'Regardless of what we discover, we understand and truly believe that everyone did the best job they could, given what they knew at the time, their skills and abilities, the resources available, and the situation at hand.'

Norm Kerth
Project Retrospectives: A Handbook for Team Reviews
http://www.retrospectives.com/
Session Objectives

- **Objectives:**
  1. Understand the Business Rationale for Implementing Peer Reviews.
  2. Understand the types of Reviews
     - Desk Check,
     - Walk-through,
     - Inspection,
     - Quality Estimation.

- **Prerequisites:**
  - Basic understanding of a Software Development or involvement in Software Development Projects

Software Development
The Demands...
Understand the Business Rationale for Implementing Peer Reviews.

- **What Does Management Want?**
  - Predictability
    - Predictable Content and Quality
    - Predictable Cost
    - Predictable Schedule

- **Benefits of Inspections to Management**
  - Provide real-time hard data to help in decision-making
    - Completion of products
      - 90% completion syndrome
    - Measure of quality of products
    - Indicators of difficulties and potential improvements to software processes and products

![Chart showing Chaos Reports 1994 - 2002]

**Chaos Reports 1994 - 2002**

- Type 1: CQFC – OK (Cost, Quality, Functions, Calendar)
- Type 2: Projects completed, but failed CQFC
- Type 3: Projects terminated!
Software Defect Injection and Detection

- Northrop Grumman Space Technology
- Investigated the effectiveness of software defection using peer reviews
- Across 12 system development phases on 14 large-scale systems.
- Mission-critical embedded software systems
  - Robotic spacecraft platforms,
  - Satellite communications,
  - Laser systems
- System sizes ranged from 25,000 to 500,000 source lines,
- Teams included from 10 to 120 developers.
- Study analyzed 3,418 defects from 731 peer reviews

Software Defect Detected when Injected in Same Phase

Defects Detected / Defects Injected (%)

System Development Phase


Defect Injection and Detection

Assumption: Defect removal effectiveness for each test, inspection is 50%

Relative Cost of Defects by Phase Located

($) defect, hours/defect)

IBM: 1978
AT&T: 1985
ICL: 1986
AT&T: 1989
JPL: 1990
IBM: 1991
Shell: 1992
Thorn EMI: 1993
Applicon: 1994
Infosys: 2000


Cost of Defect Removal

Removal Cost * $25 $105 $385 $620 $1150 $6500


Notes: 1. A defect found at requirement phase costs $25 to fix.
   If the same defect is found at Unit Testing, the cost will be $620
2. UT= Unit Test
   SIT/SAT= System Integration & Test/System Acceptance Test

* Note: These costs are data from the value of $ in 1990
Software Development Spending Profiles

Development Schedule (months)

Source: Wheeler, D., Brykcynski, B., Meeson, R. Software Inspection – An Industry Best Practice, Institute of Electrical and Electronics Engineers (IEEE), 1996, p 12

Relative cost of software fault propagation

Phase Defect Introduced

Benefit Ratio of Implementing Inspection

<table>
<thead>
<tr>
<th></th>
<th>Before Review/Inspection</th>
<th>After Implemented Review/Inspection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rework Effort</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req.</td>
<td>8%</td>
<td>1%</td>
</tr>
<tr>
<td>Design</td>
<td>12%</td>
<td>3%</td>
</tr>
<tr>
<td>Code</td>
<td>19%</td>
<td>4%</td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Post-Release</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Implementing Formal Review/Inspection increased design effort by 4% and decreased rework effort by 31%.

Cost: Benefit ratio is 4% : 31% or 1 : 7.75


Percentage of Rework on Projects

<table>
<thead>
<tr>
<th>Company</th>
<th>Percentage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRW</td>
<td>30%</td>
<td>(Boehm, 1987)</td>
</tr>
<tr>
<td>NASA-SEL</td>
<td>40%</td>
<td>(McGarry, 1987)</td>
</tr>
<tr>
<td>Hewlett-Packard</td>
<td>33%</td>
<td>(Duncker, 1992)</td>
</tr>
<tr>
<td>Raytheon</td>
<td>41%</td>
<td>(Dion, 1993)</td>
</tr>
</tbody>
</table>

Note: Rework = Waste or Scrap
Objectives

1. Understand the Business Rationale for Implementing Peer Reviews.
2. Understand the types of Reviews
   - Desk Check,
   - Walk-through,
   - Inspection,
   - Quality Estimation.

Formality Spectrum

- Ad hoc
- Desk Check
- Walk-through and Inspection

Review Objectives (to be identified before conducting reviews)

1. Estimate/measure quality
2. Identify defects for removal
3. Reduce cost and time of future document production (i.e. learning process)
4. Get management to take responsibility for resource approval decisions
5. Get a technological group to take responsibility for technical decisions
6. Teach writers how to follow technical standards
7. Motivate writers to follow technical standards
8. Estimate impacts (cost) of continuing with current plans (delays, rework, maintenance and defects removal costs at later stages)
9. Estimate the remaining major defects, with or without removal of the ones found in the review.
10. Stimulate the creation and contribution of better ideas, in the review or later
11. Measure the efficiency and effectiveness of processes
12. Measure the productivity and quality of organizations, teams and individuals.
13. Reduce defect content by removal of defects identified
14. Reduce the costs of testing
15. Reduce the time to market
16. Give early feedback, before too much is invested in some work
17. Determine entry criteria to a process
18. Determine exit criteria from a process

Adapted from: Tom Gilb, Review Process Design: Some guidelines for tailoring your engineering review processes for maximum efficiency, INCOSE 2008
### Review Differences

**IEEE –1028 - Standard for Software Reviews**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Management review</th>
<th>Technical review</th>
<th>Inspections</th>
<th>Walk-through</th>
<th>Audit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Presenter</td>
<td>Project representative</td>
<td>Development Team representative</td>
<td>A reader</td>
<td>Author</td>
<td>Auditors collect and examine information provided by audited organization</td>
</tr>
<tr>
<td>Data collection</td>
<td>As required by applicable policies, standards, or plans</td>
<td>Not a formal project requirement. May be done locally.</td>
<td>Strongly recommended</td>
<td>Recommended</td>
<td>Not a formal project requirement. May be done locally.</td>
</tr>
<tr>
<td>Output</td>
<td>Management review Documentation</td>
<td>Technical review documentation</td>
<td>Anomaly list, anomaly summary, inspection documentation</td>
<td>Anomaly list, action items, decision, follow up proposal</td>
<td>Formal audit report observation, feeding deficiencies</td>
</tr>
<tr>
<td>Formal facilitator training</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes (formal auditing training)</td>
</tr>
<tr>
<td>Defined participants roles</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Use of defect checklists</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Management participates</td>
<td>Yes</td>
<td>Optional</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Customer or user representative participates</td>
<td>Optional</td>
<td>Optional</td>
<td>Optional</td>
<td>Optional</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Types of Reviews
IEEE –1028 - Standard for Software Reviews

• **Reviews**
  - A software product is presented for comments or approval
  - Two types of review: Management and Technical reviews
  - **Technical Review**
    - Provides management with evidence to confirm whether:
      1. The software product conforms to its specifications
      2. The software product adheres to regulations, standards, guidelines, plans, and procedures applicable to the project
      3. Changes to the software product are properly implemented and affect only those system areas identified by the change specification
    - May also provide the recommendation and examination of various alternatives, which may require more than one meeting.
    - May be required to evaluate impacts of hardware anomalies or deficiencies on the software product.

