In-the-SPIN

Newsletter of the Boston SPIN

Issue 39 Winter 2000 - 2001

Editor: Carol Pilch

Editorial

This issue of *In-the-SPIN* provides you with summaries of the November, December, and January meetings. Here's a chance for you to catch up on what our speakers, Alex Marchicelli, Capers Jones, and Alan Davis had to say if you were unable to join us at any of these meetings. Also in this issue, you will find a SPIN Perspectives article provided by Johanna Rothman, a frequent contributor to *In-the-SPIN*.

I'd like to take this opportunity to make announcements about the newsletter editorship and committee. First, I'd like to let our readers know that I will be stepping down as editor after the next edition. I'm very pleased to announce that Judi Brodman and Sheila Lynch have volunteered to share the responsibility as co-editors. They've already started to get involved with the production of this issue. Second I'd like to identify and thank our recent newsletter committee volunteers: Martin Stankard, Nancy Van Schoenderwoert, and Carol Govoni. Martin provided the summary of the January meeting.

If you're a reader of this newsletter, the Boston SPIN would greatly appreciate your feedback. The Boston SPIN, and in particular the editor of *In-the-SPIN*, would like to know if the readers' expectations are being met. The SPIN steering committee also encourages broader participation in the content and production of the newsletter and also in the SPIN organization. Send letters-to-the-editor, quips, quotes, anecdotes, articles, offers to participate in the newsletter committee, and general correspondence to Carol Pilch, carol.pilch@GD-CS.COM.

Boston SPIN
Since January 1993

Software Process Improvement Network

SPONSORS:

General Dynamics Communication Systems

Raytheon Company

Edelman & Associates

We thank the Computer Science department of **UMASS- Lowell** for providing support and hosting our Web page

SPIN Perspectives

This issue's *SPIN Perspectives* column features an article by Johanna Rothman, Rothman Consulting Group, Inc. Johanna discusses why your staff members may be resisting reviews.

"You Don't Need to Review That"

In *Star Wars*, Obi-Wan Kenobe speaks to some storm troopers using the Force: "These aren't the droids you're looking for." When I was a software developer and tester, other developers would tell me in their best Force voice "You don't need to test that. That part is okay." (Sometimes they even waved their hands as Obi-Wan did.) Now that I coach review and inspection teams, they sometimes say, "You don't need to sit in on this review." When I assess organizations, my contacts sometimes say, "You don't need to see that project."

When I hear "You don't need to…" that's my clue that I should start there. Why?

- They can't admit this piece is broken. Maybe they can't talk about it, or they just can't admit that they could have developed something broken. On a recent engagement, the developers took tremendous pride in their technical abilities. They were having trouble with product performance, so one developer, Ned, ran the performance tests to analyze the data. I asked Ned if he wanted some review on the data. He said "Of course not, I can figure this out myself. You don't need to review this with me. No one needs to review this with me." Maybe he could, but after a week, he hadn't gotten close to describing the problem.

Continued on next page

IN THIS ISSUE . . .

- They can't see that the part that doesn't need review is broken. It's so much a part of the organization that they don't know why they do it anymore. One client made some tough decisions about how to organize and focus their product development work. They cut a number of products from the list of active products. My contact, Nancy, insisted on supporting one of those products, although it took time away from the products the company had committed to. When I reminded Nancy that the company had stopped supporting the product, she said "But we've always supported this product." Nancy didn't think she needed to review her list of projects, but her list of projects was broken.

- It's broken and they can't fix it or don't know how. They know they should look at it and fix it, but they can't imagine how to fix it. One of my clients migrated a system from a workstation environment to a client/server environment. Their management did not provide training in the new client/server environment. When they ran into trouble, they couldn't solve it, because they literally did not have the expertise.

- They're in a hurry because the piece is behind schedule. Of course, the piece is probably behind schedule because they've been having trouble with it. So, being behind schedule is a reason *to* review, not a reason to avoid review. Our experience tells us that virtually all the error-prone pieces of code are those that skipped reviews and tests because they were behind schedule.

- They sincerely believe it's not broken, so they've never really looked at it when trying to account for system problems. Thus, it's likely to be the least looked-at part of the system, and thus the least reliable.

