," by the International Function Points Users Group (IFPUG). IFPUG can be contacted at:

Function Points Users Group (IFPUG)
(614) 895-7130
102214.2013@compuserve.com

FPs are most appropriate for information systems. Feature Points (Reference 3) are the most well-known modification for including the algorithmic complexity typical of embedded systems.

What are typical system sizes for systems in terms of FPs? References 1 and 2 provide FPs datasets, but the distribution of FPs is statistically significantly different in the two datasets. Figure 1 shows the distribution of COBOL projects in the more recent dataset (Reference 4). These projects range from 39 to 450 Thousand Source Lines of Code (KSLOC) in the older measure. The distribution of productivity for COBOL projects is not statistically significantly different in the two datasets, and this distribution is shown in Figure 2. Finally, Table 1 (Reference 5), provides a map from FP to SLOC.
### Table 1 - SLOC per FP by Language

<table>
<thead>
<tr>
<th>Language</th>
<th>SLOC per FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assembler</td>
<td>320</td>
</tr>
<tr>
<td>C</td>
<td>150</td>
</tr>
<tr>
<td>Algol</td>
<td>106</td>
</tr>
<tr>
<td>Cobol</td>
<td>106</td>
</tr>
<tr>
<td>Fortran</td>
<td>106</td>
</tr>
<tr>
<td>Jovial</td>
<td>106</td>
</tr>
<tr>
<td>Pascal</td>
<td>91</td>
</tr>
<tr>
<td>RPG</td>
<td>80</td>
</tr>
</tbody>
</table>

**Function Points Distribution**

- Based on 13 projects
- Median: 993
- Mean: 1,064
- Std. Dev.: 577

**Productivity Distribution**

- Based on 31 projects
- Median: 5.57
- Mean: 6.64
- Std. Dev.: 4.02
<table>
<thead>
<tr>
<th>Language</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>PL/I</td>
<td>80</td>
</tr>
<tr>
<td>Ada</td>
<td>71</td>
</tr>
<tr>
<td>Lisp</td>
<td>64</td>
</tr>
<tr>
<td>Basic</td>
<td>64</td>
</tr>
<tr>
<td>4th Generation Database</td>
<td>40</td>
</tr>
<tr>
<td>APL</td>
<td>32</td>
</tr>
<tr>
<td>Smalltalk</td>
<td>21</td>
</tr>
<tr>
<td>Query Languages</td>
<td>16</td>
</tr>
<tr>
<td>Spreadsheet Languages</td>
<td>6</td>
</tr>
</tbody>
</table>

References:


Use these buttons to navigate between newsletter articles.
[Previous Article] [Next Article] [In This Issue]
ACM Sigsoft '95
Third Symposium on the Foundations of Software Engineering
10-13 October 1995
Washington, DC
John Gannon
Dept of Computer Science
University of Maryland
College Park, MD 20742
International Conference on Software Maintenance (ICSM '95)
16-20 October 1995
Opio (Nice), France
Mari Georges
Cap Gemini Innovation
Paris, France
33 1 49 10 53 98
FAX 33 1 49 10 06 15
International Symposium on Software Reliability Engineering (ISSRE '95)
24-27 October 1995
Toulouse, France
Karama Kanoun
33 61 33 62 35
kanoun@laas.fri
Sixth International Conference on Applications of Software Measurement
Orlando, Florida
30 October - 2 November 1995
Software Quality Engineering
3000-2 Hartley Road
Jacksonville, FL 32257
(904) 268-8639
FAX (904) 268-0733

DTIC's Annual Users' Meeting & Training Conference
30 October - 2 November 1995
Arlington, Virginia
Julia Foscue
(703) 274-3848
DSN 284-3848
TRI-Ada '95
5-10 November 1995
Anaheim, California
Danieli & O'Keefe Associates
Conference Management
Chiswick Park
490 Boston Post Road
Sudbury, MA 01776
(508) 443-3330
FAX (508) 443-4715

Fourth SEI Conference on
Software Risk
6-8 November 1995
Monterey, California
Customer Relations
(412) 268-5800
FAX (412) 268-5758

Tenth Knowledge-Based
Software Engineering
Conference (KBSE-95)
12-15 November 1995
Nancy Sunderhaft
(315) 334-4905
nancy@dacs.dtic.mil
http://www.dacs.dtic.mil/

Twentieth Annual Software Engineering Workshop
29-30 November 1995
NASA/Goddard Space Flight Center
Attn: Software Engineering Workshop
(301) 286-6347
http://fdd.gsfc.nasa.gov/selsew.html

Check out the DACS Calendar of Events for other upcoming Software Engineering Events.

Use these buttons to navigate between newsletter articles.
[Previous Article] [Next Article] [In This Issue]

DoD DACS Home Page
This steering committee has organized three task forces to accomplish this mission. One task force is identifying the body of knowledge and recommended practices, and another task force is addressing educational issues. The IEEE and ACM task force on Software Engineering Ethics and Professional Practices (SEEPP) is tasked with documenting and codifying standards of ethical and professional practices for Software Engineers. We invite your participation in this task. You do NOT need to be a member of either society in order to participate.

The SEEPP task force recognizes that the design, development and application of software often transcend national boundaries. We feel that it is vitally important that the global computing community have the means to insure that this project produces a product which is as reflective of the global computing community's wisdom as can be reasonably achieved. The participation of individuals from throughout the world is important to achieving this goal.

The purpose of the SEEPP task force is to document the international consensus for minimally acceptable ethical/professional responsibilities and obligations of Software Engineers.

Participation is open to all individuals who are directly and materially affected or interested in these issues. Members of the international computing community are particularly encouraged to participate.

Please contact either of the following for additional information:

Robert J. Melford
IEEE-CS Co-Chairman
r.melford@computer.org or

Donald Gotterbarn, Ph.D.
ACM Co-Chairman
d.gotterbarn@computer.org
GoldenGate selects your information sources from databases like the DROLS Technical Report (TR) Bibliographic Database and the DROLS Work Unit Information System (WUIS) plus hundreds of others from Orbit, CD-Plus, Dialog, and Legi-Slate, encompassing a broad range of subjects.

GoldenGate allows you to edit your results off-line, saving costly database connection charges. Once you've saved your search results in full-text or bibliographic form they're ready for word processing, spreadsheets, or electronic mail. GoldenGate also makes it easy to order DTIC documents right from your desktop!

At least 386 PC; 486 preferred; Microsoft Windows 3.1 or higher; DOS 5.0 or higher 4MB RAM minimum, 8MB recommended; 2MB free disk space (plus room for saved files) 12MB for Guided Tour (optional); Connection: Network: TCP/IP (WINSOCK Version 1.1 compliant) with Internet connection or modem: minimum 2400 Hayes-compatible modem; maximum 14.4K.

The cost is $50 for first year and a $20 annual renewal fee. The cost covers an on-line tutorial, extensive online help, User Guide and software upgrades. Requires accounts with one or more of the following: DTIC (DROLS), Dialog, Orbit, CD-Plus, or Legi-Slate.

For additional GoldenGate information:

(703) 274-6434, DSN 284-6434
(800) 225-3842, Selection 6, Submenu I
bcporder@dtic.mil

To use DTIC services, you must be a registered DTIC user. To register for DTIC or GoldenGate services, contact DTIC at:

(703) 274-6871, DSN 284-6871
(800) 225-3842, Selection 2
reghelp@dtic.mil

Use these buttons to navigate between newsletter articles.
[Previous Article] [DACS Order Form] [In This Issue]