Types of Reviews
IEEE –1028 - Standard for Software Reviews

• **Walk-through**
  - **Static analysis technique to evaluate a software product.**
  - **Major objectives:**
    1. Find anomalies
    2. Improve the software product
    3. Consider alternative implementations
    4. Evaluate conformance to standards and specifications
  - **Other objectives**
    - Exchange of techniques and style variations
    - training of the participants.
  - **Defined Roles**
    - Leader, recorder, author, team member
  - Management position ‘over’ participants shall not participate
Types of Reviews
IEEE –1028 - Standard for Software Reviews

• Inspection
  1. A visual examination of a software product to detect and identify anomalies including errors and deviations from standards and specifications.
  2. Peer examined,
  3. Led by impartial and trained facilitator
  4. Solutions are not determined during inspection meeting
  5. Author shall not act as inspection leader and should not act as reader or recorder
  6. Individuals holding management positions over any member of the inspection team shall not participate

---

Review Differences
IEEE –1028 - Standard for Software Reviews

• Purpose of Inspection
  • To detect and identify software product anomalies.
  • A systematic peer examination that:
    1. Verifies that the software product satisfies its specifications
    2. Verifies that the software product satisfies specified quality attributes
    3. Verifies that the software product conforms to applicable regulations, standards, guidelines, plans, and procedures
    4. Identifies deviations from standards and specifications
    5. Collects software engineering data (optional)
      • for example, anomaly and effort data
    6. Uses the collected software engineering data to improve the inspection process itself and its supporting documentation (optional) e.g. checklists
Review Differences
IEEE –1028 - Standard for Software Reviews

- Audit
  - An independent examination of a software products and processes to assess compliance with specifications, standards, contractual agreements, other criteria (plans).

Focus of Types of Reviews of Documents

- IEEE Standard 1028 Standard for Software Reviews

Source: Gilb & Graham, Inspection Course notes, Sept 1995.
Objectives

1. Understand the Business Rationale for Implementing Peer Reviews.
2. Understand the types of Reviews
   - Desk Check*
   - Walk-through,
   - Inspection,
   - Quality Estimation.


Desk Check Review

- **Goal:** To pass around a document and request peers to find defects, omissions, and contradictions; to improve the product; and to consider alternative implementations.
- **Note:** When the document is larger than 20 pages, it is recommended to iterate between document generation and review activities in order to detect and correct errors as soon as possible.
- **Input**
  - Document to be reviewed
- **Activities**
  1. **Author**
     1.1 Completes the upper part of the Review Form
     1.2 Distributes to selected reviewer(s) the document to be reviewed and:
        - Desk Check Review Form(s)
        - Checklist(s) (optional)
  2. **Reviewer(s)**
     2.1 Verify the document using the checklist(s) selected by the author
     2.2 Fill-in the Review Form (i.e. comments and effort taken to review the document)
     2.3 Forward Review Form to the author.
Desk Check Review

Activities

3. Author

3.1 Review the comment(s) documented on the Review Form(s)

3.1.1 - If author agrees with all the comment(s), he incorporates them in the document.

3.1.2 - If the author does not agree on comments, or if he thinks that comments have major impact on the document:

3.1.2.1 Calls a Review meeting.
3.1.2.2 Conducts the Review meeting to finalize the disposition of comments as follow:
   - Incorporate as is
   - Not incorporate
   - Incorporate with modifications

3.1.3 - Author incorporates comments in the document.
3.1.4 - Author logs on the Desk Check Review Form the effort required to review and correct the document.

EXIT CRITERIA

• Document has been corrected.

OUTPUT

• Document corrected
• Review Form completed

MEASURE

• Effort to review and correct the document (staff-hours)
Generic Document (GD) Checklist

- Requirements that must be satisfied by any document that is subject to a Desk check.
  - GD 1 (COMPLETE). All information relevant to the purpose of the document must be included or referenced.
  - GD 2 (RELEVANT). All information must be relevant to the purpose of the document and the section in which it resides.
  - GD 3 (BRIEF). Information must be stated succinctly.
  - GD 4 (CLEAR). Information must be clear to all checkers.
  - GD 5 (CORRECT). Information must be free of technical errors.
  - GD 6 (CONSISTENT). Information must be consistent with all other information in the document and its source documents.
  - GD 7 (UNIQUE). Ideas should be stated once only and thereafter referenced.
  - etc.

# CMMI – Staged Representation

<table>
<thead>
<tr>
<th>Level</th>
<th>Focus</th>
<th>Process Areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Optimizing</td>
<td>Continuous Process Improvement</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organizational Innovation and Deployment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Causal Analysis and Resolution</td>
</tr>
<tr>
<td>4</td>
<td>Quantitatively Managed</td>
<td>Quantitative Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organizational Process Performance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quantitative Project Management</td>
</tr>
<tr>
<td>3</td>
<td>Defined</td>
<td>Process Standardization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Requirements Development</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Technical Solution</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Product Integration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Verification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Validation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organizational Process Focus</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organizational Training</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Integrated Project Management for IPPD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Risk Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Integrated Teaming</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Integrated Supplier Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Decision Analysis and Resolution</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organizational Environment for Integration</td>
</tr>
<tr>
<td>2</td>
<td>Managed</td>
<td>Basic Project Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Requirements Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project Planning</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project Monitoring and Control</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Supplier Agreement Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Measurement and Analysis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Process and Product Quality Assurance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configuration Management</td>
</tr>
<tr>
<td>1</td>
<td>Initial</td>
<td></td>
</tr>
</tbody>
</table>

Source: Software Engineering Institute, Carnegie Mellon University

---

## Verification

**Purpose:** To ensure that selected work products meet their specified requirements.

Prepare for Verification

**Perform Peer Reviews**

Verify Selected Work Products

Source: Software Engineering Institute, Carnegie Mellon University
Peer Reviews

1. **Proven** mechanism for effective defect removal.
2. Develop a better **understanding** of the work **products** and the **processes** that produced them
   - Defects can be **prevented** and process-improvement opportunities can be **identified**.
3. **Methodical examination** of work products by the producers' **peers** to identify defects and other changes that are needed.
4. **Examples** of peer review methods include the following:
   - Structured **walk-through**
   - **Inspection** (e.g. Gilb, Fagan)

Objectives

1. Understand the Business Rationale for Implementing Peer Reviews.
2. Understand the types of Reviews
   - Desk Check,
   - Walk-through,
   - Inspection,
   - Quality Estimation.
Walk-through

- Find defects, omissions, and contradictions;
- Improve the product;
- Consider alternative implementations.
- Exchange of techniques, style variations, and education of the participants.
- Point out
  - Efficiency and readability issues in the code,
  - Modularity issues in design specifications
  - Requirements testability issues.
- Defect data is systematically collected and stored.
Walk-through

- Responsibilities.
  - **Author.**
    - Selects the peers to participate in the review,
    - Presents the product during the walk-through meeting.
  - **Walk-through Team.**
    - Review any input material prior to the walk-through meeting
    - Participate during the walk-through meeting to ensure that it meets its objective.
  - **Recorder.**
    - Write all comments made during the walk-through meeting:
      - Errors found, questions of style, omissions, contradictions, suggestions for improvement, or alternative approaches.

Walk-through Activities

- **WT 100. Plan the Walk-through Meeting**
  - The author completes sections of the Walk-through Form
    - Request Section and Planning Section
    - Checking Section: Name of Reviewers

<table>
<thead>
<tr>
<th>ACME Inc.</th>
<th>Walk-through Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Section</td>
<td>Filled by the author</td>
</tr>
<tr>
<td>Name of the author(s):</td>
<td></td>
</tr>
<tr>
<td>Date of request (yyyy-mm-dd):</td>
<td></td>
</tr>
<tr>
<td>Title and version of Document to be reviewed:</td>
<td></td>
</tr>
<tr>
<td>Project name:</td>
<td></td>
</tr>
<tr>
<td>Planning Section</td>
<td>Filled by the author</td>
</tr>
<tr>
<td>Charge code:</td>
<td></td>
</tr>
<tr>
<td>Meeting Room Location:</td>
<td></td>
</tr>
<tr>
<td>Walk-through Identification Number:</td>
<td></td>
</tr>
<tr>
<td>Section(s) or page(s) to be reviewed:</td>
<td></td>
</tr>
<tr>
<td>Effort:</td>
<td></td>
</tr>
<tr>
<td>Checking Section</td>
<td>Filled by the author</td>
</tr>
<tr>
<td>Name of reviewers:</td>
<td></td>
</tr>
<tr>
<td>Effort:</td>
<td></td>
</tr>
</tbody>
</table>
Walk-through - Activities

