ANALYZING INSPECTION DATA FOR HEURISTIC EFFECTIVENESS

Forrest Shull, Carolyn Seaman*, Madeline Diep

Fraunhofer CESE – College Park, MD
* Fraunhofer CESE and University Maryland Baltimore County
Outline

• Why inspection?
• Why analyze inspection data?
• An analysis example and lessons learned
• Parting thoughts
Formal Software Inspections

- A structured process for finding & fixing defects
- Used to remove defects as early in development as possible
- A *simplified* model:
Why do inspections matter?

- Formal inspection is applicable to all development phases:
  - One of few assurance techniques that are available in the earliest phases
  - A long history of research and experience shows it is one of the most cost-effective practices for achieving quality software:
    - “Cost savings rule”: Cost to find & fix software defects is about 100x more expensive after delivery than in early lifecycle phases.
      - IBM: 117:1 between code and use
      - Toshiba: 137:1 between pre- and post-shipment
      - Data Analysis Center for Software: 100:1
    - “Inspection effectiveness rule”: Inspections find over 50% of the defects in an artifact, regardless of the lifecycle phase applied.
      - 50-70% across many companies (Laitenberger)
      - 64% on large projects at Harris GCSD (Elliott)
      - 60% in PSP design/code reviews (Roy)
      - 50-95%, rising with increased discipline (O’Neill)
Why analyze inspection data - Scenario 1

I'm going to run an inspection, and I'm not entirely comfortable with all the decisions I have to make: inspection team, page rate, etc.

I'm working on mission-critical software and want to maximize number of defects found.

I'm working on a scientific prototype and want to maximize efficiency.

• **How the analysis helps:** Analyzing past inspection can yield heuristics (for team size, page rate) that can be a starting point for inspection planners:
  • There can be separate heuristics depending on project goal (effectiveness or efficiency)
  • Don’t need to be enforced strictly, but provide a basis from which to define deviations
Why analyze inspection data - Scenario 2

Somebody performed inspections and I'm trying to understand how much confidence to have in the quality of the software as a result.

- **How the analysis helps**: allows comparisons between different inspection instances, and evaluation of likely effectiveness
  - How likely is it that the inspection did a good job of finding a majority of defects based on what we've seen from other inspections with similar planning parameters?
An analysis example - motivation

- In early-90s, a large organization identified heuristics for key parameters based on best practices and early inspection data focusing on maximizing defect detection.

<table>
<thead>
<tr>
<th>Team Size</th>
<th>Too small: miss important expertise</th>
<th>Rule of thumb: 4-6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Too large: drive up costs, dampen discussion</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page Rate</th>
<th>Too large – thorough review impossible</th>
<th>Rule of thumb:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>For reqts: &lt;=15 pg/hr</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For code: &lt;=10 pg/hr</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For design and test: &lt;= 20 pg/hr</td>
</tr>
</tbody>
</table>

| Meeting Length     | Too long: danger of fatigue           | <2 hours |

- Software development environments have changed a lot in the past 20 years.
- Economic climate shifts focus toward efficiency - “Do more with less”.
- Are these heuristics still applicable in contemporary setting?
  - Are they still being followed?
  - Do they still provide the same significant benefit as when they were initially conceived?
  - How do the heuristics fare in providing an efficiency benefit?

We analyzed a large set of inspection data to answer these questions.
Our Inspection Data

- data from 2700+ inspections in a governmental organization developing complex and highly critical software system.
- Data spanned across 19 years (1988-2007).
- We scrubbed and verified the data.

2722 records
Our Inspection Data

- 2700+ inspections data from a governmental organization developing complex and highly critical software system.
- Data spanned across 19 years (1988-2007).
- We scrubbed and verified the data.

- 750 records
- Meeting length outliers (> 4 hours)
- Did not report # defects found
- Retained records with extreme values, e.g: page rates of > 100 pg/hour, meeting length < 30 min
- 194 records
- 1778 records
Analysis Approach

• Split inspection data into two sets.

• For each set – calculate and statistically compare inspection performance for when inspections complied with heuristics (in-compliance) and when they did not comply with heuristics (out-compliance).
  • Performance measured in term of effectiveness (# of defects) and efficiency (# of defects/hour)
Analysis Results – Degree of adherence

Observation – With the exception of meeting length, current inspections tend not to follow heuristics:
- Contemporary inspections tend to employ smaller team size
- 62% (83) of inspections that were not compliance with page rate work inspected > 50 pages/hour

Observation – Only small percentage of inspections follow all heuristics, for both historical and contemporary inspections

Analysis follow up:
- Investigate why fewer teams follow the heuristics.
- Investigate impact of not adhering to the heuristics.
- Determine if heuristics need to be updated.
Analysis Result - Inspection Effectiveness

<table>
<thead>
<tr>
<th></th>
<th>Team Size</th>
<th>Meeting Length</th>
<th>Page Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Historical</td>
<td>↑</td>
<td>↓</td>
<td>↑</td>
</tr>
<tr>
<td>Contemporary</td>
<td>↑</td>
<td>↓</td>
<td>Ω</td>
</tr>
</tbody>
</table>

- Being in compliance increases effectiveness
- Being in compliance reduces effectiveness
- No observed difference between being in- or out-of-compliance

![Warning Icon]
Although effectiveness seems to decrease, too few inspections actually deviated from the heuristic to be confident of the trend

Observation – Page rate heuristic seems to be a viable candidate to be updated considering (1) fewer inspections follow the current heuristic and (2) no difference in result when the heuristic was followed.
Analysis Result - Inspection Efficiency

<table>
<thead>
<tr>
<th></th>
<th>Team Size</th>
<th>Meeting Length</th>
<th>Page Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Historical</td>
<td>![Down Arrow]</td>
<td>![No Change]</td>
<td>![Down Arrow]</td>
</tr>
<tr>
<td>Contemporary</td>
<td>![No Change]</td>
<td>![No Change]</td>
<td>![No Change]</td>
</tr>
</tbody>
</table>

- Being in compliance increases effectiveness
- Being in compliance reduces effectiveness
- No observed differences between being in- or out-of-compliance

Observation – Heuristics did not seem adequate to address the efficiency goal. Additional analyses should be done to provide a set of planning heuristics that are more successful in providing cost-effective inspections.
Analysis next steps:

- Re-formulate guidelines for the three points of control.
  - Understand the interaction between the planning parameters.
  - Identify “sweet spot”, if it exists.

- Investigate the effect of context variables (e.g. project characteristics) when planning inspections.

- Gather more inspection data and generalize our analysis to larger set of data.
Parting Thoughts

• Many benefits from gathering and analyzing inspection data:
  • Establish best practice and guidance, backed by concrete evidence, for performing “good” inspections.
  • Establish baseline for evaluating inspection performance.
  • Enable continuous improvement and response to changes.

• Take analysis results with a grain of salt
  • Be aware of data collection and analysis perils that can threaten accuracy of analysis.
Questions?