Expectations of the workshop

1. Where should SPIN go in the future?
2. Meet likeminded people
3. Identify the pitfalls (including senior management)
4. Introduce concepts - PI and QA
5. Mechanical PI applied to software
6. Cultural aspects
7. Reenergise change
8. Re-emergence of agile and SPI (throw away processes)
9. What’s new and reacquaint
10. Tips about what works
11. General knowledge
12. Adopt theory - reality
13. Is there anything new? What really is new?
14. How can you define processes that please everyone?
15. Hints and tips
16. Different cultures - people, processes
17. Disillusioned with CMMI - how does the BCS feel about it?
18. CMMI update

Introduction to CMMI

CMMI does work if you do it correctly.

Anecdotally, 80% CMMI implementations fail on the 1st attempt, but there is always some learning and improvement despite not total failure.

Reasons for failure:

- Don’t understand why doing it
- Theory before business objectives
- No leadership
- No measurement

<table>
<thead>
<tr>
<th>Level</th>
<th>Degree of Cultural Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Baseline</td>
</tr>
<tr>
<td>1-2</td>
<td>Massive and the key cultural change</td>
</tr>
<tr>
<td></td>
<td>- Requires leadership and direction (policy)</td>
</tr>
<tr>
<td></td>
<td>- Measurement &amp; analysis to support policy</td>
</tr>
<tr>
<td></td>
<td>- QA - check that you are doing it</td>
</tr>
<tr>
<td></td>
<td>- What works, succeeds and fails</td>
</tr>
<tr>
<td></td>
<td>- Share best practice</td>
</tr>
<tr>
<td></td>
<td>- People improvement - practices, performance, people management</td>
</tr>
<tr>
<td>2-3</td>
<td>Small cultural change</td>
</tr>
<tr>
<td></td>
<td>- Measurement, QA, processes - don’t see benefits until level 3 and this can be a</td>
</tr>
</tbody>
</table>
Level | Degree of Cultural Change
--- | ---
 | problem when trying to educate senior management  
• QA personnel are required on a full time basis
3-4 | Large cultural change - need a culture that allows people to challenge management on a continual basis  
• Key practices  
• Statistical control on 50 - 100 projects  
• Focus improvement on what is important  
• Focus on education and lessons learned  
• Every process measured on the company's vision/strategy
4-5 | Small cultural change

Leadership - vision and policy  
Management - day-to-day - listen to people and remove impediments  
QA - to support project management

Processes - if people feel it works, write it down  
Capture lessons learned - what you couldn't do it correctly the 1st time around

The most important aspects of a CMMI programme are:

1. Education and training - it should be perceived as a reward  
2. Peer reviews. Look for the good practices as well as finding defects.

**SPI Rules**

*SPI Rules - Dr Clifford Shelley*  

*Energizing CMMI - Dr Clifford Shelley*

**The Ten Rules of SPI**

May 2009: Recently we have been involved in several discussions where people (including us) have expressed discontent and concern about the current state of software process improvement (SPI). It has prompted the drafting of a list to capture the essence of ‘good’ SPI. Our list of rules looks like this at the moment:

In no particular order...

1. Concentrate on fixing real problems getting in the way of business goals - if you aren't have a d****d good reason.  
2. Improvements are owned by those affected by them, i.e. those that use or perform the affected activities.  
3. Work on improvements with those who need problems solved or improved performance, or those keen to try new ways of working – especially in the early stages where progress can be slow and success uncertain.  
4. Require rapid feedback (results) on the effect of changes (solve lots of small problems fast)...
...and evaluate (measure and analyse) them, and then act on them.

5. Use a model to provide a conceptual framework and scope if you want (actually experience shows that two are better), and know how to use it, and who's in charge - don't let model compliance become the primary objective.

6. Don't manage SPI as a project.

7. Measure progress by results, not schedule.

Workshop Questions

Measurement - how do you measure it?
If measure wrong things, then you destroy it.
Difficult to measure quality
Whatever you measure, you change behaviour
No direct relationship between process and product

Suggestions

Robert Austin - How to Measure Performance

Measure provides visibility and is a critical tool to improving understanding.
GQM - reason for measuring and is in the latest version of CMMI
Record cost of good and bad quality
Measures of efficiency and productivity are damaging - the best drivers are measures of value and effectiveness but these are not easy to measure

8. Tactics determine strategy – that is, strategies are valueless until you know what you can actually change in practice.

9. SPI is exploratory; some, many even, improvements will fail. But these failures are offset by what you learn and those improvements that do work well (and a few that work spectacularly well).

10. SPI must pay for itself. Demonstrate this or stop.

What's in it for me? Applied all the way down from senior management
E.g. CEO asked for progress reports - this was very motivating
Need to understand business and outcome expectations

Initially, focus software process improvement on fundamental processes such as QA, CM and RE.

Rule 11. Document processes, practices and activities 'as is' and use this to establish a credible process baseline which can then be developed. (Do not - ever - document processes as you would wish them to be and attempt to introduce, or roll them out.)
Workshop Questions
How do you get funding for this?
Evolutionary versus revolutionary
Culture interferes with scientific approach
How do you deal with conflict?

