Software Inspection Process

Technical Overview

David F. Rico
Table of Contents

• Preface
• Introduction
• Benefits
• Process
• Forms
• Roles
• Case Studies

• Metrics
• Deployment
• Management
• Pit Falls
• Research
• Conclusion
• Resources
Preface
Reviews are formal team evaluations of software work products, which identify any discrepancies from specifications and standards, or provide recommendations after the examination of alternatives, or both.
Walkthroughs are unstructured meetings held by software managers to publicize design and implementation concepts, without obligation to use any feedback, alternative ideas, or suggested changes resulting from the meeting.
Inspections are structured and neutrally facilitated meetings for technical peers to identify defects in software work products which must be corrected, without suggesting solutions or interference from the originator of the work product.
Followup Questions

• What is a review ???
• What is a walkthrough ???
• What is an inspection ???
Introduction
• A simple process to identify defects
• Highly *structured* meeting
• Forum for *independent* evaluation
• Form of *static* analysis or *static* testing
• *Early*, *in-process* validation technique
• Form of *quality* and *reliability* engineering
• Performed by *software engineering*
What are the goals?

• Identify as many defects as possible
• Identify defects in *early* stages of life cycle
• Identify defects *before* testing and fielding
• Identify defects *cheaply* and *inexpensively*
• *Reduce* development and maintenance *costs*
• *Shorten* development *cycle time*
• *Quantitatively* control quality and reliability
Where did it come from?

• Created by Michael Fagan of IBM in 1972
• Typically referred to as Fagan Inspections
• Adaptation of statistical quality control to large systems computer programming
• First published in IBM Systems Journal
  – “Design and code inspections to reduce errors in program development”
  – Volume 15, Number 3, 1976
Why do it?

- Reduce development and maintenance costs
- Improve software quality and reliability
- Initiate effective verification and validation
- Reduce cost and risk of software testing
- Reduce dependence on quality assurance
- Support SEI Levels 3, 4, and 5 (e.g., software quality metrics, process metrics, defect prevention, change management)
# What is it like?

## Static Review Technique

<table>
<thead>
<tr>
<th>Feature</th>
<th>Inspection</th>
<th>Desk Check</th>
<th>Walkthrough</th>
<th>Audit</th>
<th>Phase Review</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purpose</strong></td>
<td>Defect Identification</td>
<td>Defect Identification</td>
<td>Design Evaluation</td>
<td>Process Verification</td>
<td>Progress Evaluation</td>
</tr>
<tr>
<td><strong>Timing</strong></td>
<td>Very Early</td>
<td>Early</td>
<td>Early</td>
<td>Late</td>
<td>Very Late</td>
</tr>
<tr>
<td><strong>When</strong></td>
<td>Product Completion</td>
<td>Product Completion</td>
<td>Product Completion</td>
<td>Phase Completion</td>
<td>Phase Completion</td>
</tr>
<tr>
<td><strong>People</strong></td>
<td>4</td>
<td>1</td>
<td>5-15</td>
<td>5</td>
<td>10-200</td>
</tr>
<tr>
<td><strong>Cost</strong></td>
<td>Very Low</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
<td>Very High</td>
</tr>
<tr>
<td><strong>Who</strong></td>
<td>Engineers</td>
<td>Engineers</td>
<td>Project Manager</td>
<td>Quality Assurance</td>
<td>Customer</td>
</tr>
<tr>
<td><strong>Independent</strong></td>
<td>Very High</td>
<td>Low</td>
<td>Very Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td><strong>Rework</strong></td>
<td>Mandatory</td>
<td>Not Mandatory</td>
<td>Not Mandatory</td>
<td>Not Mandatory</td>
<td>Not Mandatory</td>
</tr>
<tr>
<td><strong>Pace</strong></td>
<td>Slow</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td>Very High</td>
</tr>
<tr>
<td><strong>Efficiency</strong></td>
<td>Very High</td>
<td>Average</td>
<td>Low</td>
<td>Low</td>
<td>Very Low</td>
</tr>
<tr>
<td><strong>Measurable</strong></td>
<td>Very High</td>
<td>High</td>
<td>Very Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td><strong>Metric</strong></td>
<td>Defect</td>
<td>Defect</td>
<td>N/A</td>
<td>Deviation</td>
<td>Change Request</td>
</tr>
</tbody>
</table>
What isn’t it?

• Not for software quality assurance group
• Not for design alternative evaluation
• Not for management participation
• Not for individual performance evaluation
• Not for socio-political assassination
• Not at all like a Walkthrough
• Not a notoriously late and ineffective manufacturing inspection
How does it work?

- Has a *singular* objective
- Introduces notion of *counting* defects
- Precision *mechanics* of an audit
- *Mandatory* defect correction
- Uses “*second set of eyes*” principle
- Rational orchestration of *facilitated* forum
- Identifies defects at *early point-of-origin*
- Exploits uniquely-skilled *domain experts*
Followup Questions

- Inspections are simple processes for ???
- What is the goal of inspections ???
- Who created inspections ???
- Why perform inspections ???
- What are inspections similar to ???
- What aren’t inspections ???
- How do inspections work ???
Benefits
What is the return on investment?

- Average return on investment of 133:1
  - DACS estimates an ROI of 72:1
  - AT&T estimates an ROI of 234:1
  - Rico estimates an ROI of 160:1
  - BNR estimates an ROI of 114:1
  - Gilb estimates an ROI of 113:1
  - HP estimates an ROI of 104:1
What is the cycle time reduction?

