



7:00 a.m. – 8:00 a.m.	Continental Breakfast with Sponsors/Exhibitors Today's Exhibits Open 7:00 a.m. – 7:00 p.m. Registration/Bookstore Open 7:00 a.m. – 7:00 p.m.				
8:00 a.m. – 8:15 a.m.	Opening Remarks: Granville Jones and Linda Westfall, ICSQ'07 Conference co-chairs				
8:15 a.m. – 9:30 a.m.	KEYNOTE ADDRESS: Ken Schwaber – When Will Microsoft Go Out of Business?				
9:30 a.m. – 10:00 a.m.	Refreshments and Networking with Sponsors/Exhibitors				
10:00 a.m. – 11:00 a.m. Concurrent Sessions	Session A1: Computer Science Education: Fitting Software Quality into the Curriculum Trudy Howles	Session A2: Rapidly Achieving Measurable ROI Using Early Defect Detection Invited Speaker: Timothy Olson	Session A3: Taking the Plunge: Implementing Session-Based Exploratory Testing Techniques Brenda Lee	Session A4: Software Quality Assurance and the Road to Process Improvement Patricia Loo	Session A5: Stakeholder Workshop: A Prerequisite for Good Requirements Elicitation Exhibitor Presentation – The Westfall Team
11:15 a.m. – 12:15 p.m. Concurrent Sessions	Session B1: Why Software Quality Assurance Practices Become Evil! Invited Speaker: Gregory Pope	Session B2: Data Quality Measurements in a Hospital Data Warehouse Richard Biehl	Session B3: Software Testing Insanity Theresa Hunt	Session B4: Team Management: Cingular Wireless's (AT&T) Journey to SQA Team Excellence Annemarie Colino	Session B5: TBD Exhibitor Presentation - Cognence
12:15 p.m. – 1:30 p.m.	Lunch and Networking with Sponsors/Exhibitors Hot-Topic Lunch Table Discussions (Tables Marked)				
1:30 p.m. – 2:30 p.m. Concurrent Sessions	Session C1: Beyond Usability: Improving Software Quality by Closing the "Expectation Gap" Nicole Radziwill	Session C2: How Managers Get So Stupid Invited Speaker: Robin Goldsmith	Session C3: Software Test Automation 101 Kenneth White	Session C4: Outside-In, Ad Hoc Software Quality through Error Analysis Mark Underwood	Session C5: TBD Exhibitor Presentation
2:30 p.m. – 3:00 p.m.	Refreshments and Networking with Sponsors/Exhibitors				
3:00 p.m. – 4:00 p.m. Concurrent Sessions	Session D1: TBD Speaker	Session D2: Secrets of CMMI for AGILE Organizations Jeff Dalton	Session D3: Automation Framework for Validating Web Based Applications Santosh Kalaskar	Session D4: The Evolution of the Continuous Representation for Process Capability Invited Speaker: Mark Paulk	Session D5: Exhibitor Presentation - Cognence
4:15 p.m. – 5:30 p.m.	Panel – Title: Introducing Agile Methods into Traditional Environments				
5:30 p.m. – 7:00 p.m.	Reception with Sponsors/Exhibitors				



8:15 a.m. – 9:30 a.m. Tuesday's Keynote Speaker: When Will Microsoft Go Out of Business? - Ken Schwaber

Abstract: The tradeoff between functionality and quality challenges all product companies. More is needed sooner, but reputation can suffer if quality is inadequate to the need. Or, can more than reputation suffer? Is there a line past which a company's ability to compete and survive starts to diminish? Does a company have any control over when it crosses this line? Ken Schwaber presents some insights into this dilemma from his recent work with companies adopting Scrum and Agile. The results aren't encouraging, the consequences appalling, and the alternatives few and difficult.

Bio: Ken Schwaber is a co-developer of Scrum, an Agile product/project management process, along with Jeff Sutherland. Ken is a signatory to the Agile Manifesto, and one of the founders of the Agile Alliance and the Scrum Alliance. Ken's current interests are improving the state of the software development profession so that our customers look forward to seeing us, we look forward to going to work, and our products are of excellent quality.

[Back to top](#)

10:00 a.m. - 11:00 a.m. Concurrent Sessions:



Session A1: Speaker: Trudy Howles - Computer Science Education: Fitting Software Quality into the Curriculum

Summary: The document *Computing Curricula 2001* is the current guideline used for Computer Science education. This document, co-authored by the IEEE Computer Society and the Association of Computing Machinery, details the suggested curriculum for undergraduate computer science majors. However, the document focuses on education and not on software development practices. Many computer science graduates are hired into software development jobs.

How can computer science educators identify and close the gaps?

Learn how innovative courses and initiatives are used to infuse software quality practices into the computer science education program at the Rochester Institute of Technology in Rochester, New York. It provides a model for other educators, and identifies mentoring and partnering opportunities for businesses to help foster knowledge and growth in this important area.

Abstract: Few would argue that today's education must provide more than knowledge of what's hot today. Future software professionals must possess a breadth and depth in knowledge areas, but still need the tools and ability to adapt to rapid changes in the discipline. Today's economy requires that everyone be able to adapt to new technologies and changes. What is state-of-the-art knowledge today might be antiquated skills tomorrow. The industry also needs developers who understand the big picture – long-term maintenance issues, cost, risks and quality needs.