When I hear the Force, I step back and analyze what's going on. When I hear resistance to my suggestion, I think about what people might be afraid of. My colleague, Dale Emery, taught me that resistance is concern about loss. What will these people lose if other people review their work? In many cases, they will lose face, or some belief in their own capabilities, some facet of their self-esteem.

You can't improve your product development process if you do things that hurt people's self esteem. You can review pieces of development in considerate, non-damaging ways. Consider other people's concerns, and use your Force to uncover what to review.

Johanna Rothman observes and consults on managing high technology product development, looking for the leverage points that will make her clients more successful. You can reach her at jr@jrothman.com or by visiting www.jrothman.com.

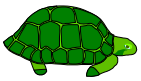
Meeting Summaries

Notes from the November Meeting

Contributed by Carol Pilch, Senior Member of Technical Staff and SEI Authorized Lead Assessor, General Dynamics Communication Systems

Title: Time to Market Process Improvement – A Case Study

Speaker: Alex Marchicelli, NewLane, LLC



The November SPIN speaker, Alex Marchicelli provided the SPIN members with a case study that explores the strategies used in a real-world scenario to shorten time-to-market and achieve process improvements. The basic question to be answered was: Is product testing taking too long, slowing down the company's ability to go to market quickly? The overall objective of this project was to reduce testing cycle time while preserving the quality of service delivered to customers. More specific objectives were to:

- Challenge existing assumptions about testing to break paradigms,
- Validate from external benchmarking perspectives, if testing is taking a "long" time, and identify creative approaches to testing, and
- Identify short-term process improvements that will have a positive impact on reducing cycle time.

A four-part formal strategy was used to address the specific objectives. First an assertion/hypothesis methodology was applied. Hypotheses relative to the objectives were formulated and then proven to be right or wrong. Second, assessments consisting of interviews, questionnaires, and workshops were conducted to collect quantitative data to determine if the hypotheses were supported and to determine corrective action. Third, benchmarking was performed to gather data and compare it against industry norms or against its own history. Fourth, performance models used on a balanced scorecard approach and based on customers, learning and growth, internal processes, and financials were used to assess and continuously measure performance.

Some of the findings resulting from applying the four-part strategy are:

- Achieving accelerated testing cycle time will require innovation and significant changes to how work is performed and managed *across* organizations.
- Changing work patterns across organizations to drive additional cycle time improvement will require senior management support, shared goals across functional departments, and an end-to-end process owner who is responsible for delivering results.

Continued on next page

- A "performance "model which identifies the required pathways for obtaining testing process improvement results.
- A number of performance pathways that include key objectives and strategy for realizing benefits.

The key objectives identified in the performance model were:

- Streamline testing process,
- Accelerate release decision making,
- Manage testing as end-to-end process,
- Transform development process,
- Develop new approach for managing vendors, and
- Accelerate customer leaning and product innovation.

All of which results in a reduction in cycle time, and applying this to the real-world situation demonstrates up to 7 weeks cycle time reduction with potential for an additional 4 week cycle time reduction.

Notes from the December Meeting

Contributed by Carol Pilch, Senior Member of Technical Staff and SEI Authorized Lead Assessor, General Dynamics Communication Systems

Title: Software Benchmarking: What Works and What Doesn't?

Speaker: Capers Jones, Chief Scientist, Software Productivity Research, an Artemis company

Capers Jones' presentation provided lots of specific and interesting data and, in particular information on organizational software benchmarking.

To begin, twelve criteria for measuring benchmarking success along with seven benchmarking hazards were provided.

The benchmark data should:

- ❑ Benefit the executives who fund it,
- ❑ Benefit the managers and staff who use it,
- ❑ Generate positive ROI within 12 months,
- ❑ Meet normal corporate ROI criteria,
- ❑ Be as accurate as financial data,
- ❑ Explain why projects vary,
- ❑ Explain how much projects vary,
- ❑ Link assessment and quantitative results,
- ❑ Support multiple metrics,
- ❑ Support multiple kinds of software,
- ❑ Support multiple activities and deliverables, and
- ❑ Lead to improvement in software results.

The benchmarks should not:

- ❑ Conceal the names of projects and units,
- ❑ Show only overall data without any details,
- ❑ Omit non-coding activities,
- ❑ Omit "soft factors" that explain variances,
- ❑ Support only one metric such as LOC,
- ❑ Omit quality and show only productivity, and
- ❑ Be used to set ambiguous or abstract targets.

