Software Inspections and Walkthroughs

Software Inspections

- “Software Inspections are a disciplined engineering practice for detecting and correcting defects in software artifacts, and preventing their leakage into field operations.” Don O'Neill, Don O' Neill Consulting for SEI

Software Walkthroughs

- a form of software peer review "in which a designer or programmer leads members of the development team and other interested parties through a software product, and the participants ask questions and make comments about possible errors, violation of development standards, and other problems" (IEEE Std. 1028-1997, *IEEE Standard for Software Reviews*, clause 38.

What's the difference?

- An inspection is a more formal process than a walkthrough used to collect metrics or statistics about the software process
- Walkthrough is a more informal version of an inspection

Why inspect software?

- Routine production of reliable software within budget and on schedule continues to elude most development organizations.
- While efforts are ongoing to make development an industrial process, much of the work is still done by “intellectual artisans”
- Artisan’s work is inherently difficult to bond and can not be specified precisely.
- Inspections are a method to reduce variability and tighten process control.

History of Inspections

- M.E. Fagan of IBM first defined inspections in 1976.
- E. Yourdon was among the first to publish a book on inspections in 1978.
- IEEE standard covering inspections first appeared in 1988.

What do inspections cover?

- Inspections and walkthroughs are primarily intended to discover defects in software artifacts.
- This is a static analysis technique of software testing.
- In addition, inspections address three major tasks of process management: planning, measurement, control.

Inspection metrics

- Inspections are used to collect quantitative quality data at defined points in the development process.
- This can be used to give feedback to the developers, feed-forward to future development, and feed-into future steps of process.
- Can also provide data on effectiveness of inspection techniques.

What can be inspected?

- Inspections can be held a various points in development process.
- Fagan recommended inspections on:
  - Detailed design
  - Cleanly compiled code
  - Completion of unit test

Who is involved?

- At a minimum a formal inspection includes:
  - Designated moderator
  - Author of the work
  - At least one peer inspector
- Walkthroughs generally do not include designated moderator and are often led by the author of the software.

Steps of inspection

- Planning
- Overview
- Preparation
- Meeting
- Rework
- Follow-up

Planning

- Planning begins when entry criteria for inspection type is met.
- Moderator is selected – usually a peer or technical leader
- Selection may be made by developer, but this is generally not an ideal situation
- Management is encouraged not to look at individual inspection results
- Moderator verifies that product meets entry criteria and schedules future steps.

Overview

- Presentation to inspectors with any background information needed to properly review software product.
- Purpose is educational only
- Data collected is author preparation time and time spent on presentation.

Preparation

- Individual activity
- Author collects all material required for inspection
- Inspectors study the material and complete inspection log.
- Defects are noted at this step, but not collected

Meeting

- Meeting is conducted by moderator
- Agenda includes:
  - Introduction
  - Establishing readiness
  - Examining material and recording defects
  - Review defects
  - Determine disposition
  - Debrief
- Defect data is collected this time

Common meeting problems

- Interpersonal tensions are most likely to arise at this point
- Experienced moderators can detect and defuse this tension
- The more inspections that occur, the less likely interpersonal tensions are to interfere
- Effort should be made by all participants to keep emphasis on producing quality product, not making fault finding personal

Rework

- Performed by the author in response to defect disposition determined at meeting

Follow-up

- Moderator verifies that corrections are made
- Moderator completes inspection management report and defect summary report

Inspection Roles

- Author – developer of work product
- Moderator – an inspector responsible for organizing and reporting on inspection
- Reader – an inspector who guies the examination of the product
- Recorder – an inspector who enters all the defects found on the defect list
- Inspector – Member of inspection team. Often chosen to represent specific role- designer, tester, technical writer, SQA, etc

Inspection as Process Control

- When employed at various points through out the process, the completion of an inspection can trigger entry into a new development phase.
- Generally, Software Development Plan spells out entry and exit criteria and required participants in each type of inspection.

Aspects of inspections

- Initial introduction of inspection into an organization can cause anxiety and tension among developers
- When it becomes clear that management supports inspection as a quality improvement technique and not a witch hunt, the effectiveness of the inspection increases.

Inspection Data

- The collection and analysis of data is what sets inspections apart from other peer review techniques such as walkthroughs.
- This data can be used in a variety of ways by a variety of personnel.

Data customers

- First-line managers – amount of rework generates schedule information
- Next phase developers or verifiers get “intelligence” report on status of software
- Quality assurance personnel use data on amount of material inspected, amount of inspection material, speed of examination to examine inspection effectiveness

More data usage

- Quality assurance is responsible for recommending inspection and preparation rates – actual review data makes these more realistic
- Defect rates and types discovered at different points can point to most effective place to review. For example, design inspections may prove more cost effective than code.

Alternatives

- There is a “cost of quality” associated with walkthroughs and inspections. In software, person-hours are the highest measurable expense
- Many organizations find that the cost of inspection does not generate a return on investment
- Some inspect a percentage of code
- Others inspect only critical portions

Conclusions

- Inspections have been proven an efficient and effective method for improving software quality
- In conjunction with testing, audits and formal verification a successful, quality product can be produced

My opinion

- When done correctly, walkthroughs and inspections are valuable defect finding tools.
- When not supported by management or bought into by development personnel, they become “busy work” for developers.
- It is important for developers to not take criticism personally.
- It is equally important for inspectors to look for defects and not criticize because developer didn’t code exactly the way they would

References

- http://www.sei.cmu.edu/str/descriptions/inspections_body.html
- IEEE Std. 1028-1997, *IEEE Standard for Software Reviews*