The computing employment world is constantly changing. How can we better equip our young workers to adapt to change? What skills will add value to make them more attractive and competitive in the job market, especially given the growing off-shore outsourcing trends? Where are these fundamentals taught? How can business help educators find a strong balance of educational outcomes from existing programs?

In the fall of 1998, the Computer Society of the IEEE and the ACM joined forces to update the *Computing Curricula 1991* document. This effort created 20 task forces, and each was charged with defining the Body of Knowledge for computer science undergrads; the others focused on pedagogy. The task forces studied both two and 4 year college programs.

The task forces addressed both the technological and cultural changes since the publication of the previous document. As a result, many important topic areas were added including the World Wide Web and its applications, networking, graphics and multimedia, embedded systems, data management, human-computer interaction, the OO paradigm, software safety, security and cryptography. The task groups also addressed educational issues resulting from the influx of technology, the overall growth of computer applications, economic influences and the broadening discipline.

Noticeably absent is a focus on overall software quality. Many believe that computer science students don't need rigorous and in-depth courses in verification and validation, software quality and reliability, and other areas typically considered "software engineering" and not "computer science." Also, with the expanded topic base, there is little time in the curriculum to teach other topic areas.

This paper will address this issue, and discuss the proposed courses and initiatives designed to infuse software quality practices into the computer science education program at the Rochester Institute of Technology in Rochester, New York. The initiatives reflect a possible model for other educators, and identify mentoring and partnering opportunities for businesses to help foster knowledge and growth in this important area.

Bio: Trudy Howles is an associate professor of computer science at the Rochester Institute of Technology in Rochester, NY. Trudy worked for over 20 years as a software designer and engineer before moving into academia. She is active in the Rochester ASQ section and serves as chair of the Software Task Group.

[Back to top](#)



Session A2: INVITED SPEAKER: Timothy Olson -Rapidly Achieving Measurable ROI Using Early Defect Detection

Summary: Measurable return on investment (ROI) usually takes years to achieve. How can organizations achieve measurable ROI faster? Learn how to apply early defect detection processes and achieve measurable ROI within a year. The secret is to use early defect detection to dramatically increase defect removal efficiency (i.e., the total amount of defects identified early in the process). Best practices such as data driven checklists are presented. Measuring ROI using defect dollarization is discussed. Explore best-in-class early defect detection processes (i.e.,

inspections, walkthroughs, peer reviews) and ROI technical approaches (defect dollarization, cost of poor quality, cost performance index).

Bio: Timothy G. Olson is President of Quality Improvement Consultants, Inc (QIC). While performing quality consulting, Mr. Olson has helped organizations measurably improve quality and productivity, save millions of dollars in costs of poor quality, and has helped numerous organizations reach higher SEI maturity levels. Mr. Olson has been formally trained in Crosby, Deming, Juran, ISO, CMM[®], and CMMISM quality approaches. He is also a Juran Institute Associate. Mr. Olson was a lead-author of a Software Quality Course for the University of Minnesota, and he is currently a senior member of ASQ and a member of IEEE and NDIA.

[Back to top](#)

Session A3: Speaker: Brenda Lee - Taking the Plunge: Implementing Session-Based Exploratory Testing Techniques

Summary: Hear how an independent test team for Sandia National Laboratories used Session-Based Exploratory Testing techniques to conduct the system test of a web-based application that has been deployed for a division of the Department of Energy. It describes:

- How the team achieved management buy-in
- Deviations from 'traditional' Session-Based Exploratory Testing
- Lessons learned
- The future of SBET in their test environment

Abstract: Session-Based Exploratory Testing (SBET) is a widely used test methodology developed by James Bach and Jonathan Bach. It formalizes ad hoc testing by providing charters that describe areas of functionality to investigate and then allows pairs of testers to determine how to approach testing the functionality. This technique was used to test a web-based application developed by Sandia National Laboratories to provide inventory management for a division of the Department of Energy. However, this team deviated from traditional exploratory testing by documenting the requirements to be addressed as part of the session, as well as providing recommended test cases to execute during the session. As a result of these modifications, test team was able to validate 41 requirements and execute over 200 test cases using 17 test charters during a 7-day test window. In addition, a high level summary of the test results was provided for customer review only two days after the conclusion of system test.

The purpose of this presentation is to:

- Explain the business environment that led to the decision to use SBET
- How management buy-in was achieved
- Why deviations from traditional SBET were made
- Challenges experienced by the team
- Lessons learned
- The future of SBET in our test environment

Bio: Brenda S. Lee is a Senior Analyst I for Parallax, Inc., a wholly owned subsidiary of EnergySolutions, LLC. Her professional experience includes system testing, technical writing, usability testing, and user interface design for companies such as IBM, Sensormatic Electronics, Tyco Safety Products, and Sandia National Laboratories. She received her Bachelor of Science degree in Psychology from West Texas State University in 1987 and her Master of Arts degree in Psychology from New Mexico State University in 1991. In December 2005, she received her CSQE. She lives in Albuquerque, NM, with her husband, Michael.