- **WT 110. Conduct a Kick off Meeting**
  - The **author**:
    - Determines if an overview of the product needs to be given to educate the reviewers of the product.
    - If overview is held: fill in Kick-off section (effort)
    - Distributes to the reviewers the product and all material

<table>
<thead>
<tr>
<th>Kick off Section</th>
<th>Filled by the author</th>
</tr>
</thead>
<tbody>
<tr>
<td># Participants</td>
<td>Kick-off duration</td>
</tr>
</tbody>
</table>

- **WT 120. Conduct Document Checking (individually)**
  - **Reviewers**
    - Prepare for the Walk-through meeting
    - Examine the product
    - Redlines on the product prior to the Walk-through meeting.
    - Prepare to discuss their comments, recommendations, questions,
    - Record the effort to review

<table>
<thead>
<tr>
<th>Checking Section</th>
<th>Filled by the author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of reviewers</td>
<td>Effort</td>
</tr>
</tbody>
</table>

---

**Generic Document (GD) Checklist (optional)**

- Requirements that must be satisfied by any document that is subject to a Desk check.
  - **GD 1 (COMPLETE)**. All information relevant to the purpose of the document must be included or referenced.
  - **GD 2 (RELEVANT)**. All information must be relevant to the purpose of the document and the section in which it resides.
  - **GD 3 (BRIEF)**. Information must be stated succinctly.
  - **GD 4 (CLEAR)**. Information must be clear to all checkers.
  - **GD 5 (CORRECT)**. Information must be free of technical errors.
  - **GD 6 (CONSISTENT)**. Information must be consistent with all other information in the document and its source documents.
  - **GD 7 (UNIQUE)**. Ideas should be stated once only and thereafter referenced.
Walk-through - Activities

1. WT 130. Conduct the Logging Meeting
   1. **Author** collects **effort** from reviewers

   - **Author** walks-through the product
   - **Team members** ask questions or raise issues on the product
   - **Recorder** writes comments and decisions for inclusion in the Issue Form
   - Author and the reviewers resolve defects discovered in the Walk-through
   - Author documents that a Walk-through was conducted on the Walk-through Form.
     - Logging section: Accept, Accept & Verify, Correct & Walk-through

<table>
<thead>
<tr>
<th>Checking Section</th>
<th>Filled by the author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of reviewers</td>
<td>Effort</td>
</tr>
</tbody>
</table>

   - **Logging section:** Accept, Accept & Verify, Correct & Walk-through

<table>
<thead>
<tr>
<th>Checking Section</th>
<th>Filled by the author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disposition of walk-through</td>
<td>Effort</td>
</tr>
<tr>
<td>Accept</td>
<td>Accept and verify</td>
</tr>
</tbody>
</table>

   - **Effort:** 15 min. = 0.25 hr

2. WT 140. Edit (Rework) the Product
   - The **author reworks** (i.e. correct and update) the product as recorded on the Issue forms during the logging meeting.
     - Completes the Editing section: effort (staff-hour)

<table>
<thead>
<tr>
<th>Editing Section</th>
<th>Filled by the author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort</td>
<td></td>
</tr>
</tbody>
</table>

3. WT 150. Complete Follow-up and Exit
   - **Author**:
     - Ensures issues and action items are tracked to closure
     - Completes the Closing section
       - **Total effort** to conduct Walk-through

<table>
<thead>
<tr>
<th>Closing Section</th>
<th>Filled by the author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Effort</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Walk-through Issue Form

Name of author: ___________________________________ Date (YYYY-MM-DD):__________________

Document Title:___________________ Name of Reviewer: ___________________

Document Version: ____________

Comment

No. | Document | Line Number / Location | Comments | Disposition of comments | Remarks |
--- | --- | --- | --- | --- | --- |
1 | | | | |
2 | | | | |
3 | | | | |
4 | | | | |
5 | | | | |
6 | | | | |
7 | | | | |
8 | | | | |

Effort to review document: _________

Effort to correct document: _________

* Disposition: Inc: Incorporate as is, NOT: Not incorporate, MOD: Incorporate with modification

Walk-through - Exit

**Exit Criteria**
- Product reworked
- Completed Issue forms
- Walk-through Form Completed

**Output.**
- Product updated
- Completed Issue forms
- Walk-through Form
Objectives

1. Understand the Business Rationale for Implementing Peer Reviews.
2. Understand the types of Reviews
   - Desk Check,
   - Walk-through,
   - Inspection,
   - Quality Estimation.

Inspection - Four Elements

1. Structured and documented process
   - Planning, checking, logging, editing, reporting
2. Defined roles
   - Attended by peers
   - Leader/moderator, author(s), checkers, recorder/scribe
3. Checklists
   - To increase probability of finding defects
     - e.g. checking for ambiguity, correctness.
4. Forms and Report
   - Provide uniformity in defect reporting, data collection and storage.

Using Source Documents, such as standards, specifications, internal process, procedures, templates, to generate a Product Document (deliverable).

Document Production Process to Produce Specifications

- Rules & Standards (Organisation Standards)
- Rules & other Standards
- IEEE Standard EN 50128
- Contract
- System Specs.
- SW Specs.

Product
Inspection Process
- Peers check the product, using checklists, to verify that the author used correctly the source documents to generate the product

Exercise
- Can you identify ‘Defects’ in this Requirement Statement?
  - “The objective is to get higher maintainability using product XYZ”
Exercise

“The objective is to get higher maintainability using product XYZ”

GD 1. They should be unambiguously clear to the intended reader.
GD 2. They shall break down complex concepts into a set of measurable concepts.
GD 3. To define 'relative' terms like 'higher' they shall specify at least two points of reference on the defined SCALE.
GD 4. They shall not mix design ideas in the specification of objectives.

**Exercise**

- Can you identify ‘Defects’ in this Requirement Statement?
  - “All the lights in any room shall have a single on-off switch”

---

**Exercise**

“All the lights in any room shall have a single on-off switch”

GD 1. They should be **unambiguously clear** to the intended reader.

GD 2. They shall **break down complex concepts** into a set of measurable concepts.

GD 3. To **define 'relative' terms** like 'higher' they shall specify at least two points of reference on the defined SCALE.

GD 4. They shall **not mix design** ideas in the specification of objectives.
Inspections Versus Tests

- Tests are dynamic verification of correct performance
  - Require equipment, test plan, procedures, and test software
- Inspections do not replace tests
- Error detection with inspection is about 10X less expensive
- Data from IBM: cost at inspection: 1 unit
cost in test: 9 X
cost with customer discovery: 117 X
- Data from Goddard Space Centre - NASA
  - Using Testing only: 5 - 17 Hours per Defect
  - Using Inspection: 1.6 Hour per Defect, (i.e. 1.1 hour to
detect + .5 hour to fix each defect).

‘Inspections find defect, while testing – which usually occurs one or more
development phase after the opportunity to inspect has passed – finds only
the symptoms’
Priscilla Fowler, Software Engineering Institute.

Inspections Versus Tests

- Typical Test Time Observed *
  - CMM level 1: 40% - 60%
  - CMM level 2: 35% - 40%
  - CMM level 3: 30% - 35%
  - CMM level 4: 25% - 30%
  - CMM level 5: 20% - 25%
- For organization below level 4**
  - Around 65%

  Test time as a percentage of the project schedule
**Inspection - Textbook**

- Also called Formal Inspection, Fagan Inspection.

---

**ETVX Process Notation**

**Process Name - Step Number and Step Title**

ETVX = Entry Task Verification eXit
Entry and Exit Criteria

- **Quality Gatekeepers**
  - Quantitative
    - ‘numeric judgement’ of whether
      - a process is sound to begin (Entry) or
      - a document is sound to release (Exit)

- **Exit Criteria**
  - A formally written condition which must be met for a task to be successfully completed.
    - Part of a process definition
    - ‘Owned’ by process owner
    - Learning from hard experience
    - Based on ‘economics’
  - **Example**
    - Estimated number of major errors OK?
    - All major errors fixed?

Adapted from: Gilb T, Inspection from a Metrics Point of View’, SEPG Conference, Bangalore India, February 2000.