Suggestions
Focus on key processes and model them at a high level
Study root cause analysis of major problems and address those
Don’t tell senior management! (Level 2)
What’s in it for me?
Understand and have some apathy for the culture of the organisation
Record what, why and measurements

Rule 12. Focus software process improvement on fundamental software development capabilities initially (managing quality, configuration management, requirements engineering... (thank you CMM)), then when these are in place, but not before, align development capability with business and customer needs.

This is essentially a requirement to learn to walk before trying to run. This seems to conflict with rule 2. Not sure how to resolve this at present. Perhaps it is a matter of putting the tools in place before you can use them (to meet business needs).

Can Process Make You Happy?

What People Need

Want productivity and creativity
People doing the work should be happy and having fun together because they produce better work

People need:
- Living wage
- Resources and the tools to do their job
- Skills, competencies and knowledge
- Continuity - stable organisation
- Stability - constant priorities
- Plan - how does what I do fit into the bigger picture?
- To see the finished product
- Vision and end objective - what will the organisation look like in 5 years time
- Shared vision - "our vision"
- Pleasant and private working environment (Open plan offices reduce productivity by 20-25%.
  It takes 20 minutes to regain concentration after an interruption.)
- Leadership - to inspire and to follow
- Team - to support and includes all stakeholders including senior management. Support - help me along and keep me focussed.
- Control - need to be able to flag up risks, issues, problems and deviations from plan.
- Recognition - not just management but also customers. Recognition and reward to hand in hand. It can also be public recognition, such as being given the time to write an article for a publication.
- Feedback - How am I doing on an on-going basis and not just one a year
- Trust - two way
- Eustress - positive stress
- Freedom to work within a framework - given everything you need to do your job correctly

<table>
<thead>
<tr>
<th>Acceptance</th>
<th>Measurements</th>
<th>Know it</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ability</td>
<td>Direction</td>
<td>Resources, tools, models</td>
</tr>
<tr>
<td>Aspiration</td>
<td>Vision</td>
<td>Role models</td>
</tr>
<tr>
<td>Attitude</td>
<td>Behaviour</td>
<td>How do you react when things go wrong? Only way to change is through leadership - the managers must do it first.</td>
</tr>
</tbody>
</table>

**Trust**
Lack of trust affects creativity and communication
Without trust, measurement is not real
Demonstrate trust in small steps.
Start to build trust slowly. E.g., triangular peer reviews - 1st week, 1st person, 2nd week, 2nd person, 3rd week, 3rd person. Record time taken and defects anonymously.
In the current environment of constant reorganisation it is easy for trust to breakdown.
Need to recognise who you can trust?

**Working with 3rd Parties**
Verify all products and change requests
However, need to get own ship under control
Difficult relationship if mismatch of maturity
Customer always evaluates supplier, but should the supplier evaluate the customer if they are the more mature organisation???

**SPI Timeline and the Future**

<table>
<thead>
<tr>
<th>Date</th>
<th>Who</th>
<th>What</th>
</tr>
</thead>
</table>
| 1817 | Thayer | US Military Academy  
Gave marks to exams so that they could be ordered. Previously only pass or fail.  
Also introduced progress reporting. |
| 1832 | Tyler | Quality control of armaments |
| 1839 | Whistler | Rail roads  
Quality control to line manager |
### Date | Who | What
---|---|---
1911 | Taylor | Scientific management
1920 | Shewhart | Plan - Do - Check - Act
1930-40s | Deming | Quality management
1950s-60s | | TQM
1970s | Crosby | Application of TQM to the software industry
1980s | CMM | CMMI, Agile

**Agile**

Developed as a reaction to project management  
Widespread and misunderstood  
No common definition of what it is

A structured approach to project management for SCRUM - [Agile Project Management](#) - Ken Schwaber

**Issues for the Software Industry**

- Approach and the environment depends on the context
- Can use a risk profile to help determine the approach - [Risk Assessment Template](#) - Niwot Ridge Consulting
- Outline approach in the Quality Plan and say why you are adopting a particular approach
- The role of a SPI specialist should be to know what tools are available and to choose the right one based on the business objectives
- Abbot and Fisher have examined processes and culture - [The Art of Scalability](#) - Abbot and Fisher
- CMMI - focus on processes
- Agile - focus on people
- Software development competencies across the profession will emerge in 5-10 years
- Weinberg examined the psychology of computer programming and his conclusions are still relevant today - [Psychology of Computer Programming](#) - Gerald Weinberg
- Properly defined outputs and quality gates help to reduce egos and mavericks

**Principles for Success**

Definition of Quality - meeting your customer's expectations

- [Kotter's Principles of Change](#)
- Empowerment - freedom within the framework
- Real improvement at least every 3 months
- Determine business reason, value and the cost
- Risk management - most senior executive to present where we are in 5 years if this fail,s and the next most senior executive to present where we are in 5 years if it succeeds
- Urgency - relates to the cost of failure, waste and rework. Cost of quality and the risks.
- Vision - what will the organisation look like in 5 years time
- Quality - satisfied customers
- Sponsor - budget, needs and the required result
- Process improvement requires constant investment and reinforcement
- Change in culture - listen to people doing the work. Specialists tell you how they do their job.
- Respect the past - successes and failures
- Start by focusing on early adopters

Top Five SPI Tools

Top 5 SPI Tools - Dr Clifford Shelley