James A. Reitzel is a Managing Analyst for Parallax, Inc., a wholly owned subsidiary of EnergySolutions, LLC. He has been working as a System Test Manager at Sandia National Laboratories for eleven years developing and supporting a system test. He earned his CSQE in June of 2001. He also worked as an Industrial Engineer for United Parcel Service for seven years. He has a Bachelor of Science in Biology (1977), a Master of Wildlife Science (1982) and a Master of Business Administration (1984), from New Mexico State University. He resides in Albuquerque, New Mexico, with his wife and two children.

[Back to top](#)

Session A4: Speaker: Patricia Loo - Software Quality Assurance and the Road to Process Improvement

Summary: What is software quality assurance like at a DOE government laboratory? Come, listen and take home examples of artifacts used at the Idaho National Laboratory (INL) for the implementation of Software Quality Assurance. From compliance to process improvement – how do you make the transition? Implementation strategies and tools will be presented that will provide benefit and process improvement initiatives for any size software development shop.

For additional guidance in software management practices, the INL Information Technologies (IT) organization has initiated implementation of a Software Lifecycle Framework. This framework has been integrated within the laboratory-wide management system. A demonstration of the implementation of the Software Lifecycle Framework will be provided.

Lessons learned and best practices are discussed along with possible methods for the successful implementation of a corporate-or laboratory-wide SQA Program and associated software engineering practices.

Abstract: The Idaho National Laboratory (INL) is a science-based, applied engineering national laboratory dedicated to meeting the nation's environmental, energy, nuclear technology, and national security needs. The INL is the leading laboratory in basic nuclear and radiological science research and applications and is a multiprogram, federally funded research and development center (FFRDC) providing solutions for use across the DOE complex, as well as regionally, nationally, and world wide.

The implementation of a Software Quality Assurance (SQA) Program at the Idaho National Laboratory (INL) is in compliance with DOE O 414.1C, "Quality Assurance," and NQA-1-2000, "Quality Assurance Requirements for Nuclear Facility Applications." For many years, software quality assurance has been envisioned as a compliance rather than a process improvement issue. SQA requirements were met to pass audits... rather than to really improve the quality of software. Discussion will be held on the graded approach and making SQA an integrated part of the software management process. Implementation strategies and tools will be presented that will provide benefit and process improvement initiatives for any size software development shop. These tools include:

1. Supporting procedures and guidance including the graded approach and referenced templates
2. IT Planning and Strategies including Service Level Management
3. Project Management
4. Business Process Modeling
5. Requirements management
6. Configuration management
7. Configuration Management Database and Enterprise Architecture Repository
8. Verification and Validation
9. Collaboration

The SQA Program at the INL is structured around a graded approach that is tightly integrated with the laboratory's Quality Assurance Program (QAP) and is based on consequence, probability of failure, and the software type. The level of rigor for software management practices can be graded in scope to support a website used for collaboration versus a software application directly associated with the controls of a nuclear reactor. A detailed discussion on this grading strategy and tools available for the end user will be presented.

For additional guidance to software project managers, software technical leads, and software engineers, the INL Information Technologies (IT) organization has initiated implementation of a Software Lifecycle Framework. This framework has been integrated within the laboratory-wide management system, Systems Integrating Management System (SIMS). The SIMS is the management system tasked with engineering the transformation of the INL. This overarching management system integrates management systems for the enterprise including project management, engineering, work management, document and records management, and contract management. Attendees of this session will have the opportunity to see a demonstration and review of this state-of-the-art management system and the integrated Software Lifecycle Framework.

In conclusion, for more take home ideas, suggested lessons learned and best practices will be discussed along with possible methods for the successful implementation of a corporate- or laboratory-wide SQA Program and associated software engineering practices.

Bio: Patricia Loo has worked in the Software Engineering field at the Idaho National Laboratory (INL) for over 22 years. Ms. Loo is an ASQ Certified Software Quality Engineer (CSQE), IEEE Certified Software Development Professional (CSDP) and a PMI Certified Project Management Professional. She has acted in the capacity of Technical Lead for systems including the DOE Occurrence Reporting and Processing System (ORPS) and the Integrated Waste Tracking System (IWTS). She currently works on the INL Enterprise Architecture and is responsible for the INL Software Quality Assurance Program. Ms. Loo holds a MS in Computer Science from the University of Idaho.

[Back to top](#)



Session A5: Exhibitor Presentation - The Westfall Team: Linda Westfall - Stakeholder Workshop: A Prerequisite for Good Requirements Elicitation

Summary: Before you can effectively elicit requirements information, you must identify the stakeholders for your software project. Participate in a interactive stakeholder workshop to learn stakeholder identification and participation strategy techniques.

Bio: Linda Westfall is the President of The Westfall Team, which provides Software Engineering, Software Quality and Software Project Management training and consulting services. Prior to starting her own business, Linda was the Senior Manager of Quality Metrics and Analysis at DSC Communications where her team designed and implemented a corporate wide metrics program. Linda has more than thirty years of experience in real time software engineering, quality and metrics. She has worked as a Software Engineer, Systems Analyst, Software Process Engineer and Manager of Production Software.