Function points are the major metrics for software benchmarks. Metrics include productivity, schedules, cost, or quality. Software Productivity Research (SPR) has been performing assessments and benchmark studies since 1985. SPR data now exceeds 10,000 projects representing more than 600 enterprises. Benchmarks are categorized by software classes: systems software, information systems, outsource vendors, commercial software, and military software.

Capers Jones' presentation included the factors causing significant variations in software quality and productivity. The top ten ranked factors for best-case results are:

- ❑ Experienced managers,
- ❑ Experienced staff,
- ❑ Achievable schedules,
- ❑ Stable requirements,
- ❑ Structured methods,
- ❑ Formal inspections,
- ❑ Planned testing,
- ❑ Automated estimates and plans,
- ❑ Careful milestone tracking, and
- ❑ High design/code reuse.

Eight attributes of best in class companies are:

- ❑ Good project management,
- ❑ Good technical staffs,
- ❑ Good support staffs,
- ❑ Good measurements,
- ❑ Good organization structures, good methodologies,
- ❑ Good tool suites, and
- ❑ Good environments.

Attributes of good project management are:

- ❑ Fairness to staff,
- ❑ Desire to be excellent,
- ❑ Strong customer orientation,
- ❑ Strong people orientation,
- ❑ Strong technology orientation,
- ❑ Understands planning and estimating tools,
- ❑ Can defend accurate estimates to clients and executives,
- ❑ Can justify investments in tools and processes.

The December presentation concluded with a summary of software benchmark and improvement plan Do's and Don'ts.

- ❑ **Do** think long range (i.e., 3 to 5 years), consider all factors (management, process, tools, organization, skills and training, programming languages, environment), plan expenses of up to $15,000 per staff member over 3 years, and consider your corporate culture.
- ❑ **Don't** expect immediate results, concentrate only on process or any other "silver bullet," expect major improvements for minor expenses, ignore resistance to change.

Notes from the January Meeting

Contributed by Martin Stankard, President Productivity Development Group, Inc. Process Consulting and Training, Westford, MA. Martin is a member of the SPIN Newsletter Committee.

Title: Aligning Project Management Goals, With Corporate Goals

Speaker: Alan M Davis

Alan M. Davis, President of Omni-Vista, Inc. gave a spirited presentation on aligning software development programs with overall corporate goals and producing profitable revenue. Alan opened with two problems: (1) the last fifteen years of attention to software process improvement has made software better, but later; too often we forget time to market, and (2) even processes at CMM level 5 can produce software that no one wants, while many firms operating at level 1 sell lots of software.

Davis said that the standard engineering model that starts with a problem, visualizes a solution, then picks detailed features and comes up with a working solution only works under perfect resourcing, timing and other conditions. Real projects exist in a business context that often includes too little time and resources, ever-changing problems, poor communication, and pressure to meet cost and revenue goals. Davis cited a study by Don Reinertson that found 31% of software development projects cancel before completion; while 40% of products completed are never used, for an overall failure rate of 71%.

Davis went on to state that changing conditions and poor up front planning cause most of these failures. For example, the Motorola Iridium communications network was a great plan, but market conditions changed, making it unsalable. Poor product planning covers a list of sins, ranging from overestimating the demand for the product to picking the wrong requirements to build into the product. The need to juggle many factors and complex interrelationships among them makes it hard to start a new software product off on the right foot and keep it moving in the right direction.

Davis laid out a conceptual approach to solving the problem (he called it a "think piece, not a prescription") that recognizes there is no perfect answer. The approach models each requirement or resource in terms of its impact on other variables, such as market penetration, revenue, cost, or time to market. He uses a decision support system to juggle all these factors in order to start a project with:

- Resources, Schedule and Features (requirements)

Compatible with:

- Acceptable market risk
- Market size and penetration goals
- Price, Revenue, Profit and Breakeven goals

Davis calls this thinking process a "Triage" and described how it begins with the customer's view of the situation. He says without understanding the customer's pain, for which you have a cure , NO product. The triage process tries various combinations of features or requirements, plays them against the market estimates and then converts them into a value proposition (from your company's perspective) that outlines how you will make money on the product. This value proposition is based on your track record doing projects in terms of productivity, quality and cost.