- Average cycle time reduction of 5.5x
  - DACS estimates a reduction of 1.55x
  - Fagan estimates a reduction of 6.67x
  - AT&T estimates a reduction of 8.37x
  - Rico estimates a reduction of 6.54x
  - BNR estimates a reduction of 5.17x
  - Gilb estimates a reduction of 5.13x
  - HP estimates a reduction of 4.84x
What is the quality increase?

- Average quality increase of 16.4x
  - Rico estimates an increase of 3.03x
  - Bull HN estimates an increase of 76.93x
  - Aetna estimates an increase of 5.56x
  - IBM estimates an increase of 14.29x
  - BNR estimates an increase of 4.99x
  - AT&T estimates an increase of 3.35x
  - Fagan estimates an increase of 6.67x
What is the productivity increase?

• Average productivity increase of 6x
  – DACS estimates an increase of 1.55x
  – Fagan estimates an increase of 6.67x
  – AT&T estimates an increase of 8.37x
  – Rico estimates an increase of 6.54x
  – BNR estimates an increase of 5.17x
  – Gilb estimates an increase of 5.13x
  – HP estimates an increase of 4.84x
What is the defect removal efficiency?

- Average defect removal efficiency of 82.2%
  - Rico estimates an efficiency of 67%
  - Bull HN estimates an efficiency of 98.7%
  - Aetna estimates an efficiency of 82%
  - IBM estimates an efficiency of 93%
  - BNR estimates an efficiency of 80%
  - AT&T estimates an efficiency of 70%
  - Fagan estimates an efficiency of 85%
What is the break even point?

- Average break even point of 23.02 hours
  - AT&T estimates a break even of 21.58 hours
  - Rico estimates a break even of 22.43 hours
  - BNR estimates a break even of 23.56 hours
  - Gilb estimates a break even of 23.61 hours
  - HP estimates a break even of 23.95 hours
What is the accuracy?

- Average accuracy of 92.3%
  - Estimate accuracy of 100% for a 680K project
  - Estimate accuracy of 100% for a 30K project
  - Estimate accuracy of 75% for a 70K project
  - Estimate accuracy of 89% for a 1,700K project
  - Estimate accuracy of 86% for a 290K project
  - Estimate accuracy of 96% for a 70K project
  - Estimate accuracy of 92% for a 540K project
  - Estimate accuracy of 100% for a 700K project
Follow-up Questions

• What is the return-on-investment ???
• What is the cycle time reduction ???
• What is the quality increase ???
• What is the productivity increase ???
• What is the defect removal efficiency ???
• What is the break even point ???
• What is the accuracy ??
Process
What is the overall process?

<table>
<thead>
<tr>
<th>SOFTWARE INSPECTION PROCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purpose</strong></td>
</tr>
<tr>
<td>✓ Team Identification of Software Work Product Defects</td>
</tr>
<tr>
<td><strong>Input</strong></td>
</tr>
<tr>
<td>✓ Predecessor Specifications</td>
</tr>
<tr>
<td>✓ Software Work Product Standards</td>
</tr>
<tr>
<td>✓ Software Work Product</td>
</tr>
<tr>
<td>✓ Software Work Product Overview</td>
</tr>
<tr>
<td>✓ Statement-of-Work</td>
</tr>
<tr>
<td>✓ Software Defect Types</td>
</tr>
<tr>
<td>✓ Checklists</td>
</tr>
<tr>
<td>✓ Inspection Defect List</td>
</tr>
<tr>
<td><strong>Activity</strong></td>
</tr>
<tr>
<td>✓ Planning Activity - Organize Inspections</td>
</tr>
<tr>
<td>✓ Overview Activity - Describe Software Work Products</td>
</tr>
<tr>
<td>✓ Preparation Activity - Analyze Software Work Products</td>
</tr>
<tr>
<td>✓ Meeting Activity - Identify Software Defects</td>
</tr>
<tr>
<td>✓ Rework Activity - Correct Software Defects</td>
</tr>
<tr>
<td>✓ Followup Activity - Verify Software Defect Corrections</td>
</tr>
<tr>
<td><strong>Output</strong></td>
</tr>
<tr>
<td>✓ Inspection Defect Summary</td>
</tr>
<tr>
<td>✓ Inspection Report</td>
</tr>
<tr>
<td>✓ Software Work Product</td>
</tr>
</tbody>
</table>
## What is the planning activity?

### Purpose
- Organize Software Work Product Inspections

### Input
- Predecessor Specifications
- Software Work Product Standards
- Software Work Product

### Activity
- Authors Submit Software Work Products
- Moderators/Authors Review Software Work Products
- Moderators Select Inspectors/Assign Roles
- Moderators Schedule Overviews/Meetings
- Moderators Arrange Overview/Meeting Locations
- Moderators Prepare Inspection Meeting Notices
- Moderators Distribute Inspection Materials

### Output
- Inspection Meeting Notice
- Statement-of-Work
- Predecessor Specifications
- Software Work Product Standards
- Software Defect Types
- Checklists
- Software Work Product
What is the overview activity?

<table>
<thead>
<tr>
<th>OVERVIEW ACTIVITY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purpose</strong></td>
</tr>
<tr>
<td><strong>Input</strong></td>
</tr>
</tbody>
</table>
| **Activity**      | ✓ Authors Prepare Software Work Product Overviews  
|                   | ✓ Moderators Facilitate Overviews  
|                   | ✓ Authors Distribute Software Work Product Overviews  
|                   | ✓ Authors Present Software Work Product Overviews  
|                   | ✓ Inspectors Ask Software Work Product Questions  
|                   | ✓ Authors Answer Software Work Product Questions  
|                   | ✓ Inspectors Note Software Work Product Problems |
| **Output**        | ✓ Software Work Product Overview |
What is the preparation activity?