---

**Inspection Process - Overview**

100 Plan Inspection
110 Conduct Kickoff Meeting
120 Conduct Document Checking
130 Conduct Logging Meeting
140 Edit Document

**Inspection - Timeframe**

- **Product document**
- **Inspection Request**
- **100 Plan Inspection**
- **110 Conduct Kickoff Meeting**
- **120 Conduct Document Checking**
- **130 Conduct Logging Meeting**
- **140 Edit Document**
- **Process improvements**
- **Source**
- **Checklists**
- **Rules**
- **Change requests**
- **Maximum – 2 calendar weeks**
- **3-4 calendar days**

Maximum – 2 calendar weeks


---

**Example of Inspection effort distribution at British Aerospace**

Summary of hours:
- **Kick-off 5%**
- **Planning 7%**
- **Checking 24%**
- **Logging 18%**
- **Edit 37%**
- **Ledadmin 9%**

Adapted from: Gilb T, 'Inspection from a Metrics Point of View', SEPG Conference, Bangalore India, February 2000.
Inspection is NOT!

1. A review of the style of a work product
2. A review of the author
   • i.e. collect performance measures about the author
3. Design Optimization
4. A meeting to fix defects or discuss possible solutions
5. Approval of quality of design
6. A finger pointing exercise
7. ‘Subjective’ error identification


Data collection

- Inspections shall provide data for:
  - The analysis of the quality of the software product, the effectiveness of the acquisition, supply, development, operation and maintenance processes,
  - The effectiveness and the efficiency of the inspection itself.
- In order to maintain the effectiveness of inspections, data from the author and inspectors shall not be used to evaluate the performance of individuals.
- To enable these analyses, anomalies that are identified at an inspection meeting shall be classified.
- Inspection data shall contain:
  - The identification of the software product, the date and time of the inspection, the inspection team, the preparation and inspection times, the volume of the materials inspected,
  - The disposition of the inspected software product.
  - This information shall be used to optimize local guidance for inspections.
  - The management of inspection data requires a capability to enter, store, access, update, summarize, and report classified anomalies.
Candidates for Inspections: Any Work Product

1. Procedures
2. Requirements
3. High-level design specifications
4. Module-level design specifications
5. User manuals
6. Plans
7. Individual test cases
8. Change Requests, Problem Reports, Fixes
9. Electrical designs, Proposals, Contracts, Drafts, etc.


Request for Inspection

Exercise

- Conduct an inspection of the product document titled ‘Conduct Brainstorm’

**Source**
- Brainstorming
  - Meeting Description

**Rules & Standards**
- ETVX Process
  - Notation

**Checklist**
- Generic
  - Document

**Product**
- Conduct
  - Brainstorm

Fill in the Request Section of the Inspection Form
Step 100 – Plan Document Inspection

1. Inspection Leader receives an Inspection Form from the author of a document.
2. Leader checks the product to verify that it meets the entry criteria (e.g. sample for high number of defects, spell check completed)
   - i.e. the document meets the Entry Conditions
3. Leader, in collaboration with Author:
   1. Identify the source documents used to generate the product.
   2. Select the checklist(s)
   3. Identify potential checkers
4. Leader plans Inspection
   1. If document is too large for one inspection cycle
      - Documents is divided in ‘chunks’
   2. Determines roles, checking rate, checkers
   3. Prepares material for kick off meeting
   4. Sends a call for kick-off meeting (if needed)
   5. Completes the Form – Planning Section

Prioritizing What to Inspect

<table>
<thead>
<tr>
<th>Requirement Phase</th>
<th>Development Phase</th>
<th>Code / Implementation Phase</th>
<th>Testing Phases</th>
<th>Maintenance Phase</th>
<th>Customer Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL Reqts</td>
<td>ALL Design</td>
<td>Quality-Critical Areas;</td>
<td>Feature Enhancements</td>
<td>Defect Fixes</td>
<td>Defect-Prone Areas</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(examples)</td>
<td>Remaining Areas, if economic payoff</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Security</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Error handling</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Algorithms</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Interfaces (ex: user)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>and Complex Areas</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Optimum Checking Rate

- **The most effective individual speed for ‘checking a document against all related documents’**.
  - *Not ‘reading’ speed but Correlation-Studying speed*
    - **Optimum Range** = 1 Logical page per hour
    - Logical page = 300 words or 100 LOC per page
  - Failure to use it
    - *Not effective* in identifying and removing *major* errors
  - **Major error**
    - Would probably have significantly (e.g. 10X) increased costs to find and fix later in the development/maintenance process
Relationship between code inspection rate and defect density- Effectiveness

![Graph showing the relationship between inspection rate and defect density.](image)


Relationship between inspection rate and defect density (i.e. Error/page)

![Graph showing the relationship between inspection rate and defect density.](image)

Individual Learning Curve

- **Individual Learning Curve**
  - The speed at which the individual learns to follow the rules,
  - As measured by reduced Major Defects found in Inspections
  - Faster, earlier and more dramatic than "process improvement"

Marie Lambertsson’s Learnability Curve,
Ericsson, Stockholm, 1997

---

**Rate Guidelines – After Many Months of Utilisation**

<table>
<thead>
<tr>
<th>Inspection Type</th>
<th>Checking Rate</th>
<th>Logging Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>2 – 3 Pages Per Hour (PPH)</td>
<td>2 - 3 PPH</td>
</tr>
<tr>
<td>Requirements</td>
<td>2 - 3 PPH</td>
<td>2 - 3 PPH</td>
</tr>
<tr>
<td>Preliminary Design</td>
<td>3 – 4 PPH</td>
<td>3 – 4 PPH</td>
</tr>
<tr>
<td>Detailed Design</td>
<td>3 – 4 PPH</td>
<td>3 – 4 PPH</td>
</tr>
<tr>
<td>Source Code</td>
<td>100 – 200 Lines of code Per Hour (LPH)</td>
<td>100 – 200 LPH</td>
</tr>
<tr>
<td>Test Plan</td>
<td>5 – 7 PPH</td>
<td>5 – 7 PPH</td>
</tr>
<tr>
<td>Fixes and Changes</td>
<td>50 – 75 LPH</td>
<td>50 – 75 LPH</td>
</tr>
<tr>
<td>User Documentation</td>
<td>8 – 20 PPH</td>
<td>8 – 20 PPH</td>
</tr>
</tbody>
</table>

Inspection Form

- Measures collected by each inspector at end of checking
  - Number of errors (anomalies)
  - Effort to review document
- Will be captured by leader at beginning of Logging meeting

<table>
<thead>
<tr>
<th>Checking Section</th>
<th>Filled by the inspection leader</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of reviewers</td>
<td>Anomalies</td>
</tr>
</tbody>
</table>

12/5/2009

Example of Roles (RO) - Optional

Checkers *could* be assigned a particular role, which ensures that they have a unique viewpoint.

- **RO 1 (CALCULATIONS):** Concentrate on all calculations, measurements, data estimates and predictions.
- **RO 2 (BACKWARDS):** Concentrate on the material from the back pages first (procedural role).
- **RO 3 (CROSS-REFERENCES):** Concentrate on all cross-references and implied corrections.
- **RO 4 (FINANCIAL):** Concentrate on cost and revenue implications, estimates uncertainty, dates, quantities.
- **RO 5 (GRAPHICAL):** Concentrate on all graphics and symbols.
- **RO 6 (INTERFACES):** Concentrate on all interfaces.
- etc.

12/5/2009

Anomaly Categories

- Anomaly categories can be representative of when the anomaly was found, its investigation, its impact, resolution activities, and final disposition.

- For example, a software documentation non conformance-type category may include the following classifications:
  - Missing
  - Extra (superfluous)
  - Ambiguous
  - Inconsistent
  - Not conforming to standards
  - Risk-prone, i.e., the review finds that, although an item was not shown to be “wrong,” the approach taken involves risks (and there are known safer alternative methods)
  - Incorrect
  - Unachievable (e.g., because of system, time, or technical constraints)
  - Editorial

IEEE Standard 1028-2008

Example of a Generic Document (GD) Checklist

- Requirements that must be satisfied by any document that is subject to inspections.
  - GD 1 (COMPLETE). All information relevant to the purpose of the document must be included or referenced.
  - GD 2 (RELEVANT). All information must be relevant to the purpose of the document and the section in which it resides.
  - GD 3 (BRIEF). Information must be stated succinctly.
  - GD 4 (CLEAR). Information must be clear to all checkers.
  - GD 5 (CORRECT). Information must be free of technical errors.
  - GD 6 (CONSISTENT). Information must be consistent with all other information in the document and its source documents.
  - GD 7 (UNIQUE). Ideas should be stated once only and thereafter referenced.
  - etc.