Davis cited Bosworth's book on Solution Selling as a good source on how to take the customer's perspective. He feels that project execution is also well explored. That leaves a focus on how you create your initial value proposition and then manipulate it to find the sweet spot that balances market and customer need with company profitability.

Davis described a high level process in which you begin with an Excel or Access matrix of requirements. For each requirement the database includes estimates or ratings of customer importance, effort to develop, duration of development, technical risk, and tentative decisions about in which release to include each requirement. Next, you sort the requirements from the most important to the least important to stakeholders (customers and users) and cumulate their total resource requirements. You "draw the resource availability line" on your sorted list and see if essential requirements are included above the line. If not, you either add resources or you do not build the product.

Assuming that you have enough budget to build an initial version of the product with enough features to attract customers, you now enter a second phase of triage that looks at the implications of different sets of features. This means looking at the balance of requirements, cost, and schedule. Davis uses historical software development data to show how adding and deleting requirements affects development effort and first ship date of the product under development. The higher the cost, the lower the profitability. The greater the number of requirements included in the specification, the later the first ship date of the product, and the longer your company must wait for revenue.

Davis then illustrated several tradeoff studies with example data from his firm's decision support and planning tool. The tool, which could be duplicated with a spreadsheet, translates the selection of requirements for inclusion into the product into a forecast of cash in and out flows as well as a time to breakeven. In effect it provides a business simulation showing how the inclusion of more or fewer features adds time, market attractiveness, cost, and risk to a software product development effort. In the triage process that Davis depicts, a product manager would simulate various configurations of the product, and test the sensitivity of the market to price and features until he or she arrives at a satisfactory balance of developer, customer and company goals.

Davis closed with the suggestion that once a product-planning tool such as he outlined is built, a product planner can use it to convert product planning into a win-win game against the tool. To do this, Davis says you bring marketing, development, finance, and customers together to play with the decision support system as a game. They all play against the game instead of each other. Those interested in contacting Al Davis can reach him at (710)-9556664 X101.

Information about Upcoming Meetings

by Anna Allison, Program Chair

March Meeting Announcement

Topic: Four R's of Software Process Improvement: Requirements, Reviews, Retrospectives, and Results

Speaker: Johanna Rothman, Rothman Consulting Group, Inc

When: Tuesday, March 20, 2001
6:30-7:00 Networking and Round Tables
7:00-7:10 Announcements
7:10-8:10 Featured Speaker
8:10-8:30 Questions and Answers

Who: Everyone (Academia, Government, Industry)

Location: General Dynamics, 77 "A" St., Needham MA.

Abstract: Process improvement projects can be difficult to start, keep on track, and assess results. We can use the same requirements gathering and specification techniques that we use on product-projects on our process improvement projects. In this talk, Johanna will discuss how to define requirements for process improvement projects and how to use reviews and retrospectives to assess the results of process improvement efforts.

Johanna will discuss: how to define process improvement requirements; how to move from mandates (Level 3 or bust!) to actual problems to be solved; the roles of reviews in process improvements, and using results and retrospectives to assess your work.

About the Speaker: Johanna Rothman observes and consults on managing high technology product development. She works with her clients to find the leverage points that will increase their effectiveness as organizations and as managers, helping them ship the right product at the right time, and recruit and retain the best people.

A frequent speaker and author on managing high technology product development, Johanna has written articles for *Software Development, Cutter IT Journal, IEEE Software, Crosstalk, IEEE Computer, Software Testing and Quality Engineering, Catapulse, StickyMinds.com,* and publishes *Reflections*, an acclaimed quarterly newsletter about managing product development. Johanna is the founder and principal of Rothman Consulting Group, Inc., and is a member of the clinical faculty of The Gordon Institute at Tufts University, a

practical management degree program for engineers. Johanna leads workshops in the areas of project management, software quality, and software management.

SPIN Roundtables: Roundtables are focused group or "birds-of-a-feather" discussions, with a facilitator, to stimulate and moderate discussion. Roundtables are held during the Networking portion of the SPIN meeting. See our web page, http://www.cs.uml.edu/Boston-SPIN to see which topics are selected for this SPIN meeting.