<table>
<thead>
<tr>
<th>PREPARATION ACTIVITY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purpose</strong></td>
</tr>
</tbody>
</table>
| **Input**             | ✓ Statement-of-Work  
|                       | ✓ Predecessor Specifications  
|                       | ✓ Software Work Product Standards  
|                       | ✓ Software Defect Types  
|                       | ✓ Checklists  
|                       | ✓ Software Work Product |
| **Activity**          | ✓ Inspectors/Recorders Review Software Defect Types  
|                       | ✓ Inspectors Review Checklists  
|                       | ✓ Inspectors Review Statements-of-Work  
|                       | ✓ Inspectors Review Software Work Product Standards  
|                       | ✓ Inspectors Review Predecessor Specifications  
|                       | ✓ Inspectors Review Software Work Products  
|                       | ✓ Readers Select Narration Techniques |
| **Output**            | ✓ Software Work Product |
**What is the meeting activity?**

<table>
<thead>
<tr>
<th>MEETING ACTIVITY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purpose</strong></td>
</tr>
<tr>
<td>✓ Team Identification of Software Work Product Defects</td>
</tr>
<tr>
<td><strong>Input</strong></td>
</tr>
<tr>
<td>✓ Statement-of-Work</td>
</tr>
<tr>
<td>✓ Predecessor Specifications</td>
</tr>
<tr>
<td>✓ Software Work Product Standards</td>
</tr>
<tr>
<td>✓ Software Defect Types</td>
</tr>
<tr>
<td>✓ Checklists</td>
</tr>
<tr>
<td>✓ Software Work Product</td>
</tr>
<tr>
<td><strong>Activity</strong></td>
</tr>
<tr>
<td>✓ Moderators Facilitate Meetings</td>
</tr>
<tr>
<td>✓ Readers Narrate Software Work Products</td>
</tr>
<tr>
<td>✓ Inspectors Ask Software Work Product Questions</td>
</tr>
<tr>
<td>✓ Authors Answer Software Work Product Questions</td>
</tr>
<tr>
<td>✓ Inspectors Identify Software Work Product Defects</td>
</tr>
<tr>
<td>✓ Recorders Transcribe Software Work Product Defects</td>
</tr>
<tr>
<td>✓ Moderators Review Inspection Defect Lists</td>
</tr>
<tr>
<td>✓ Moderators Disposition Software Work Products</td>
</tr>
<tr>
<td><strong>Output</strong></td>
</tr>
<tr>
<td>✓ Inspection Defect List</td>
</tr>
</tbody>
</table>
What is the rework activity?

## REWORK ACTIVITY

<table>
<thead>
<tr>
<th>Purpose</th>
<th>✓ Mandatory Software Work Product Defect Correction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>✓ Inspection Defect List ✓ Software Work Product</td>
</tr>
<tr>
<td>Activity</td>
<td>✓ Authors Obtain Inspection Defect Lists ✓ Authors Obtain Software Work Products ✓ Authors Review Inspection Defect Lists ✓ Authors Correct Software Work Product Defects ✓ Authors Correct New Software Work Product Defects ✓ Authors Verify Defect Type, Class, and Severity ✓ Authors Submit Reworked Software Work Products</td>
</tr>
<tr>
<td>Output</td>
<td>✓ Software Work Product</td>
</tr>
</tbody>
</table>
What is the followup activity?

| FOLLOWUP ACTIVITY |
|-------------------|---------------------------------------------------|
| **Purpose**       | ✓ Verify/Summarize Software Work Product Corrections |
| **Input**         | ✓ Inspection Defect List ✓ Software Work Product |
| **Activity**      | ✓ Moderators Obtain Reworked Software Work Products ✓ Moderators/Authors Review Software Work Products ✓ Moderators Prepare Inspection Defect Summaries ✓ Moderators Prepare Inspection Reports ✓ Moderators/Authors Verify Summaries/Reports ✓ Moderators Distribute Inspection Reports ✓ Moderators Submit Software Work Products to SCM |
| **Output**        | ✓ Inspection Defect Summary ✓ Inspection Report ✓ Software Work Product |
Followup Questions

• What is the purpose of inspections ???
• What is the purpose of planning ???
• What is the purpose of the overview ???
• What is the purpose of preparation ???
• What is the purpose of the meeting ???
• What is the purpose of rework ???
• What is the purpose of the followup ???
Forms
## Inspection Meeting Notice

<table>
<thead>
<tr>
<th>Date:</th>
<th>Component:</th>
<th>Moderator:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Project:</th>
<th>Release:</th>
<th>Phone:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Activity:</th>
<th>Document:</th>
<th>Location:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Meeting Type:
- [ ] Overview
- [ ] Inspection
- [ ] Re-Inspection

### Inspection Type:
- [ ] Software Installation Plan
- [ ] Software Integration Plan
- [ ] Test or Validation Plan
- [ ] System Requirements Specification
- [ ] Database Design Description
- [ ] Software Architecture Description
- [ ] System Architecture Description
- [ ] Software Design Description
- [ ] Software Interface Design Description
- [ ] Software Requirements Description
- [ ] User Documentation Description
- [ ] Test or Validation Procedure
- [ ] Software Integration Audit Report
- [ ] Test or Validation Results Report

