CISC 327 - Software Quality Assurance

Lecture 4

Software Process Evaluation
Software Process Evaluation

• How can we measure software processes?
  – Today we look at methods for evaluating and improving software processes, regardless of which process is being used
  – There are several methods and standards for software process evaluation
  – Most are aimed at improving existing development processes as they are applied, calling maturing them
  – Idea is that as a company or team gains experience with a process, they continually improve it to make it better in their use
The Defect Prevention Process

• DPP - Defect Prevention Process
  – DPP is not itself a software development process, but rather a process for continually improving the development process
  – Modelled on quality assurance techniques used in Japan for decades
The Defect Prevention Process

• Based on three simple steps:
  – Analyze existing defects or errors to trace their root causes in the process (i.e., how they were missed)
  – Suggest preventive actions to eliminate the defect root causes from the process
  – Implement the preventive actions to improve the process
The Defect Prevention Process

- **Formal DPP Reviews**
  - First used at IBM Communications Programming Lab (1985)
The Defect Prevention Process

• 1) Defect Causal Analysis Meeting
  – At end of each stage of development, review and analyze defects that occurred in that stage in a short meeting
  – Developers trace root causes of errors, suggest possible actions for preventing similar errors in future

• 2) Action Team
  – Action team has cross-organization members
  – Evaluates suggested actions, initiates actions across the organization, including development team actions
The Defect Prevention Process

• 3) Process Improvement Team
  – Members of the development team
  – Implements process changes and provides advice for next stage of development

• 4) Stage Kickoff Meeting
  – Development teams meet to review process changes and re-emphasize focus on quality
DPP Applied to Waterfall Model
Process Quality Standards

• **Software Process Assessments and Standards**
  – There are two kinds of process quality standards
  – **Maturity Models**
    • Maturity models attempt to measure how well developed (mature) the software process in a particular organization is, and thus how likely it is to produce quality results
  – **Certification Standards**
    • Certification standards measure an organization's software process against a defined standard, and certify the organization if its process meets the standard
Capability Maturity Model (CMM)

• The SEI Process Capability Maturity Model
  – CMM defines a five-level scale of process maturity, and an organization's software process is assessed as "CMM-1", "CMM-3", "CMM-5" indicating it level on the scale
  – Used by government agencies and companies in the U.S.
  – Assessed using an 85-item questionnaire
Capability Maturity Model (CMM)

• CMM Level 1 - "Initial"
  – Characteristics: chaotic; unpredictable cost, schedule, and quality

• CMM Level 2 - "Repeatable"
  – Characteristics: intuitive; cost and quality highly variable, reasonable control of schedules, ad hoc methods and procedures
  – Key elements: requirements management, project planning, configuration management, quality assurance procedures
Capability Maturity Model (CMM)

- CMM Level 3 - "Defined"
  - Characteristics: qualitative; reliable costs and schedules, improving but unpredictable quality
  - Key elements: process definition and improvement, training program, integrated software management, product engineering, intergroup coordination, peer reviews
Capability Maturity Model (CMM)

• CMM Level 4 - "Managed"
  – Characteristics: quantitative; reasonable statistical control over product quality
  – Key elements: process measurement and analysis, quality management

• CMM Level 5 - "Optimizing"
  – Characteristics: quantitative basis for continual process automation and improvement
  – Key elements: defect prevention, technology innovation, process change management
The CMM Integration (CMMI)

- Integrate practices from four CMMs to generalize (not just for software maturity)
  - Maturity Level 1: Initial
    - Processes are ad-hoc and chaotic
  - Maturity Level 2: Managed
    - Focuses on basic project management
  - Maturity Level 3: Defined
    - Focuses on process standardization
  - Maturity Level 4: Quantitatively Managed
    - Focuses on quantitative management
  - Maturity Level 5: Optimizing
    - Focuses on continuous process improvement
SPR Maturity Assessment

• **Software Productivity Research (SPR) Assessment**
  – Much like [CMM](#), but focuses more broadly on corporate **strategy** and **tactical** issues as well as CMM's issues of software organization and process
  – Also uses a questionnaire, but has **400 questions** as opposed to CMM's 85, and uses a five-point **scale** instead of yes-no answers
    • Excellent, Good, Average, Below Average, Poor
SPR Maturity Assessment

• **SPR Assessment**
  
  – Assessment uses measures such as:
    
    • Quality and productivity measurements
    
    • Experience of programmers in defect removal and testing
    
    • Project quality and reliability targets
    
    • Defect removal history in each phase (design, coding, testing, release)
Baldrige Assessment Standard