Directions: From Route 128 in Needham, take exit 19A onto Highland Avenue East. Take your first right by the Ground Round and take your second left onto "A" Street. General Dynamics is the last building on the right. Enter the parking lot by the General Dynamics sign and come into the building by the cafeteria entrance, which is located to the left of the main entrance. There will be a security guard at the entrance. See http://www.gd-cs.com/needham.html for directions.

Info: See our web page, http://www.cs.uml.edu/Boston-SPIN For SPIN info, contact Anna Allison, anna_allison@yahoo.com

Cancellations (including weather cancellations): We will notify the membership via email to the SPIN distribution list, post the notice on the SPIN web page, and send the cancellation announcement to Channel 7 TV and radio, WRKO AM 680 starting at 3pm.

SPIN '00-'01 Program and Speaker Schedule as of 02/14/01

Date	Speaker/Topic
Mar. 20, 2001 @ General Dynamics	Johanna Rothman "The 4R's of Software Process Improvement"
Apr. 19, 2001 Thursday! @ TBA	Joint ASQ meeting Speaker: Rob Peck
May 15, 2001 @ General Dynamics	Steve Rakitin "Management's Role in Achieving Predictable Software Development:
June 19, 2001 @ General Dynamics	Unmesh Gundewar "Ensuring Clients Achieve Superior Value"

Looking for Interesting Speakers

We are always looking for interesting speakers. If you'd like to speak at Boston SPIN, please review these criteria before sending us an abstract.

Speaker Criteria:

1. The topic must be timely, an issue of concern to our membership.
2. We want to hear about real-world topics. If you have an academic topic, we're interested in how it applies to the real world.
3. If you are interested in creating a panel, please write an abstract, and suggest at least one panelist. We can work with you to find other panelists.
4. The topic should either explain how to *do* something, or bend our brains in another direction.
5. The presenter should be intimately involved with the "hows" of the thing that got done.
6. We are not interested in sales pitches.

If you have information you'd like us to hear, please send an abstract to Anna Allison, anna_allison@yahoo.com.

We developed a speaker checklist so that none of us would have to rely on our short-term memories. Please use the checklist to prepare for your SPIN talk.

Speaker Checklist:

1. 60 days in advance of meeting deliver: 2-paragraph abstract, one paragraph bio, and picture to anna_allison@yahoo.com

2. Within one week of meeting date: If desired, email copy of paper or overheads to heimann@world.std.com so that it is downloadable from the SPIN web page.

3. At the meeting: Speaker provides one copy of overheads to Linda McInnis for our library.

4. Optional, but highly desired: Send a copy of overheads, paper, etc. for our web page as of the day of the meeting. If possible, provide 50-60 copies of overheads at the SPIN meeting. (The attendees really appreciate this.)

5. Optional: If you've written a book and are willing to donate it to SPIN, we'd be happy to offer the book as a door prize by conducting a free drawing.

Boston SPIN

The Boston SPIN is a forum for the free and open exchange of software process improvement experiences and ideas. Meetings are usually held on third Tuesdays, September - June. Boston SPIN welcomes volunteers and sponsors. There is no charge to attend the meetings.

For more information about our programs and events contact:

Linda McInnis
80 Harris Street, Suite 2
Acton, MA 01720
Telephone: (978) 635-9281
Email: Boston_SPIN@yahoo.com

For information about SPINs in general including ***HOW TO START A SPIN*** contact:

Dawna Baird of SEI (412) 268-5539, dbaird@sei.cmu.edu, http://www.sei.cmu.edu/collaborating/spins/spins.start.html.

IN THE SPIN is available on our Web page:

http://www.cs.uml.edu/Boston-SPIN.

TO RECEIVE NOTIFICATION OF NEW IN-THE-SPIN ISSUES and Boston SPIN specific notices send email addressed to withall@mediaone.net.

We have 2 separate email lists: one for this newsletter and one containing announcements that we receive from other process organizations and forward out.

IF YOU WANT TO JOIN THE ANNOUNCEMENTS LIST: http://groups.yahoo.com/group/Boston-SPIN-Announcements/join.

Send letters-to-the-editor, and general correspondence to Carol Pilch, carol.pilch@GD-CS.COM.

Send job postings to heimann@world.std.com.

Back issues and other information about Boston SPIN can be found at our WEB HOME PAGE:
http://www.cs.uml.edu/Boston-SPIN/