<table>
<thead>
<tr>
<th>Meeting Date:</th>
<th>Meeting Duration:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Meeting Time:</th>
<th>Software Work Product Size:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Meeting Location:</th>
<th>Expected Preparation Time:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Meeting Participants:

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Inspection defect list

## Inspection Defect List

<table>
<thead>
<tr>
<th>Date:</th>
<th>Component:</th>
<th>Moderator:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Project:</th>
<th>Release:</th>
<th>Phone:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Activity:</th>
<th>Document:</th>
<th>Location:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Meeting Type:
- [ ] Overview
- [ ] Inspection
- [ ] Re-Inspection

### Inspection Type:
- [ ] Software Installation Plan
- [ ] Software Integration Plan
- [ ] Test or Validation Plan
- [ ] System Requirements Specification
- [ ] Database Design Description
- [ ] Software Architecture Description
- [ ] System Architecture Description
- [ ] Software Design Description
- [ ] Software Interface Design Description
- [ ] Software Requirements Description
- [ ] User Documentation Description
- [ ] Test or Validation Procedure
- [ ] Software Integration Audit Report
- [ ] Test or Validation Results Report

### Disposition:
- [ ] Accept
- [ ] Conditional
- [ ] Re-Inspect

### Location:

<table>
<thead>
<tr>
<th>Defect Description:</th>
<th>Type:</th>
<th>Class:</th>
<th>Severity:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Type:
- Data, Documentation, Functionality, Human Factors, Interface, Input/Output, Logic, Maintainability, Performance, Syntax, Standards, Test, Other

### Class:
- Missing, Wrong, Extra

### Severity:
- Major, Minor
Inspection defect summary

Inspection Defect Summary

- **Date:**
- **Component:**
- **Moderator:**
- **Project:**
- **Release:**
- **Phone:**
- **Activity:**
- **Document:**
- **Location:**

**Meeting Type:**
- Overview
- Inspection
- Re-Inspection

**Inspection Type:**
- Software Installation Plan
- Software Integration Plan
- Test or Validation Plan
- System Requirements Specification
- Database Design Description
- Software Architecture Description
- System Architecture Description
- Software Design Description
- Software Interface Design Description
- Software Requirements Description
- User Documentation Description
- Test or Validation Procedure
- Software Integration Audit Report
- Test or Validation Results Report

**Disposition:**
- Accept
- Conditional
- Re-Inspect

**Defect:**

<table>
<thead>
<tr>
<th>Defect</th>
<th>MINOR DEFECTS</th>
<th>MAJOR DEFECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Missing</td>
<td>Wrong</td>
</tr>
<tr>
<td>Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Documentation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Functionality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Human Factors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interface</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input/Output</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Logic</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Inspection Report

<table>
<thead>
<tr>
<th>Date:</th>
<th>Component:</th>
<th>Moderator:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Project:</th>
<th>Release:</th>
<th>Phone:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Activity:</th>
<th>Document:</th>
<th>Location:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Meeting Type:
- [ ] Overview
- [ ] Inspection
- [ ] Re-Inspection

### Inspection Type:
- [ ] Software Installation Plan
- [ ] Software Architecture Description
- [ ] User Documentation Description
- [ ] Software Integration Plan
- [ ] System Architecture Description
- [ ] Test or Validation Procedure
- [ ] Test or Validation Results Report
- [ ] Test or Validation Plan
- [ ] Software Design Description
- [ ] Software Integration Audit Report
- [ ] System Requirements Specification
- [ ] Software Interface Design Description
- [ ] System Architecture Description
- [ ] Software Interface Design Description
- [ ] Database Design Description
- [ ] Software Requirements Description

<table>
<thead>
<tr>
<th>Est. Rework Effort:</th>
<th>Re-Inspection Date:</th>
<th>Duration:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actual Rework Effort:</th>
<th>Inspector Number:</th>
<th>Size of Materials:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Size of Materials:</th>
<th>Certification:</th>
<th>Completion Date:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rework Author:</th>
<th>Meeting Number:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Inspectors:

<table>
<thead>
<tr>
<th>Inspectors:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

### Comments:

<table>
<thead>
<tr>
<th>Comments:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>


Followup Questions

- What is the inspection meeting notice ???
- What is the inspection defect list ???
- What is the inspection defect summary ???
- What is the inspection report ???
- What is the meeting type ???
- What is the inspection type ???
- What are the defect severity types ???
What are the roles?

- Moderator
  - Facilitator
- Author
  - Producer of work product
- Inspector
  - Identifier of defects
- Reader and recorder (separate roles)
  - Paraphraser of product and logger of defects
What is a moderator?

• Function → Classical facilitator (*maestro*)
• Who → Specially *trained* technical lead
• Responsibilities → Careful *coordination*
  – *Maintains* time limits for all activities
  – *Verifies* entry criteria and *schedules* meetings
  – *Manages* overview and inspection subprocesses
  – *Keeps* project managers out of inspection meeting
  – *Allows* only inspectors to identify defects
  – *Mutes* author from interfering with inspection
  – *Prevents* inspectors from identifying “solutions”
  – *Prevents* inspectors from insulting author
  – *Verifies* rework and *records* inspection results
What is an author?

- Function → *Developer* of work product
- Who → *Trained* project manager/engineer
  - *Project or test manager* (project or test plan)
  - *Analyst* (requirements specification)
  - *Designer* (design specification)
  - *Programmer* (software source code)
  - *Tester* (test report)
- Responsibilities → *Passive participation*
  - *Introduces* work product to inspectors
  - *Answers* any questions
  - *Corrects* defects
What is an inspector?