Example of a Generic Document (GD) Checklist

- GD 4 (CLEAR). Information must be clear to all checkers.
  1. Would your boss/peer get the correct interpretation?
  2. Would customers/suppliers understand it like engineers?
  3. Can you think of 2 or more possible interpretations?
  4. Are there any words like ‘very’, ‘good’?
  5. Are there phrases like ‘quality’, ‘performance’ undefined?

Adapted from: Gilb T, Inspection from a Metrics Point of View’, SEPG Conference, Bangalore India, February 2000.

Exercise – Develop a checklist for Requirement Specifications (RS)

- The elements of the RS checklist are not be the same as the GD checklist
- Use the following notation:
  - RS 1 (xxxxxxxx) ........
  - RS 2 (yyyyyyyy)....... 
  - RS 3 (zzzzzzzz).......
Example of a Checklist for Requirements Specification (RS)

- **RS 1 (TESTABLE)** – All requirements are verifiable (objectively)
- **RS 2 (TRACEABLE)** – All requirements must be traceable to a systems specification, contractual/proposal clause.
- **RS 3 (ELEMENTARY)** – Requirements must be broken into their most elementary form
- **RS 4 (HIGH LEVEL)** – Requirement must be stated in terms of final need, not perceived means (solutions)
- **RS 5 (QUALITY)** – Quality attributes have been defined.
- **RS 6 (HARDWARE)** – Is hardware environment is completely defined (if applicable).
- **RS 7 (SOLID)** – Requirements are a solid base for design

---

**Example of a Checklist for Requirement Specifications**

<table>
<thead>
<tr>
<th>Each Individual Requirement Should Be:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Necessary</td>
</tr>
<tr>
<td>Feasible</td>
</tr>
<tr>
<td>Correct</td>
</tr>
<tr>
<td>Concise</td>
</tr>
<tr>
<td>Unambiguous</td>
</tr>
<tr>
<td>Complete</td>
</tr>
<tr>
<td>Consistent</td>
</tr>
<tr>
<td>Verifiable</td>
</tr>
<tr>
<td>Traceable</td>
</tr>
<tr>
<td>Allocated</td>
</tr>
<tr>
<td>Design independent</td>
</tr>
<tr>
<td>Non-redundant</td>
</tr>
<tr>
<td>Stated using a standard construct</td>
</tr>
<tr>
<td>Associated with a unique identifier</td>
</tr>
<tr>
<td>Devout of escape clauses</td>
</tr>
</tbody>
</table>


Young, R., Twelve Requirements Basics for Project Success, Crosstalk, December 2006.
Example of a Checklist for Code C++ (CC)

- **CC1 (COMPLETE)** - Verify that the code covers all the design.
- **CC2 (INCLUDES)** - Verify that includes are complete.
- **CC3 (INITIALIZATION)** - Check variable and parameter initialization:
- **CC4 (CALLS)** - Check function call formats:
- **CC5 (NAMES)** - Check name spelling and use:
- **CC6 (STRINGS)** - Check that all strings are:
- **CC7 (POINTERS)** - Check that:
  - Pointers are initialized NULL,
  - Pointers are deleted only after new, and
  - New pointers always deleted after use.
- **CC8 (OUTPUT FORMAT)** - Check the output format:
  - Line stepping is proper.
  - Spacing is proper.
- **CC9 (PAIRS)** - Ensure the { } are proper and matched.
- **CC10 (LOGIC OPERATORS)** - Verify that the proper use of ==, =, //, and so on.
  - etc.


Example of a Checklist for Test Plan (TP)

- **TP 1 (Intro)** - Summarizes software items to be tested with references to project plan, quality plan and test design document
- **TP 2 (Item)** – Identify test items including version level, hardware and software requirements, and references to requirements specifications and design documents
- **TP 3 (Feature)** – List features to be tested and features not to be tested (with reasons)
- **TP 4 (Approach)** – Describe test approach to be used, including techniques and level on comprehensiveness
- **TP 5 (Criteria)** – Specify pass/fail criteria, and person responsible for the decision.
- **TP 6 (Suspend)** – Specify test suspension and resumption criteria
- **TP 7 (Environment)** – Specify test environment, including tools required, user involvement and special equipment
- **TP 8 (Schedule)** – Specify schedule, resources required, responsibilities and contingencies
- **TP 9 (Stop)** – Specify test stopping criteria
  - etc.
Inspection Form - Sections Filled by Inspection Leader

- Identification of Inspectors
  - Filled in collaboration with the author
- Identification of section(s) or page(s) to be inspected
- Identification of Roles (optional)
- Identification of Charge code (if applicable)
- Identification of Dates: Kick-off, logging.
- Attribution of an Inspection Identification Number
  - e.g. 2010 - 001
- Identification of Checklist(s)

---

Planning Section

<table>
<thead>
<tr>
<th>Checklist(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort</td>
</tr>
</tbody>
</table>

Checking Section

<table>
<thead>
<tr>
<th>Name of reviewers</th>
<th>Anomalies</th>
<th>Improvement</th>
<th>Effort</th>
</tr>
</thead>
</table>

12/5 2009

---

Step 110 – Conduct Kick-off Meeting

Step 110 – Conduct Kick-off Meeting

- **Distribute documents** to inspectors:
  - Inspection Form
  - Document to be inspected
  - Checklist(s)
  - Source document(s)
  - Attribution of roles (optional)

- **Ensure all inspectors agree with the checking effort and Logging meeting date**

- **Fill in the Form**
  - Kick off section
    - Number of participants, duration = **Effort**

<table>
<thead>
<tr>
<th>Kick off Section</th>
<th>Filled by the inspection leader</th>
</tr>
</thead>
<tbody>
<tr>
<td># Participants</td>
<td>Kick off duration</td>
</tr>
</tbody>
</table>

---

Step 120 – Conduct Document Checking

- Change requests (to source)
- 150 Complete Follow-up Exit & Release
- Inspection Request
- Product document

Step 120 - Conduct Document Checking

- **Checkers work alone** on the product document
  - Using the source documents, the rules, the procedures and the checklists provided.
- The **issues are recorded** for later personal reference,
  - e.g., for the logging meeting, **on the product document**.
- **Measures** are recorded
  - Number of pages checked, time spent, etc.
- **Inspection leader assists** checkers when needed


Example of a Checked Document

Text is line numbered

- **1. The objective is to get higher**
- **2. maintainability using product XYZ**

**Exercise: Conduct Checking**
Step 120 – Conduct Document Checking

• Task of each inspector:
  • Record issues on the product document
  • Fill in Checking section with measures:
    • Number of Anomalies
    • Effort to complete checking

<table>
<thead>
<tr>
<th>Checking Section</th>
<th>Filled by the Inspection leader</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of reviewers</td>
<td>Anomalies</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

12/5/2009

Step 130 – Conduct Logging Meeting


12/5/2009
Step 130 - Conduct Logging Meeting

Purpose

1. To log the issues which have already been identified by each checker in individual checking.

2. To identify more (i.e. new) major issues during the meeting.

3. To identify and log ways of improving the software development process or the Inspection process:
   - Improvement suggestions to procedures, rules, forms or checklists
   - Issues could also be identified in source documents

4. To make the decision to:
   - Accept with no verification or with rework verification (no major defect found)
   - Accept with rework verification (major defects to be fixed and verified),
   - Re-inspect (important rework needed and a new inspection will follow).


Step 130 - Conduct Logging Meeting

The inspection team shall identify the software product disposition as one of the following:

1. Accept with no verification or with rework verification. The software product is accepted as is or with only minor rework (for example, that would require no further verification).