Very active professionally, Linda Westfall is a past chair of the American Society for Quality (ASQ) Software Division. She has also served as the Software Division's Program Chair and Certification Chair and on the ASQ National Certification Board. Linda is a past-chair of the Association for Software Engineering Excellence (ASEE) and chaired several conference program committees.

[Back to top](#)

11:15 a.m. - 12:15 p.m. Concurrent Sessions:



Session B1: INVITED SPEAKER: Gregory Pope - Why Software Quality Assurance Practices Become Evil

Summary: Gregory Pope explores the challenge of determining the best practices for software development and why the topic usually sparks a lively debate. The premise is that best practices are application specific and are not easily portable from one industry to another. He considers a case study of three different types of software developments and contrasts their differences. Learn how to apply an alternative method to best practices, which is a common set of principles that are turned into appropriate best practices based on project risk.

Abstract: After some 35 years of being involved (and hopefully evolved) in almost every aspect of the software development business from programmer to CEO, I have a theory on why Software Quality Assurance (SQA) practices become evil. When I say SQA Practices, I mean doing things like requirements management, software design, coding standards, inspections, unit test, integration test, system test, and, of course, management reviews.

Let me define my meaning of evil as well, using Webster's dictionary. I do not mean evil in the sense that SQA practices are morally reprehensible or sinful or wicked (definition #1). As far as I can tell SQA Practices do not violate the Ten Commandments. (Is it still legal for Microsoft Word to force me to capitalize Ten Commandments? How did the ACLU miss this one?) I do not mean that SQA Practices are evil because they arise from people who have actual or reputed bad character (definition #2). As far as I know, the people that gave us these SQA practices are well intentioned and reputable. Maybe I missed something, but I haven't yet heard of an illegal requirements dealer, defect embezzler, or inspection bimbo. What I do believe is that the third definition of evil is right on, that SQA practices cause discomfort and repulsion and are offensive. I can't remember a time when any developers I worked with bragged about what a great weekend they had doing SQA practices. The topic of SQA practices is usually greeted by developers with the same enthusiasm as a bad odor; of course the source of the odor is often oblivious to the smell.

[Back to top](#)

Session B2: Speaker: Richard Biehl - Data Quality Measurements in a Hospital Data Warehouse

Summary: Understand how a range of quality metrics and indicators have been designed into an enterprise data warehouse in an academic medical center setting. Learn how those metrics and indicators provide for a view of data and process quality that can drive error recognition and reduction, organizational performance improvement, and business trending and opportunity identification. Richard Biehl shares the designs and outcomes that have been achieved in the example hospital setting, as well as describing the design process and discussions that were necessary to achieve the design and its implementation. Receive a list of the quality rules that were implemented in this hospital, and consider adapting those rules into your own organizations. No knowledge or experience in data warehousing is required.

Abstract: Data warehouses bring together a diversity of raw data from across an enterprise. They provide an integrated view of customer and supplier interactions, as well as business processes, that can not be seen within any of the individual application systems from which the data is drawn. Such data consolidation offers the quality practitioner unique opportunities to monitor and control both the quality of the data collected, and the quality of the business processes described by such data.

This paper describes how a range of quality metrics and indicators have been designed into an enterprise data warehouse in an academic medical center setting, and how those metrics and indicators provide for a view of data and process quality that can drive error recognition and reduction, organizational performance improvement, and business trending and opportunity identification.

Examples of metrics described in this paper include detailed fact-level data quality indicators, hierarchy exception analysis (e.g., the generalized "other"), time-series exception analysis, and the use of generalized statistical process control to spot data or process exceptions. A generalized notification mechanism for assuring that alerts, errors, and notices are delivered to the appropriate organizational stakeholders using user subscriptions is also described.

The objectives of the presentation include sharing the designs and outcomes that have been achieved in the example hospital setting, as well as describing the design process and discussions that were necessary to achieve the design and its implementation. Attendees will be provided with a list of the quality rules that were implemented in this hospital, and will be encouraged to adapt those rules into their own organizations. Attendees from organizations not implementing data warehousing will be shown how these rules can be used by adapting them into their organization's project peer review processes.

This paper does not require any particular data warehousing knowledge or experience. While this paper uses a specific hospital example, the principles and techniques discussed generalize immediately into virtually any data warehousing project or setting. Indeed, they could be generalized to the design of any non-trivial information system application.

Bio: Rick Biehl is a senior quality consultant with twenty-nine years of experience specializing in quality assurance, logical and physical data architectures, and strategic planning for the business application of information technology. He holds a master's degree in education from Walden University, as well as CSSBB and CSQE certifications from ASQ.

[Back to top](#)



Session B3: Speaker: Theresa Hunt - Software Testing Insanity

Summary: Rich or poor – companies have the same bugs. The rich ones just spend more money not finding them. Surprisingly, even companies with abundant resources often lack understanding of where their real problems crawl. In this session we'll expose a list of commonly repeated testing inefficiencies, discuss how to avoid them, and determine which testing approaches are right for your application.