- **Function → Identifier** of defects
- **Who → Trained** project manager/engineer
  - *Project plan*
    - Program and fellow project managers
    - Technical leads (responsible for executing plan)
  - *Requirements, design, code, and test (domain specialist)*
    - Engineers, analysts, designers, coders, and testers
- **Responsibilities → Passive participation**
  - Attends product overview
  - Analyzes defect history and prepares for inspection
  - Identifies defects during inspection subprocess
What is a reader?

- **Function** → *Consumer* of work product

- **Who** → *Trained* project manager/engineer
  - *Project plan* (technical lead is reader)
  - *Test plan* (tester is reader)
  - *Requirements* (designer is reader)
  - *Design* (programmer is reader)
  - *Code* (tester is reader)

- **Responsibilities** → *Passive* participation
  - Attends product overview
  - Studies and practices reading work product
  - Paraphrases work product during inspection subprocess
What is a recorder?

• Function → \textit{Writes down} defects
• Who → \textit{Trained} project manager/engineer
• Responsibilities → \textit{Passive} participation
  – \underline{Becomes} familiar with process and forms
  – \underline{Writes down} defects identified by inspectors
  – \underline{Negotiates} adequate time to write down defects
  – \underline{Subject} to moderator’s direction
  – \underline{Does not write} down information other than defects
  – \underline{Provides} completed defect lists to moderator
Followup Questions

• What is a moderator ???
• What is an author ???
• What is an inspector ???
• What is a reader ???
• What is a recorder ???
Case Studies
Bull HN Information Systems

- System
  - Operating system
  - 11 million lines of code
  - 600,000 lines of code added annually
  - “C” programming language

- Experience
  - 7,413 inspections conducted
  - 11,557 “major” defects identified
  - 98.7% inspection efficiency achieved
  - 667,170 inspection data points (in 3 years)
Bell Northern Research

- **System**
  - Embedded, real-time digital switching systems
  - 15 million lines of code
  - 312,500 lines of code added quarterly
  - Modern, high-level programming languages

- **Experience**
  - 2,778 inspections conducted
  - 240,000 defects identified
  - 80% inspection efficiency achieved
  - 250,020 inspection data points (in 2 years)
• System
  – Operating system
  – 7.1 million lines of code
  – 2 million lines of code added annually
  – PL/1, Jovial, and RPG programming languages

• Experience
  – 7,889 inspections conducted
  – 681,600 defects identified
  – 70% inspection efficiency achieved
  – 710,010 inspection data points (in 3.5 years)
• System
  – Embedded, real-time systems
  – 111,600 lines of code
  – 9,300 lines of code per project average
  – “C” programming language

• Experience
  – 324 inspections conducted
  – 4,860 defects identified
  – 70% inspection efficiency achieved
  – 29,160 inspection data points (in 7 years)
• System
  – Computer aided drafting (CAD) tools
  – 25,920 lines of code
  – 12,960 lines of code per year
  – “C” programming language

• Experience
  – 211 inspections conducted
  – 3,857 defects identified
  – 70% inspection efficiency achieved
  – 18,990 inspection data points (in 2 years)
Lockheed Martin

• System
  – Embedded, real-time system
  – 2 million lines of code
  – 200,000 lines of code added per year
  – “C” programming language

• Experience
  – 23 inspections conducted
  – 324 defects identified
  – 67% inspection efficiency achieved
  – 2,070 inspection data points (in 1 year)
IBM Space Shuttle

• System
  – Man-rated spacecraft avionics system
  – 500,000 lines of code
  – 25,467 lines of code added/maintained per year
  – HAL-S programming language (custom)

• Experience
  – 1,061 inspections conducted
  – 36,672 defects identified
  – 90% inspection efficiency achieved
  – 95,490 inspection data points (in 15 years)
Followup Questions

- Who conducted over 7,000 inspections???
- What was the highest efficiency???
- How many data points can be generated???
- How many cases were maintenance???
- What was the lowest efficiency???
- What languages were inspected???
- Do inspections apply to 4GLs???
Metrics
What are the overall metrics?

- Estimated versus actual effort and duration
  - Estimated versus actual defects
    - Major/minor defects per hour and inspection
    - Defect types per inspection
  - Inspection suppression and gain rate
  - Participants per hour and inspection
  - Subprocess intervals
    - Planning-Overview-Preparation-Rework-Followup
    - Overview-Preparation-Rework-Followup
    - Preparation-Rework-Followup
    - Rework-Followup
What are the planning metrics?

- Effort and duration of planning subprocess
  - Effort to verify entry criteria
    - Number of products passing and failing entry criteria
    - Number of passed and failed entry criteria
    - Entry criteria validation rate
  - Effort to select participants
  - Effort to schedule inspection
  - Effort to prepare notice
What are the overview metrics?

- Effort and duration of overview subprocess
  - Effort to facilitate overview
  - Effort to introduce product
    - Product presentation rate
    - Number of product inquiries
    - Number of inquiries handled
    - Number of inquiries deferred
  - Effort to assign roles
  - Number of participants
What are the preparation metrics?

- Effort and duration of preparation subprocess
  - Effort to analyze specifications
  - Effort to analyze checklists
  - Effort to analyze defect history
    - Number and type of estimated/expected defects
  - Effort to analyze product
    - Product analysis rate
  - Effort to note potential defects
    - Number of potential defects noted
  - Number of participants
What are the meeting metrics?

• Effort and duration of meeting subprocess
  – Effort to facilitate meeting
  – Effort to inspect product
    • Product inspection rate
  – Effort to review and summarize defects
  – Number of major and minor defects
  – Number of participants
What are the rework metrics?