2. Accept with rework verification. The software product is to be accepted after the inspection leader or a designated member of the inspection team (other than the author) verifies rework.

3. Reinspect. The software product cannot be accepted. Once anomalies have been resolved a reinspection should be scheduled to verify rework.
   - At a minimum, a reinspection shall examine the software product areas changed to resolve anomalies identified in the last inspection, as well as side effects of those changes.

Step 130 - Conduct Logging Meeting

- **Confidentiality**
  - Checkers shall *never reveal the numeric result* of an Inspection to anyone else except the author(s).
  - The author(s) may reveal the results if they want to, but they are not obliged to do so even to their direct manager (who should not even ask!).

- **Logging the issues**
  - *Round robin*
  - *Brainstorming rules* for every participant: *no criticism or discussion*
    - An issue is a violation of an item on a checklist
  - *Be diplomatic!*
  - **Author does not explain or defend**
    - Author listens, may *ask for clarifications only*
    - To learn not to repeat the same errors in the future
  - Author *checks the issue log* for legibility and intelligibility
  - Issues are logged on the Issue Log Form
    - The author, when editing, will classify each ‘issue’ as a major defect
    - *Junior inspectors may ask for clarifications* if needed to better understand the issue identified

---

**Step 130 - Conduct Logging Meeting**

- **Ways to conduct Logging**
  1. Verbatim
  2. Paraphrase
  3. Mixed styles – Verbatim and Paraphrase
  4. Section by section or line by line enumeration
    - ‘Does anyone have an issue with section X?’
  5. Perspective based
    - Inspectors stand in for specific stakeholders
  6. Line by line
    - Inspectors state: *Checklist tag number, keyword of violation.*

Step 130 - Conduct Logging Meeting

- Issues are logged at a rate of .5 to 1 per minute
- Duration – Maximum of 2 hours
- Need a quiet room, non interruption (e.g. cell phones).
- A scribe (recorder) will fill-in the Issue Log Form
  - May have to change scribe after 30-45 minutes

Anomaly ranking

- Anomalies shall be ranked by potential impact on the software product, for example, as follows:
  - **Catastrophic.** Anomalies that would result in software failure with grave consequences, such as loss of life, failure of mission, or very large economic or social loss;
    - Mitigation * is not possible.
  - **Critical.** Anomalies that would result in software failure with major consequences, such as injury, major system degradation, partial failure of mission, or major economic or social loss;
    - Partial to complete mitigation is possible.
  - **Marginal.** Anomalies that would result in software failure with minor consequences;
    - Complete mitigation is possible.
  - **Negligible.** Anomalies that would not result in software failure;
    - Mitigation is not necessary.

* Risk mitigation: a course of action taken to reduce the probability of and potential loss from a risk factor.

---

**Complete the Inspection Form**

- Inspection Leader at the end of Logging Meeting completes the Logging section:
  - Total number of anomalies collected on Issue Form
  - Disposition of inspection
    - Accept with no verification or with rework verification.
    - Accept and verify corrections
    - Reinspect
  - Effort to Log

---
Step 130 – Brainstorming Meeting – Process Improvement

Process Brainstorming Meeting Log

<table>
<thead>
<tr>
<th>Brainstorm Item no</th>
<th>Issue Ref.</th>
<th>Classification of Cause (tick)</th>
<th>Root Cause (keywords)</th>
<th>Improvement Suggestions (keywords)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Communication, Oversight, Transcription, Education</td>
<td>New design guidelines not widely known</td>
<td>Publicize in newsletter Memo to all designers</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Communication, Oversight, Transcription, Education</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Step 140 – Edit Document

The author is given the Issue Log forms and improvement suggestions to resolve.

Edit Time
   - Typically 2 to 8 minutes per defect
   - The author, when editing, will classify, on the log forms, each 'issue' as a major defect
   - The author cannot directly correct a defect in a source document
     - A Change Request (CR) to its owner is made.
   - The author should not make a large number of changes to the document
     - May require another inspection

**Inspection Form**

- Author completes the Inspection Form - Editing Section:
  - Number of issues/anomalies not corrected
  - Effort (hours) to correct the document

<table>
<thead>
<tr>
<th>Editing Section</th>
<th>Number of Anomalies NOT Corrected</th>
<th>Editing Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Step 150 – Complete Follow-up, Exit, Release**

Step 150 - Complete Follow-up, Exit and Release Activities

- **Inspection leader** checks that satisfactory author action has been taken on all logged anomalies.
- **Change Request** (CR) to correct defects in any source document must have been sent to the owner of the document.
- Process improvement suggestions should be sent to the process owner.
- **Exit process**
  1. Exit criteria are satisfied *
  2. If a document has been divided up to be inspected in separate inspection cycles, all chunks must have exited before the document as a whole can exit Inspection.
  3. Complete the Form.

---

Step 150 - Complete Follow-up, Exit Criteria

1. All editing has been completed
   - Inspection Leader has successfully validated the edit activities.
2. All necessary Change Requests (CR) cross-references are inserted in the product document
   - To make sure we do not forget to edit product document if a defect is acknowledged in source document.
3. **Inspection Database** has been updated.
4. Check that the rate (pages or lines per hour) of individual checking and of the logging meeting did not exceed the known optimum rate by more than 20% on average.
5. Either the author or the Inspection Leader can veto the exit based on their subjective belief that release would be a danger to their document customers.
6. There are no more than 0.25 (2 to 3 for beginners) estimated/measured major defects per page remaining.
Step 150 - Complete Follow-up, Exit and Release Activities

- Calculate the Number of remaining defects per page
  - Example: 16 major defects found in a 36 page document with an effectiveness of 60% and errors in defect correction of 17%.*

\[
\text{Total defects} = \frac{\text{Number of Major Defects}}{\text{Effectiveness}} = \frac{16}{.60} = 27
\]

\[
\text{Remaining defects (REM)} = \text{Total} - \text{Major Defects} = 27 - 16 = 11
\]

\[
\text{Not-fixed (NF)} = \text{Number of Major} \times 17\% = 16 \times .17 = 3
\]

\[
\text{Remaining defect per page} = \frac{\text{REM} + \text{NF}}{\text{No. pages}} = \frac{11 + 3}{36} = \frac{14}{36} = .38
\]

* Fix Fail Rate

Inspection Form

- Inspection leader completes the Closing Section of the Form:
  - Total number of defects corrected
  - Effort to close the inspection (Follow-up, Exit and Release)
  - Total effort to do the inspection and correct the defects
  - Comments (if applicable)

<table>
<thead>
<tr>
<th>Closing Section (Follow-up, Exit and Release)</th>
<th>Filled by the Inspection leader</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Errors Corrected:</td>
<td>Closing Effort:</td>
</tr>
<tr>
<td>Comments:</td>
<td>Total Effort:</td>
</tr>
</tbody>
</table>

Adapted from: Gilb, Graham, Software Inspection Course notes, 1995.
Commitment Requested from Management

- **Support**
  - **Resource and time** to perform inspections
  - Inspections should be mandated in project plans
    - Start with most important documents
      - e.g. specifications, design.
    - If a project has already started
      - Inspect document produced in current development phase

- **Confidentiality** of individual inspection data
  - Request access only to global measures
    - e.g. number of documents inspected, number of defects per page, savings from inspections, process improvement suggestions.
  - Inspection data is **not used for performance evaluation**.
  - Inspection data is **not attributed to an individual**.