• Malcolm Baldrige National Quality Award (MBNQA)
  – Originally U.S. Department of Commerce award, given to recognize outstanding achievement in quality management and achievement in any industry
  – Also basis of IBM's Market Driven Quality strategy and the European Quality Award
  – An "examination" for award criteria, companies get a "mark" out of 1,000
Baldrige Assessment Standard

• **MBNQA**
  
  – 28 examination items, in seven categories: leadership, information and analysis, quality planning, human resources, quality assurance, quality results, customer satisfaction
  
  – Three evaluation dimensions of each item
    
    • **Approach**: methods used to achieve the examination item
    
    • **Deployment**: how well approach is actually applied
    
    • **Results**: quality of outcome in examination item
Malcolm Baldrige, Jr.

• 26th United States Secretary of Commerce
  – In his prior career as a businessman, he led the conversion of a troubled brass mill to a highly diversified manufacturer of industrial goods
  – His experience with process improvement led to the guidelines in the National Quality Improvement Act of 1987
Impact of the MBNQA

- Evaluated in 2001 for economic benefit
  - Social costs of the program were $119 million
  - Net private benefits to the economy were conservatively estimated at $24.65 billion
  - The social benefit-to-cost ratio was 207-to-1
  - Prior to the quality improvement act, many U.S. businesses either did not believe that quality mattered for them or they did not know where to begin
ISO 9000 Standard

• ISO 9000
  – A set of standards and guidelines for quality assurance management
  – Many customers, especially in Europe, require ISO 9000 registration of their suppliers
  – Companies become ISO 9000 "registered" as a result of a formal audit by ISO
  – ISO 9000 standards are documentation-based

  • Every aspect of every step of every process must be backed up by formal documents in a precisely defined format keeping records of how processes are applied
ISO 9000 Standard

• ISO 90003
  – ISO 90003 gives the standards for software development, supply, and maintenance
  – ISO 90003 specifies 20 elements to be assessed, with detailed requirements for each element
## ISO 90003

<table>
<thead>
<tr>
<th>Management responsibility</th>
<th>Inspection, measuring, and test equipment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality system</td>
<td>Inspection and test status</td>
</tr>
<tr>
<td>Contract review</td>
<td>Control of nonconforming products</td>
</tr>
<tr>
<td>Design control</td>
<td>Corrective action</td>
</tr>
<tr>
<td>Document control</td>
<td>Handling, packaging, delivery</td>
</tr>
<tr>
<td>Purchasing</td>
<td>Quality records</td>
</tr>
<tr>
<td>Purchaser-supplied product</td>
<td>Internal quality audits</td>
</tr>
<tr>
<td>Product identification and traceability</td>
<td>Training</td>
</tr>
<tr>
<td>Process control</td>
<td>Servicing</td>
</tr>
<tr>
<td>Inspection and testing</td>
<td>Statistics</td>
</tr>
</tbody>
</table>
ISO 9000 Standard

• ISO 9000
  – Standards are complex, detailed, and stringent
  – "Say what you do, do what you say, and prove it."

• Example:
  – The documentation standard goes so far as to specify:
    • owner of document must be specified on title page
    • distribution of document must be controlled with an archived master copy, distribution record book, etc.
    • version level must be clearly identified
    • all pages must be consecutively numbered
    • total number of pages must be indicated on title page
    • procedure for destruction of obsolete documents must be documented
ISO 9000 Standard

• ISO 9000
  – Most companies (60-70%) fail the ISO audit the first time
  – Most software companies are deficient in corrective actions and document control
  – Companies take steps to meet the standards in these areas and usually can be registered on the second try
  – Over a million organizations worldwide are independently certified
Criticisms of ISO 9000

• Companies may misunderstand the goal
  – ISO 9000 certification is desirable for getting customers
  – A company must want to apply the knowledge gained from obtaining ISO 9000 certification to improve quality processes
    • It is not enough to simply get the certificate and be done with it!
    • "A company can produce a poor quality product consistently, and with the proper documentation can put an ISO 9000 stamp on it."
Summary

• Software Process Evaluation
  – Software processes can be continually improved using meta-processes such as the Defect Prevention Process
  – Software processes can be evaluated with respect to their maturity or by comparison with a process standard
  – Maturity models include CMM and SPR
  – Process quality standards include Baldrige and ISO 9000
Summary

• Today's References
  – Kan, Metrics and Models in Software Quality Engineering, ch. 2.

• Next Time
  – The 2015 CISC 327 software project
  – Then: The eXtreme Programming software process