- Effort and duration of rework subprocess
  - Effort to review defect list
  - Effort to correct defects
  - Number of major and minor defects corrected
  - Effort to verify defect correction
What are the followup metrics?

- Effort and duration of followup subprocess
  - Effort to verify rework
  - Effort to summarize inspection
How are metrics collected?

- Software metrics plan
- Software quality plan
- Software project plan
- Inspection forms
- Spreadsheets
- Desktop databases
- Multi-user databases
Followup Questions

• What were the overall metrics ???
• What were the planning metrics ???
• What were the overview metrics ???
• What were the preparation metrics ???
• What were the meeting metrics ???
• What were the rework metrics ???
• What were the followup metrics ???
Deployment
What is a defect?

- Nonconformance to requirements
- Deviation from specification
- Untestable requirement
- Abnormal condition
- Unmet standards
- Erroneous state
- Failure
What should be inspected?

• Strategic enterprise artifacts
  – Statement of work (SOW)
  – Project plans
  – Requirements
  – Designs
  – Code
  – Tests
  – Quality plans
What is the required training?

- Executive overview
  - Organizational, economic, and political impacts
- Introductory overview
  - Function, uniqueness, and power of inspections
- Management overview
  - Do’s, don’ts, and project planning
- Other (technical, metrics, and auditing)
  - Mechanics, measurement, and enforcement
Who should be trained?

- **Executives**
  - *Costs and benefits*

- **Managers**
  - *Planning, estimating, and tracking inspections*

- **Engineers**
  - *Mechanics, rules, and optimization*

- **Software process and quality analysts**
  - *Deployment support, auditing, and analysis*
Why is training required?

- Train managers to
  - Plan and manage projects using *metrics*
  - Use metric data responsibly (*not abuse staff*)
- Certify moderators to
  - *Facilitate* and maintain *order*
  - Yield *successful* inspections
- Prepare inspectors to
  - *Identify* defects quickly and efficiently
Why are moderators certified?

- Keep managers out
- Planning and coordination
- Halt unproductive inspections
- Maintain non-threatening forum
- Yield optimal defect identification
- Ensure precision process execution
- Maintain independence and objectivity
Who enforces the process?

- **Inspectors**
  - Responsible for *obeying* rules
- **Moderators**
  - Most *effective* defense (hence, “*certified*”)
- **Project managers**
  - Responsible for intra-inspection *monitoring*
- **Software process improvement and quality**
  - Responsible for *optimization* and *conformance*
Followup Questions

- What is a defect ???
- What should be inspected ???
- Who should be trained ???
- Why is training required ???
- What are moderators trained to do ???
- Who enforces the process ???
- Who performs inspections ???
Management
How is effort estimated?

Hours = \frac{Product\ Size}{(Inspection\ Rate \times 2) \times (Team\ Size \times 4 + 1)}
### When is it done?

<table>
<thead>
<tr>
<th>I. Software Development</th>
<th>Month 1</th>
<th>Month 2</th>
<th>Month 3</th>
<th>Month 4</th>
<th>Month 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Analysis Phase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Requirements</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Inspections</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B. Design Phase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Inspections</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B. Code Phase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Code</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Inspections</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### How often is it done?

<table>
<thead>
<tr>
<th>People</th>
<th>Inspection Duration</th>
<th>Number of Inspections</th>
<th>Hours</th>
<th>Days</th>
<th>Weeks</th>
<th>Months</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>6</td>
<td>83</td>
<td>500</td>
<td>63</td>
<td>13</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>42</td>
<td>250</td>
<td>31</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>28</td>
<td>167</td>
<td>21</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>28</td>
<td>6</td>
<td>6</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SLOC/Hour</th>
<th>10,000</th>
<th></th>
<th>100,000</th>
<th></th>
<th>1,000,000</th>
<th></th>
<th>Total Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>1,417</td>
<td>708</td>
<td>472</td>
<td>14,167</td>
<td>7,083</td>
<td>4,722</td>
<td>141,667</td>
</tr>
<tr>
<td>120</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>180</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>17,500</td>
<td>875</td>
<td>583</td>
<td>18,750</td>
<td>9,375</td>
<td>5,833</td>
<td>175,000</td>
</tr>
<tr>
<td>120</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>180</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20,833</td>
<td>1,042</td>
<td>694</td>
<td>20,833</td>
<td>10,417</td>
<td>6,944</td>
<td>208,333</td>
</tr>
<tr>
<td>12</td>
<td>24,167</td>
<td>1,208</td>
<td>806</td>
<td>24,167</td>
<td>12,083</td>
<td>8,056</td>
<td>241,667</td>
</tr>
<tr>
<td>7</td>
<td>2,417</td>
<td>1,208</td>
<td>806</td>
<td>24,167</td>
<td>12,083</td>
<td>8,056</td>
<td>241,667</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SLOC/Hour</th>
<th>10,000</th>
<th>120</th>
<th>180</th>
<th>60</th>
<th>120</th>
<th>180</th>
<th>60</th>
<th>120</th>
<th>180</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1,417</td>
<td>708</td>
<td>472</td>
<td>14,167</td>
<td>7,083</td>
<td>4,722</td>
<td>141,667</td>
<td>70,833</td>
<td>47,222</td>
</tr>
<tr>
<td>5</td>
<td>1,750</td>
<td>875</td>
<td>583</td>
<td>17,500</td>
<td>8,750</td>
<td>5,833</td>
<td>175,000</td>
<td>87,500</td>
<td>58,333</td>
</tr>
<tr>
<td>6</td>
<td>2,083</td>
<td>1,042</td>
<td>694</td>
<td>20,833</td>
<td>10,417</td>
<td>6,944</td>
<td>208,333</td>
<td>104,167</td>
<td>69,444</td>
</tr>
<tr>
<td>7</td>
<td>2,417</td>
<td>1,208</td>
<td>806</td>
<td>24,167</td>
<td>12,083</td>
<td>8,056</td>
<td>241,667</td>
<td>120,833</td>
<td>80,556</td>
</tr>
</tbody>
</table>

Estimated Source Lines of Code (SLOC) per Software Project
## How are defects measured?