---

Understanding the Quadrant Chart

![Defect Density Vs Preparation Rate Chart]

*Source: Ed Weller, Inspection Data: Now that I have it, what do I do with it?*
### Understanding the Quadrant Chart

<table>
<thead>
<tr>
<th>Defects Found Rate</th>
<th>Preparation Rate</th>
<th>Quality of Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>High</strong></td>
<td><strong>Low</strong></td>
<td><strong>Potentially Low</strong></td>
</tr>
</tbody>
</table>
|                    |                  | ▪ Well conducted inspection  
|                    |                  | ▪ Poor quality code  
|                    |                  | ▪ A re-inspection may be needed  |

<table>
<thead>
<tr>
<th>Defects Found Rate</th>
<th>Preparation Rate</th>
<th>Quality of Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Low</strong></td>
<td><strong>Low</strong></td>
<td><strong>Acceptable</strong></td>
</tr>
</tbody>
</table>
|                    |                  | ▪ Well conducted inspection  
|                    |                  | ▪ Higher quality code  
|                    |                  | ▪ A re-inspection is not usually needed  |

<table>
<thead>
<tr>
<th>Defects Found Rate</th>
<th>Preparation Rate</th>
<th>Quality of Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Low</strong></td>
<td><strong>High</strong></td>
<td><strong>Potentially Low</strong></td>
</tr>
</tbody>
</table>
|                    |                  | ▪ Poorly conducted inspection  
|                    |                  | ▪ Poor quality code, even after the inspection  
|                    |                  | ▪ A re-inspection should almost always be done (or perhaps the code has to be rewritten)  |

<table>
<thead>
<tr>
<th>Defects Found Rate</th>
<th>Preparation Rate</th>
<th>Quality of Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>High</strong></td>
<td><strong>Low</strong></td>
<td><strong>Low</strong></td>
</tr>
</tbody>
</table>
|                    |                  | ▪ Well conducted inspection  
|                    |                  | ▪ Poorly conducted inspection  
|                    |                  | ▪ Defects may have been undetected.  
|                    |                  | ▪ Could also mean the code was very simple, easily understood, or had a lot of “cut and paste” structure to it.  
|                    |                  | ▪ A re-inspection may be indicated depending on the team analysis of the high preparation rate.  |

---

**Ed Weller, Inspection Data: Now that I have it, what do I do with it?**

---

### Inspection Process - Summary

- **Product document**
- **Checklists**
- **Process improvements**
- **Source**
- **Change requests (to source)**
- **100 Plan Inspection**
- **110 Conduct kickoff Meeting**
- **120 Conduct Document Checking**
- **130 Conduct Logging Meeting**
- **140 Edit Document**
- **150 Complete Follow-up Exit & Release**

### Example of a Peer Review Document Selection Matrix

<table>
<thead>
<tr>
<th>Technical Drivers - Complexity</th>
<th>Low</th>
<th>Average</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Product</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
<td><strong>Walk-through</strong></td>
</tr>
<tr>
<td>Software Requirements</td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
</tr>
<tr>
<td>Design</td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
</tr>
<tr>
<td>Software Code and Unit Test</td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
</tr>
<tr>
<td>Qualification Test</td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
</tr>
<tr>
<td>User/Operator Manuals</td>
<td><strong>Desk check</strong></td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
</tr>
<tr>
<td>Support Manuals</td>
<td><strong>Desk check</strong></td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
</tr>
<tr>
<td>Software Documents, e.g. Version Description Document (VDD), Software Product Specification (SPS), Software Version Description (SVD)</td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Walk-through</strong></td>
</tr>
<tr>
<td>Planning Documents</td>
<td><strong>Walk-through</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
</tr>
<tr>
<td>Process Documents</td>
<td><strong>Desk check</strong></td>
<td><strong>Walk-through</strong></td>
<td><strong>Inspection</strong></td>
</tr>
</tbody>
</table>

### Inspection Effectiveness

<table>
<thead>
<tr>
<th>CMM Maturity Level</th>
<th>Defect Removal Effectiveness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 5</td>
<td>90% +</td>
</tr>
<tr>
<td>Level 4</td>
<td>75% - 90%</td>
</tr>
<tr>
<td>Level 3</td>
<td>65% - 75%</td>
</tr>
<tr>
<td>Level 2</td>
<td>50% - 65%</td>
</tr>
<tr>
<td>Level 1</td>
<td>Less than 50%</td>
</tr>
</tbody>
</table>

Return on Investment (ROI) of Inspections

• Total Net Savings (staff-hours) from major errors =
  \[ ((110 \times \text{Requirement}_\text{Error}) + (61 \times \text{Design}_\text{Error}) + (41 \times \text{Code}_\text{Error}) + (5 \times \text{Doc}_\text{Error}) + (5 \times \text{Change}_\text{Request}_\text{Error})) - \text{Tot}_\text{Find}_\text{and}_\text{Fix}_\text{Cost} \]

– Savings (over 8 months) = 9 staff-weeks per inspection
  – Effort to ‘find and fix’ per defect = 1.5 to 3 hours


Organizational Policy

1. REVIEW TO RULE: All technical specifications will be reviewed against their standards (rules, templates, concept definitions, exit and entry conditions), and with regard to related specifications (like requirements which were input to the spec writing process).
2. QC BY SAMPLING: At the minimum a sufficient sample of a large specification will be taken, in order to determine its defect density.
3. REVIEW RIGOROUSLY: The reviews will be conducted with corporate defined rigor, and sufficient rigor for management to trust their conclusions.
4. DATA DRIVEN REVIEWS: The data collected from reviews will be made available for the purpose of improving the review process, and the related specification processes.
5. ENTRY CONTROL: There will be intelligently defined entry conditions to start reviews, that will prevent us wasting time on them.
6. EXIT CONTROLS: There will be intelligently defined formal exit conditions from reviews that will be respected, and will prevent us from approving work under pressure, which is going to cause problems later.
7. QUALITY MEASURED: The primary exit condition for a specification shall be based on a realistic measure of major defects/page, and a realistic estimate of remaining defects per page after corrections. If there is any doubt that the level of exit defects standard has been met, a new sample review will be used to determine safe exit.
8. DEFECT DENSITY MANAGEMENT: The purpose of most reviews will primarily be to determine if the specs are of corporate standard quality, in terms of major defects possibly remaining per page. Approval decisions will not be made in the review process itself, but will use the quality data from the review (major defects/page remaining) as partial input to an approval process.
9. AVOID CLEANUP: The purpose of reviews should never be to clean up bad engineering specification work. The reason is that the best review processes are very ineffective in cleaning up (20%-60% level). Bad work needs to be re-done properly.
10. REVIEW EARLY: We need to do partial reviews early in the production of a large quantity of specification. For example after a week of work, a random, representative sample of specs should be taken, especially of untried and unproven individuals, teams or suppliers. We need to manage QC so well that we never have a large amount of work that is done substandard.

Inspection Pitfalls

Successful implementation & performance is at risk when these occur:

- **Lack of** supportive Software Development Life-Cycle infrastructure
- **Poor management understanding** of the inspection process, its benefits, and their responsibilities
- No computerized management planning tools
- **Too little schedule time** for inspections; No criteria for selecting areas to inspect
- No computerized Inspector tools
- **Inadequate tools for monitoring** of inspection execution and tracking of results
- No post-class practitioner refresher
- **No inspection facilitator** / project champion
- Slow inspection implementation by project teams
- No Inspection process capture

Source: Stewart, R., Priven, L., How to Avoid Software Inspection Failure and Achieve Ongoing Benefits, Crosstalk, January 2008

Objectives

1. **Understand the Business Rationale for Implementing Peer Reviews.**
2. **Understand the types of Reviews**
   - Desk Check,
   - Walk-through,
   - Inspection,
   - Quality Estimation.
 Inspection Versus Quality Estimation Review

- **Inspection**
  - To detect and identify anomalies including errors and deviations from standards and specifications.
  - Source: IEEE-1028 Standard.

- **Quality Estimation Review**
  - To rapidly, i.e. within less than an hour, estimate, from a sample, the defect density for management decision about the release of a document.


---

Case-Study: System Requirements Specification

- **Major US Corporation.**
  - Jet Engine Manufacturer

- **82 page document, 2 pages were sampled and reviewed by 4 managers.**

- **Rules used for Requirements:**
  1. Unambiguous to intended Readership
  2. Clear enough to test.
  3. No Design.

- **Results**
  - For page 81 (¼ page) 15 Majors, 15, 20, 4.
    - Page 81 had about 40 total unique major defects.
  - For page 82 (a full page) 24 majors, 15, 30 and 3 majors/page.
    - Page 82 had about (2 x 30) 60 majors /page.