Bio: Theresa Hunt is owner/consultant at Hunt Technologies, Inc, specializing in Software Quality Engineering, Process Definition and Improvement, and Testing. She has worked as a lead consultant for The Westfall Team and as a Principal Software Quality Engineer and a member of the Engineering Process Group responsible for corporate wide improvement of systems, software, and hardware engineering at ECC International Corp. based in Orlando, Florida. Theresa managed ECC's Software Quality Engineering (SQE) department and directed Software Quality Engineers in the planning and execution of project-specific software quality functions.

Theresa serves as Vice Chair of Programs for the ASQ Software Division. The Programs Committee is responsible for planning and conducting the annual International Conference on Software Quality (ICSQ), the software tracts at ASQ's Annual Quality Congress, and partners with other international organizations in planning and conducting the World Congress on Software Quality.

[Back to top](#)

Session B4: Speaker: Annemarie Colino - Team Management: Cingular Wireless's (AT&T) Journey to SQA Team Excellence

Summary: How do you ensure that your Software Quality Assurance team is meeting the same high quality expectations they set for others and providing value to the company? Share the Cingular Wireless (now AT&T) SQA team's journey to Team Excellence which includes a Six Sigma improvement project, organizational change, automation, regular measurement and feedback, and analysis of the value provided by the SQA team.

Abstract: The Cingular Wireless Software Quality Assurance (SQA) Team partners with selected development project teams to develop an SQA Plan, provide process coaching, and monitor compliance to the SDLC process. Besides better quality outcomes, the SQA team ensures that project teams will easily meet SOX requirements.

Nonetheless too many audited projects in 2005 had some sort of documentation issue that required internal auditor and SQA management follow-up when it was supposed to be seamless. The quality of the SQA team's work was not up to par. In addition management wanted to measure the value of the investment in SQA. With this input, the SQA Team embarked on a quality improvement journey in 2006 which continues through today.

The first part of this journey was a Six Sigma improvement project which established process measurement and launched several improvement initiatives including peer reviews, procedure documentation, and an on-boarding process. Through the Voice of Customer and other Six Sigma methods, the team was able to understand the contributing factors to monitoring defects and what measures would be useful to the organization. In addition, the SQA Team contributed to SQA/SDLC tool automation and implemented customer satisfaction surveys.

By year-end several analyses were conducted to understand the impact of SQA including non compliance occurrences between major releases, impact on Cost Performance Index, and customer survey results. These started to provide an objective picture of how SQA was positively impacting IT.

While these efforts helped, the SQA monitoring process was still not "in control" at the end of the year with regards to monitoring quality. Likewise, impact analysis had suggested more avenues of measurement. Thus 2007 started with an organizational shift and a focus on Team Excellence, which has included team building, increased measurement, structured change management, and analyst driven improvement.

This presentation shares the details of the approaches, measurements, analyses and lessons from Cingular Wireless (now AT&T) SQA Team improvement journey so that others may benefit from the insights gained and think about the quality and value of their own SQA organizations.

Bio: Annemarie Colino is a Senior Member of ASQ and a Certified Quality Engineer. She has 20 years experience in quality engineering in roles ranging from engineer to consultant to director. She has been in her current role of SQA Program Director for several years which has included overseeing the integration of AT&T Wireless with Cingular Wireless and now with AT&T.

[Back to top](#)

Session B5: Exhibitor Presentation - Congnence

1:30 p.m. - 2:30 p.m. Concurrent Sessions:

Session C1: Speaker: Nicole Radziwill - Beyond Usability: Improving Software Quality by Closing the "expectation Gap"

Summary: Have you ever been part of a development project where all the requirements were fulfilled, the change control process was effective, the software was on time and close to budget, verification and validation was well organized and successful, usability tests were promising – and yet customers still were not satisfied by the product? This may be a result of the "expectation gap" between the written requirements, the assumed requirements, and the ever-increasing consumer expectations for quality in the 21st century. Where does this gap come from, and how can software quality professionals effectively address it? Theory and findings from human factors, ergonomics, and design science (an emerging focus area for software intensive systems development) can help us understand and shrink this expectation gap, improving the perceived quality of the software that we produce. Examples are provided from Nicole Radziwill's experience managing development in the complex software environment of giant telescopes.

Abstract: Have you ever been part of a development project where all the requirements were fulfilled, the change control process was effective, the software was on time and close to budget, verification and validation was well-organized and achieved its goals, usability tests were promising – and yet customers still were not satisfied by the product? This may be a result of the "expectation gap" between the written requirements, the assumed requirements, and the ever-increasing consumer expectations for quality in the 21st century. Where does this gap come from, and how can software quality professionals effectively address it? ISO 8402 defines quality as "the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs." In software development, the requirements management process is a mechanism to capture stated needs (e.g. through use cases that outline functional requirements), and to approximate implied needs by defining standards for performance, reliability and other quality attributes. Because software is a product-driven industry, the Deloitte Product Framework can be used to define software success in terms of product/brand alignment, segment dynamics, and value-centric engineering. Product/brand alignment essentially reflects expectation setting: the product must deliver not only what is explicitly promised, but also what the reputation of the producer implies that the product and process should deliver. Consideration of segment dynamics means that a product must be able to compete in the way in which it was intended. Value-centric engineering, in software development, refers to all of the elements addressed by the CSQE body of knowledge. But who is responsible for addressing alignment and segment dynamics, and how is this done? Theory and findings from human factors, ergonomics, and design science (an emerging area of focus for software intensive systems development) can help us understand and shrink this expectation gap, improving the perceived quality of the software that we produce.