<table>
<thead>
<tr>
<th>Source</th>
<th>Metric Name</th>
<th>Metric Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE</td>
<td>Defect Density</td>
<td>$\frac{\text{Defects}}{\text{KSLOC}}$</td>
</tr>
<tr>
<td>IBM (Michael Fagan)</td>
<td>Defect Removal Effectiveness</td>
<td>$\frac{\text{Inspection Defects}}{\text{Inserted Defects}} \times 100%$</td>
</tr>
<tr>
<td>IBM (NASA Space Shuttle)</td>
<td>Early Detection Percentage</td>
<td>$\frac{\text{Major Inspection Defects}}{\text{Inserted Defects}} \times 100%$</td>
</tr>
<tr>
<td>Dunn</td>
<td>Effectiveness</td>
<td>$\frac{\text{Defects}}{\text{Current Phase + Post Phase}} \times 100%$</td>
</tr>
<tr>
<td>Motorola</td>
<td>Total Defect Containment Effectiveness</td>
<td>$\frac{\text{Pre-Release Defects}}{\text{Pre-Release + Post-Release Defects}}$</td>
</tr>
<tr>
<td>Motorola</td>
<td>Phase Containment Effectiveness</td>
<td>$\frac{\text{Phase Errors}}{\text{Phase Errors + Phase Defects}}$</td>
</tr>
</tbody>
</table>
How are defects estimated?

• Basic (simple, but powerful)
  – Observed defect density (immediate)
  – Complete estimation of detection ratio

• Intermediate (popular)
  – Partial estimation of detection ratio
    (a.k.a. Capture-recapture models)

• Advanced (accurate and methodical)
  – Rayleigh life cycle reliability models
How are metrics analyzed?

\[ \sum_{i=1}^{n} x \pm \sigma \cdot \sqrt{\frac{\sum_{i=1}^{n} x^2}{n} - \left( \frac{\sum_{i=1}^{n} x}{n} \right)^2} \]
How are metrics applied?

Defect Reduction

Software Process Improvement

Earlier Defect Removal

Residual Defects Delivered to Customers

Defects

Analysis  Preliminary Design  Detailed Design  Code  Unit Test  Component Test  System Test  Field Operation

Software Inspections
(10-100x Cheaper than Software Testing)

Software Testing
(10-100x More than Inspections)

Software Testing
(10-100x More than Inspections)
Followup Questions

• What is inspection effort a factor of ???
• When are inspections performed ???
• Where are inspections scheduled ???
• How many inspections per work product ???
• Can defects be estimated ???
• Why should defects be analyzed ???
• Why should defects be eliminated early ???
Pitfalls
What are the confusing points?

• What is a defect?
• Why do inspections?
• When do inspections occur?
• Why not just have a meeting?
• Who’s responsible for inspections?
• Isn’t it better to use an independent group?
• Shouldn’t quality assurance do inspections?
Why aren’t inspections used?

- Programming viewed as a trade
- Large body of amateur practitioners
- Not in computer science curriculum
- Good descriptive literature is lacking
- Inspections spread by word-of-mouth
- Many are highly critical of inspections
- Not in software engineering curriculum
- Benefits of inspections are known by a few
What are the common obstacles?

• Engineers are only interested in design
• Using latest technologies is high priority
• Managers don’t understand their benefits
• Managers don’t perceive them as valuable
• Software quality/reliability is not a priority
• Managers won’t institutionalize inspections
• Engineers will not participate in inspections
• Winning contracts is the only success factor
What are the common myths?

- Too expensive
- No government mandate
- Obsolete mainframe era technique
- Identical to structured walkthroughs
- No more effective than desk checking
- Equivalent of manufacturing inspections
- Not a verification and validation technique
- Not applicable to Internet age technologies
What are the common mistakes?

- Attack the author
- Don’t take them seriously
- Inspect at a high rate of speed
- Inspect for longer than 2 hours
- Don’t prepare for the inspections
- Evaluate design and style alternatives
- Let the participants control the inspection
What are the pitfalls of metrics?

- Used for personal attacks
- Collection is cumbersome
- Dissimilar data often compared
- Transcription errors are common
- Invalid data is also very common
- Data from chaotic process isn’t as good
- Dissimilar circumstances often compared
What are the pitfalls of politics?