Case-Study: System Requirements Specification

- **Results**
  - Rule of thumb: checkers find one third of defects
    - This indicates an average of **150 major errors per page**.
  - An estimation that we have about 150 (average per physical page) x 82 (total pages) = **12,300 Majors total**.
  - Not all major defects in specs lead to a error.
    - A rule of thumb, that correlates well with observed reality
      - One third of the major defects will cause errors in the system.
  - That implies that **4,100** (i.e. 12,300/3) errors will occur.


Case-Study: System Requirements Specification

- **Results**
  - The median downstream cost of not finding them would have been **9.3 hours per error** (range up to 80 hours).
    - 10 hours was used as a rough number.
  - This implies **41,000 hours effort lost** in the project through faulty requirements!
  - ‘We have 10 people on the project, using about 2,000 hours/year, and the project is already 1 year late (i.e. 20,000 hours) and we have at least one more year (i.e. 20,000 hours) of correcting the problems before we can finish.’
    - Total of roughly **40,000 hours**!

Quality Estimation Review (QER)

- Peers check the product, using checklists, to verify a **sample** of the product.

**Inputs**

- Set of checklists

**Activities**

1. Identify Checkers.
2. Select Rules.
3. Choose Sample.
4. Instruct Checkers
5. Check Sample.
6. Report Results.
7. Analyze Results.

**Outputs**

- Document: if document meets the exit criteria.
- Decision:
  - To rewrite the document, or
  - To Inspect the document using the Inspection Process.

**Entry Criteria**

Sufficient time to complete an ARP: total elapse time of 30 to 60 minutes. There is a trained leader to facilitate the ARP.

**Measures**

- Effort to perform ARP (Staff-hours).
- Number of estimated number of defects per page.

**Exit Criteria**

- If less than 5 majors/page extrapolated total density, or if a decision to ‘rewrite’ the document has been agreed.
Quality Estimation Review - 1

1. Identify Checkers.
   • QER leader selects about two people.

2. Select Rules.
   • The group identifies about three rules to use for checking.

3. Choose Sample(s).
   • The group selects sample(s) of about one page in length (300 non-commentary words).
     • Choosing a page at random can add credibility – so long as it is representative of the content subject
     • The group should decide whether all the checkers should use the same sample or whether different samples are more appropriate.

4. Instruct Checkers.
   • QER leader briefly instructs the checkers about the rules, the checking rate, and how to document issues and how to determine major defects.


Quality Evaluation Review - 2

5. Check Sample.
   • Checkers use between 10 and 30 minutes to check their sample against the selected rules.
   • Checkers should ‘mark up’ their copy of the document as they check
     • underlining/circling issues.
   • At the end of checking, each checker should count the number of ‘possible majors’ they have found in their page.

6. Report Results.
   • QER team leader leads a discussion to determine how many of the ‘possible majors’ are actually likely to be majors.
   • Each checker determines their number of majors and reports it.

7. Analyze Results.
   • Estimate the major defect density.
     • 6 times the highest number of major errors found by a single person,
     • Or 3 times the unique major defects found by a 2 to 4 person team.
     • If using more than one sample, average the densities found by the group in different pages.
   • The QER leader extrapolates the number of major defects in a single page.
     • Total number of major defects
       • Multiply this average major defects/page density by the total number of pages of the document.

   • If the number of major defects/page found is 10 or more,
     • Determine how to get someone to write the document properly.
     • There is no economic point in looking at the other pages to find ‘all the defects’, or correcting the majors already found.
   • If less than 10 defects per page
     • Inspect
     • Release document

   • Choose any major defect and think for a minute why it happened.
   • Give a short sentence to capture your idea.
Quality Estimation Review Form

Date Started (dd/mm/yy): ____
Leader: Tom
Author: Tino
Other Checkers: Arthur
Document Reference: Test Plan V 2.0 Total Physical pages: 10
Document Sample Reference: page 3
Rules Used: Generic Rules, Test Plan Rules
Sample Size: ~300 (Non commentary words)
Checking Time Planned: 30 minutes
Major Defects Identified: 6, 8, 3
Estimated Unique Majors Found by Team: about 16
Estimated Average Majors/Logical Page: 16 x 3 = 48
Estimated Total Majors in Document: 48 x 10 = 480
Recommendation: no exit, redo and resubmit
Causes (of defect level): author not familiar with rules
Actions suggested to mitigate Causes: author studies rules, all authors given training in rules
Responsible for Action: project manager
Completion Date/Time: May 29 2003 18:08


Example of a Completed Form

Date Started: May 29 2003
Leader: Tom
Author: Tino
Other Checkers: Arthur
Document Reference: Test Plan V 2.0 Total Physical pages: 10
Document Sample Reference: page 3
Rules Used: Generic Rules, Test Plan Rules
Sample Size: ~300 (Non commentary words)
Checking Time Planned: 30 minutes
Major Defects Identified: 6, 8, 3
Estimated Unique Majors Found by Team: about 16
Estimated Average Majors/Logical Page: 16 x 3 = 48
Estimated Total Majors in Document: 48 x 10 = 480
Recommendation: no exit, redo and resubmit
Causes (of defect level): author not familiar with rules
Actions suggested to mitigate Causes: author studies rules, all authors given training in rules
Responsible for Action: project manager
Completion Date/Time: May 29 2003 18:08

Case-Study: Air Traffic Control Simulator

- Sweden/Germany Company
- **Contract dictated** about 80,000 pages of logic specifications.
  - Supplier had written and approved about 40,000 pages of these.
    - The next stage for the logic specs was writing the software.
- Pulled a sample of 3 random pages from the 40,000
  - Sample was representative of the others.
- Managers found 19 ‘major defects’ in the 3 sample pages.
- Director took 30 minutes to check the 19 defects personally, while his managers and Tom Gilb waited in his office.
  - Director said, ‘If I let one of these defects get out to our customer, the CEO would fire me!’


---

Case-Study: Air Traffic Control Simulator

- The managers had signed off on about 0.8 million errors (bugs), i.e. 20 x 40,000 = .8 million
  - One manager who signed the third signature on the spec approval, was asked why he signed off on what we all acknowledged was a tragedy. He told me it was because ‘the other managers signed it ahead of him’.
- **Conclusion**
  - The company had a ‘factory’ of analysts producing about 20 major defects per page of specification.
  - The project got completed; but only after being sold off to another industry.
  - The director lost his job.

Limitations of Quality Estimation Review

1. It is only a small sample so the accuracy is not as good as a full or larger sample.
2. A team of two people does not have the known effectiveness of 3 or 4 people.
3. You will not have the basis for making corrections to the entire specification.
4. The checking will not have been carried out against all the source documents.
   - Usually no source documents are used and memory is relied on. While this means that the checking is not nearly as accurate, it does considerably speed up the process.
   - If the sample turns up a defects density estimation of 50 to 150 major defects/page, that is more than sufficient to convince the people participating, and their managers that they have a serious problem.


Bibliography

- Articles from CROSSTalk - The Journal of Defense Software Engineering
  - How to Avoid Software Inspection Failure and Achieve Ongoing Benefits, January 2008

- Articles from IEEE
  - IEEE Software Magazine
    - http://www.computer.org/software/
Books

- “High Quality Low Cost Software Inspections”
  - Ronald A. Radice
  - Paradoxicon Pub, 2002.

- ‘Peer Reviews in Software - A Practical Guide’
  - Karl Wiegers

Web Sites

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

- Inspection Support Centre
  - http://www.cs.strath.ac.uk/research/efocs/assist.html

- Process Impact from Karl Wiegers
  - http://www.processimpact.com/

- US Navy
Objectives

1. Understand the Business Rationale for Implementing Peer Reviews.
2. Understand the types of Reviews
   - Desk Check,
   - Walk-through,
   - Inspection,
   - Quality Estimation.

‘One year of experience working in an environment that regularly requires participation in peer-reviews is equivalent to 3 years of experience working in a non-reviewing environment’

Why is inspection so difficult to use?

'It's because of the way (software engineers have) been trained and managed. Starting with first programming courses, they learn that the most admired programmers produce code at lightning speed and fix defects in testing. This fix-it attitude fosters poor practices. To get quality work, you must change this culture.'


‘Survival is not compulsory’

W. Edwards Deming