Bio: Nicole Radziwill is Assistant Director for End-to-End Operations at the National Radio Astronomy Observatory (NRAO) headquarters in Charlottesville, Virginia, overseeing software development and service delivery for several telescopes. Before NRAO, she managed consulting engagements worldwide and worked in scientific computation at the National Oceanic and Atmospheric Administration (NOAA) in Boulder, Colorado, gaining over a decade of experience managing continuous improvement in business and technology. Nicole has a degree in meteorology, an MBA, and a PhD in Technology Management and Quality Systems in progress. An ASQ CQM, she also serves on the Regional Councilor team for ASQ's Software Division.

[Back to top](#)



Session C2: INVITED SPEAKER: Robin Goldsmith - How Managers Get So Stupid

Summary: My bosses are idiots, unable to tell their proverbial aperture from their proverbial appendage. You probably know from whence I speak. Whether dooming projects from the start with unrealistic budgets and schedules, failing to allow people to use the superior concepts and techniques the organization has paid big bucks for them to learn, or pushing projects into production before they're ready, boneheaded bosses aren't just to be laughed at in "Dilbert." In this attention-grabbing presentation, Robin Goldsmith (who is his own boss and still works for an idiot sometimes) explains how bosses become the way they are.

Bio: Robin F. Goldsmith, JD is President of Go Pro Management, Inc. consultancy, which he co-founded in 1982. He works directly with and trains business and systems professionals in improving software development/acquisition processes through more effective requirements definition, Proactive Testing™, managing outsourcing, project management, and measurement. He was International Vice President of the Association for Systems Management and Executive Editor of the Journal of Systems Management. A frequent speaker at leading conferences, he is the author of the recent Artech House book, Discovering REAL Business Requirements for Software Project Success.

[Back to top](#)

Session C3: Speaker: Kenneth White - Software Test Automation 101

Summary: What do the following three things have in common: Bigfoot, the Loch Ness Monster, and Automated Software Testing? Answer: Everyone has heard of them, but no one actually has any proof they exist. And at least there are pictures of the first two.

Nearly every company that writes software has, at one time or another, flirted with the idea of using automation to test their software. Many of these projects end in failure however, because the company jumps in head first not knowing what to expect or worse, expecting the wrong thing.

So the first step to doing test automation right is setting ground rules, and that is what Kenneth White proposes to do. The intent is to help the attendee answer the question: Just what am I getting myself into? The author has done test automation over the course of his career with half a dozen different tools and has noticed many common themes. The reader can find these tools via a simple web engine search, so the author will not focus on any particular tool or make tool recommendations. Rather, the intent is to focus on broader concepts that apply universally, regardless of the tool specifics.

Abstract: Software Test Automation can be an invaluable tool in the software quality assurance team's arsenal. Though automating tests can be a challenge, the team stands a better chance of success by being prepared, informed and realistic about what test automation can accomplish. Picking an automation methodology that fits the team's situation and avoiding poor automation practices is a key to success. Understanding the limits of automation will help the team avoid pitfalls; just as knowing where automation is useful will help the team maximize their investment in both time and money. The team will need to develop core skill sets in not just the technical aspects of the tool itself, but in broader areas such as data management. Being prepared and having a realistic understanding of the challenges and benefits of Software Test Automation will increase the odds that the team will successfully implement their automation strategy and not leave the box of automation software collecting dust on a shelf.

[Back to top](#)

Session C4: Speaker: Mark Underwood - Outside-In, Ad Hoc Software Quality through Error Analysis

Summary: Consider this genuine error message from Google's RSS reader: "Oops! That wasn't supposed to happen." A quality engineer might choose one of two paths. The traditional path focuses on avoiding the error condition altogether. Top down and life cycle techniques to enhance software quality can achieve excellent results, but they are often not utilized. The other path focuses on productive uses for the error condition itself, which is the focus of this analysis. For such situations, a novel outside-in, bottom-up approach is suggested which exploits external error reports and uses them as quality probes to improve code quality. Drawing upon numerous representative error reports from public and private web sites, desktop and open source software, a general method is suggested that can benefit even the smallest teams. Implications for UML and design patterns are identified.

Abstract: Error messages matter. A reliance on search engines within developer communities reveals an interesting fact. Phrases associated with common error conditions are frequently given to a search engine for lookup. In fact, entire web sites are devoted to performing root cause analysis for narrowly prescribed problems E.g., <http://www.updatexp.com/0x8007007E.html> exploits error code 0x8007007E with Windows Update. So frequently are such sites visited that they appear higher in Google rankings than Microsoft's own FAQs on the same subject. This workflow is so predictable that it is possible for Google's page examination logic to identify the error messages as they are rendered and to auto-suggest possible solution sites (using rankings based upon traffic rather than other software engineering considerations).