- Programming is very competitive
- Inspections depend on heavy teamwork
- Cooperation and teamwork are uncommon
- Managers don’t want engineers to succeed
- Engineers don’t want managers to succeed
- Managers can use metrics for personal gain
- Engineers report incorrect data to managers
Followup Questions

- What are common points of confusion ???
- Is computer programming a trade ???
- Why don’t engineers use inspections ???
- Are inspections too expensive ???
- What is the most common mistake ???
- How are metrics commonly abused ???
- Does division of labor hinder inspections ???
David F. Rico

- Research
  - Costs, benefits, process modeling, and training

- Findings
  - Cost estimation (only published models)
  - Advanced cost and benefit evaluation method
  - Return-on-investment (ROI) model
  - Total life cycle cost analysis

- Contact
  - http://davidfrico.com
University of Maryland

• Research
  – Cost and benefits of inspection “variations”

• Findings
  – New inspection metrics and models
  – Groups no more effective than individuals
  – Reading technologies are promising focus areas

• Contact
  – http://www.cs.umd.edu
• Research
  – Orthogonal defect classification (ODC)

• Findings
  – Inspection defect type classification is subjective
  – Designed objective defect classification method
  – Defect type signatures identify process failures

• Contact
  – http://chillarege.com
• **Research**
  – Costs and benefits of inspections

• **Findings**
  – Basic cost and benefit evaluation method
  – Return-on-investment (ROI) model
  – Total life cycle cost analysis

• **Contact**
  – http://www.dacs.dtic.mil
Fraunhofer Gesellschaft

• Research
  – Costs, benefits, and quality modeling

• Findings
  – Cost and benefit evaluations
  – Capture-recapture models (latest research)
  – Inspection metrics and models

• Contact
  – http://www.iese.fhg.de
University of Hawaii

• Research
  – Inspection bibliographic studies

• Findings
  – Extensive online annotated bibliography

• Contact
  – http://www.ics.hawaii.edu
• Research
  – Costs and benefits of inspection “variations”

• Findings
  – Good alternatives to Fagan inspections
  – Flagship capture-recapture modeling research
  – Testbed for University of Maryland

• Contact
  – http://www.research.att.com
Followup Questions

- Who has the only published cost models ???
- What does University of Maryland claim ???
- What is orthogonal defect classification ???
- Does ITT do inspection research ???
- Who’s doing the latest research ???
- Who has an extensive bibliography ???
- What was AT&T known for ??
Perceived power of inspections

- Group synergy
- Ghost inspector
- Structured teamwork
- Focused human intelligence
- High defect removal efficiency
- Many eyes are better than two eyes
- Group review better than individual review
Real power of inspections

- Counting defects
- Identifying defects early
- Mandatory defect correction
- Defect classification & analysis
- Defect prevention (causal analysis)
- Defect modeling (however imperfect)
- Life cycle reliability modeling (Rayleigh)
Inspections and management

• Inspections enable managers to
  – Estimate defects
  – Plan defect removal
  – Track defect removal
  – Track total life cycle costs
  – Quantitatively track progress
  – Track return-on-investment (ROI)
  – Learn how to manage projects using metrics
Inspections and engineering

• Inspections enable engineers to
  – Learn how to apply metrics
  – Assume responsibility for quality
  – Assume responsibility for reliability
  – Perform proactive quality engineering
  – Perform proactive reliability engineering
  – Build trust in the software life cycle process
  – Gain respect for software engineering discipline
Inspections and quality assurance

• Inspections enable quality assurance to
  – Focus on process automation
  – Focus on process simplification
  – Focus on quality and reliability modeling
  – Focus on defect analysis and classification
  – Assume their rightful process analysis role
  – Focus on root cause analysis and prevention
  – Return responsibility for quality to engineering
Inspections and testing

• Inspections are
  – 10x cheaper than testing
  – 100x cheaper than maintenance
• Inspections enable managers to
  – Reduce the cost of testing
  – Reduce total life cycle costs
  – Reduce software maintenance costs
  – Quantify and manage the costs of testing
Inspections and the CMM

- Inspections embody key CMM principles:
  - Software Project Planning
  - Software Project Tracking & Oversight
  - Peer Reviews
  - Software Quality Management
  - Quantitative Process Management
  - Defect Prevention
  - Technology Change Management
  - Process Change Management
Followup Questions

• What is the perceived power ???
• What is the real power ???
• How do managers use inspections ???
• How do engineers use inspections ???
• How does SQA use inspections ???
• What do inspections mean to testing ???
• What do inspections mean to CMM ??
Resources
Books

• “Software Inspection Process”
  – Robert G. Ebenau
  – Susan H. Strauss

• “Software Inspection”
  – Tom Gilb
  – Dorothy Graham
  – Addison Wesley (1993)
Industry case studies

• “Lessons from Three Years of Inspection Data”
  – Edward F. Weller
  – IEEE Software (September 1993)

• “Experience with Inspection in Ultralarge-Scale Developments”
  – Glen W. Russell
  – IEEE Software (January 1991)
Tools

• SDT ReviewPro
  – http://www.sdtcorp.com/reviewpr.htm

• SyberNet CheckMate
  – http://www.sybernet.ie/source/checkmate.htm

• StrathClyde ASSIST
  – http://www.cs.strath.ac.uk/research/efocs/assist.html
Training

- Michael Fagan Associates
  - http://www.mfagan.com

- Grove Consultants
  - http://www.grove.co.uk/Inspection_Courses.html

- Don O'Neill
  - http://hometown.aol.com/ONeillDon

- Tom Gilb
  - http://www.result-planning.com
Websites

- David F. Rico
  - http://davidfrico.com

- University of Hawaii
  - http://www2.ics.hawaii.edu/%7Ejohnson/FTR/

- DoD Data & Analysis Center for Software
Scientific research

• University of Maryland
  – http://www.cs.umd.edu

• Fraunhofer Gesellschaft
  – http://www.iese.fhg.de
Standards

- IEEE Standard for Software Reviews and Audits

- NASA Software Formal Inspections Standard

- NASA Software Formal Inspections Guidebook
  - http://satc.gsfc.nasa.gov/fi/gdb/fitext.txt