In a wittily titled proposal for random methods in software analysis, "The Strangest Thing About Software," Menzies, Owen and Richardson muse that "modern software is so complex that it should never work" (*Computer*, 40(1), 2007, p. 54). Reasons given for this vary greatly, but certainly the collaborative aspect of software development is more complex today, with ever greater dependence upon frameworks, libraries, open source components and widgets a part of the story. There is increased reliance upon a worldwide

community of developers. The resulting collection of potentially heterogeneous modules brings with it a need for developers to understand interfaces between components. Instead, it seems equally likely that the very ubiquity of these modules (e.g., consider libraries linking php and MySQL) ensures that interface assumptions and graceful error processing will be ignored.

The quality practitioner all too often enters this milieu at the end of the pipeline. "The traditional thrust of software quality assurance has been to use 'brute force' testing at the end of development" (Moody, Sindre, 2003). Solving this conundrum is beyond the scope of the present discussion. Instead, a highly specific focus on errors is suggested as a means for meaningfully affecting the development process while maintaining a respectable distance from developer styles and habits, which vary greatly, notoriously even among the most gifted. The proposed method parallels the quality engineer's fault analysis model, but there are significant differences. First, a simplistic bottom-up approach is recommended that treats representative "errors" as a new class of use cases to augment the developer's design. Second, it seeks out errors that can function in ways analogous to use cases, and gives them added attention while minimizing others.

Recommendations encompass several facets of the SDLC: (1) error detection and user interfaces; (2) logging and a quality engineer's access to the ontology of an application domain; (3) errors as triggers for post hoc test plans; (4) role of customer service teams in reshaping developer views of quality thresholds; (5) overcoming apparent error message dead ends; (6) expanded scope of failure for web applications; (7) identifying requirements for exogenous failure cascades; (8) divided responsibilities for providing error context.

Bio: Mark Underwood is a Senior Systems Analyst with Applied Visions, Inc. in Northport NY. His technical interests are in serious games, simulation, and rule-based systems. He is currently working to provide enterprise network defenders with access to simulation datasets (Simblend) and a commercial game concept, Network Combat. His other professional associations include AAAI, ACM and IEEE-CS. Underwood was founder or co-founder of three software startups. With Peter Culicover, he designed the Context Activated Memory Device (CAMD), a linguistic processing engine that incorporated human memory constraints.

[Back to top](#)

[Session C5: Exhibitor Presentation - To Be Announced](#)

[Back to top](#)

3:00 p.m. - 4:00 p.m. Concurrent Sessions:

[Session D1: Speaker To Be Announced](#)

[Back to top](#)



[Session D2: Speaker: Jeff Dalton - Secrets of CMMI for AGILE Organizations](#)

Summary: As the CMMI gains popularity and acceptance across the globe, many agile IT and Engineering organizations feel that the CMMI is not a fit for them and that the SEI has left them out. Not only do they want to leverage a best-practices model for process improvement, but many of their customers are asking them to reach CMMI Maturity Level two or even level three!

Some think the CMMI is a heavy, burdensome process model only meant for large scale software development in the defense, aerospace, and pharmaceutical industries. It just isn't so! The CMMI is a valuable tool for helping small and large organization's alike achieve a superior level of performance through greater efficiency, less rework and all-around higher quality.

Join Lead Appraiser, author, and consultant Jeff Dalton as he reveals the secrets of CMMI appraisals by taking you behind the scenes using actual appraisal data and client case studies.

Abstract: As the CMMI gains popularity and acceptance across the globe, many agile IT and Engineering organizations feel that the CMMI is not a fit for them and that the SEI has left them out. Not only do they want to leverage a best-practices model for process improvement, but many of their customers are asking them to reach CMMI maturity level two or even level three!

There is a perception in the software industry that the Capability Maturity Model Integration is a heavy, burdensome process model only meant for large scale software development in the defense, aerospace, and pharmaceutical verticals.

This just isn't true!

The CMMI is a perfect fit for agile organizations whether they are engineering or IT, embedded or workstation, and internal or commercial. The key to success isn't "what" is in the model, but "how" it is interpreted and used.

"The Secrets of CMMI for Agile Organizations" begins with a history of how the CMMI came to be perceived as burdensome, and walks the attendee through a set of concrete recommendations, methods, and timelines for developing light, useful processes for agile software development.

Join Jeff Dalton, President of Broadword, SCAMPI Lead Appraiser, CMMI Instructor, author, and software engineer as he explodes the myths about CMMI (*yes, it's Agile*), about process deployment (*no, it doesn't take extra time*), and management sponsorship (*yes, they have to change too*).

Excerpted from the content of his book, "Agile CMMI," Jeff will discuss practical real-world approaches to CMMI adoption and organizational change with ideas that you can use today to make your process initiative more successful.

Bio: Jeff Dalton is an author, CMMI Lead Appraiser, CMMI Instructor, and SPI consultant with over 25 years in the software engineering industry. His client list includes NASA, Ford Motor Company, Johnson Controls, DaimlerChrysler, Detroit Edison, and Hitachi, where he helps IT and Engineering organizations become more competitive in the global marketplace. His experience includes more than a decade with Ernst & Young as well as shorter stints with EDS, Hewlett-Packard, and the RL Polk Company. Jeff lives in S.E. Michigan with his family where they are building an airplane in their garage.

[Back to top](#)

Session D3:Speaker: Santosh Kalaskar - Automation Framework for Validating Web Based Applications

Summary: Learn how the proposed automation framework can be used to validate web applications which need flexible and scalable automation solutions. This framework dwells in detail on the syntax and semantics which can be easily customized to fulfill most of the automation needs. Using this framework in a multi-layered automation architecture simplifies things to such an extent that the automation can be driven through with a single script residing in any one layer. For an easy reference an empirical case study has been provided with full details of the implementation.

Abstract: Commercial Web based applications in general have 3-tier architecture. Validation of the 3-tier architecture depending on the complexity can be a challenging proposition. The proposed Automation Framework helps ease the task of setting up a test framework which will support such type of architecture. The framework will have specifications for the syntax, semantics and the interfaces between framework and its components. This framework allows circular comparison/ validation of data between the front end, backend (UNIX flat files, etc) and databases. The framework provides support for front end data validation through a GUI testing tool while the backend and database data validation can be accomplished through a combination of scripting languages and SQL. The main USP of this framework is its flexibility. The framework allows addition/ deletion of any number of layers. The approach is demonstrated with an empirical case study.

Bio: Santosh Kalaskar is currently working for Yahoo! India as a Quality Engineering Lead. He has a total experience of around 7 years in Development as well as Quality Assurance. He has completed his post graduation in Computer Science from University of Toledo, OH. He is very much into exploring new trends and technologies related to the internet.

[Back to top](#)



Session D4: INVITED SPEAKER: Mark Paulk - The Evolution of the Continuous Representation for Process Capability

Summary: A continuous representation model characterizes a set of processes by a structured set of process attributes, typically organized into six Capability Levels. The continuous representation is a flexible and easily extended architecture, so understanding its evolution can be helpful to those wishing to use a continuous representation model or to adapt it for different environments. The capability dimension in the reference model in ISO/IEC 15504, and the variants developed for the Systems Engineering Capability Maturity Model (SE-CMM), the Software CMM, and CMM Integration (CMMI), capture a tradeoff between reliability and usability that remains a challenge for model developers today.

Abstract: A continuous representation model characterizes a set of processes by a structured set of process attributes, typically organized into six Capability Levels. The continuous representation is a flexible and easily extended architecture, so understanding its evolution can be helpful to those wishing to use a continuous representation model or to adapt it for different environments. The reference model in ISO/IEC 15504 includes a “continuous” process capability dimension that has evolved to address conflicting requirements and that has been adapted for a variety of other improvement frameworks, including the Systems Engineering Capability Maturity Model (SE-CMM), the Software CMM, and CMM Integration (CMMI). The evolution of the continuous representation in ISO 15504, and the variants developed for other frameworks, capture a tradeoff between reliability and usability that remains a challenge for model developers today. The purpose of this paper is to describe the evolution and the important variants of the ISO 15504 continuous representation with a discussion of the objectives and tradeoffs associated with each.

Bio: Mark Paulk is a Senior Systems Scientist at the IT Services Qualification Center at Carnegie Mellon University, where he works on best practices for IT-enabled services. From 1987 to 2002, Mark was with the Software Engineering Institute at Carnegie Mellon, where he led the work on the Capability Maturity Model for Software. Mark’s research interests revolve around high maturity practices, statistical process control, and agile methods. Mark received his PhD in industrial engineering from the University of Pittsburgh. He is a Senior Member of the IEEE, a Senior Member of the ASQ, and an ASQ Certified Software Quality Engineer.

[Back to top](#)

Session D5: Exhibitor Presentation Cognence

[Back to top](#)

4:15 p.m. – 5:30 p.m. ICSQ 07 Panel Discussion - Introducing Agile Methods into Traditional Environments

Panel Leader: Scott Duncan

Panelists:

- David Kane (SRA)
- Stephen B. Ornburn (GBC-Group, Inc.)
- David E. Percy (Sandia National Laboratories)
- Ken Schwaber (Control Chaos)
- Richard Turner (Systems and Software Consortium)

Agile software development methods have been reported to achieve some impressive results, but seemingly in very small projects of low risk and complexity using technologies associated with e-commerce. Many software quality professionals work in environments where this is not the typical software product profile and where “traditional” methods, organizational structures and assessment expectations exist (e.g., regulated and/or safety-critical environments). This session will

present experiences of several individuals who have applied agile methods in a variety of “traditional” software project environments.

[Back to top](#)

Join the ICSQ 2007
Sponsors & Exhibitors
Tuesday Evening 5:30 p.m. - 7:00 p.m.
Reception in the Exhibit Area
Food - Prizes - Networking

Colorado Section's Joint Meeting
Tuesday, October 16, 2007
7:00 p.m. - 8:00 p.m.