Fagan Inspection and Gilb & Graham Inspections

Software Quality Assurance

ANU/IT/2021/F/053

I.G. Iresha Kaushalya Chandrarantha

FAGN INSPECTION

INTRODUCTION

One way to locate flaws is through Fagan Inspection. It is a methodical procedure that entails finding errors in development documents at different stages of the software development life cycle, such as designs, specifications, and programming code. Michael Fagan, who integrated a number of formal software inspection techniques, is the one who gave rise to the term "Fagan." A certain activity has predetermined entry and exit criteria when using the Fagan inspection method. Therefore, the Fagan inspection process provides a means of verifying that the output satisfies the exit criterion meant for each activity that has entry and exit requirements.

We describe in this article a software development group's usage of Fagan's inspection technique to verify and validate software requirements specifications for functions inside a large production software package. The package in question has more than 250 different high-level geophysical functions, more than 2 million lines of Fortran code, and runs on a range of hardware configurations. The package is frequently updated. A new version of the product is made available to users twice a year. A normal release involves at least 25% of the overall code changed in some way, whether it's for bug patches, new processing functions, or the removal of out-of-date ones.

FAGN INSPECTION PURPOSE

The main objective of the Fagan Inspection is to find flaws early in the development process, which lowers the expense and time needed to remedy mistakes discovered later in the cycle or after release.

1. Defect Detection: Finding defects in a variety of work products, including as requirement documents, design documents, source code, and test plans, is the main goal of the Fagan inspection process. Through close examination of these artifacts, teams may identify problems early on and stop them from developing into bigger ones down the road.

2. Quality Assurance: Fagan checks help ensure that the software product is built with quality from the beginning. Instead of addressing quality issues after the fact, this process makes sure that every stage of creation complies with strict guidelines. The total quality attained at every phase determines the quality of the finished software product.

3. Cost-Effectiveness: As faults move through the development cycle, fixing them gets more and more expensive. Fagan inspections save total costs and improve the project's economic sustainability by identifying problems early on.

4. Independent Evaluation: Deliverables are carefully examined by experts who are not affiliated

with the original author. This objective evaluation guarantees that the program satisfies requirements, follows design guidelines, and serves the intended purpose.

FAGN INSPECTION RROCESS

a method of searching for errors in documents (like source code or formal specifications) at different stages of the software development life cycle. It bears Michael Fagan's name, who is recognized as the creator of formal software inspections.

The seven separate steps that make up the Fagan inspection process each contribute to the overall efficacy of flaw identification and removal:

1. Planning: By outlining roles, duties, and goals, the team gets ready for the inspection. The moderator oversees the inspection and makes sure that the protocol is followed.
The team defines the goals, parameters, and materials needed for the inspection during this first stage. It entails establishing precise objectives and success criteria.

2. Overview: The author gives a summary of the deliverable, emphasizing its main points, background, and important components. Ensuring that all participants comprehend the information is the aim of this step.
The team gains a broad understanding of the procedure or item under inspection at this step. They collect the required data and examine pertinent documentation.

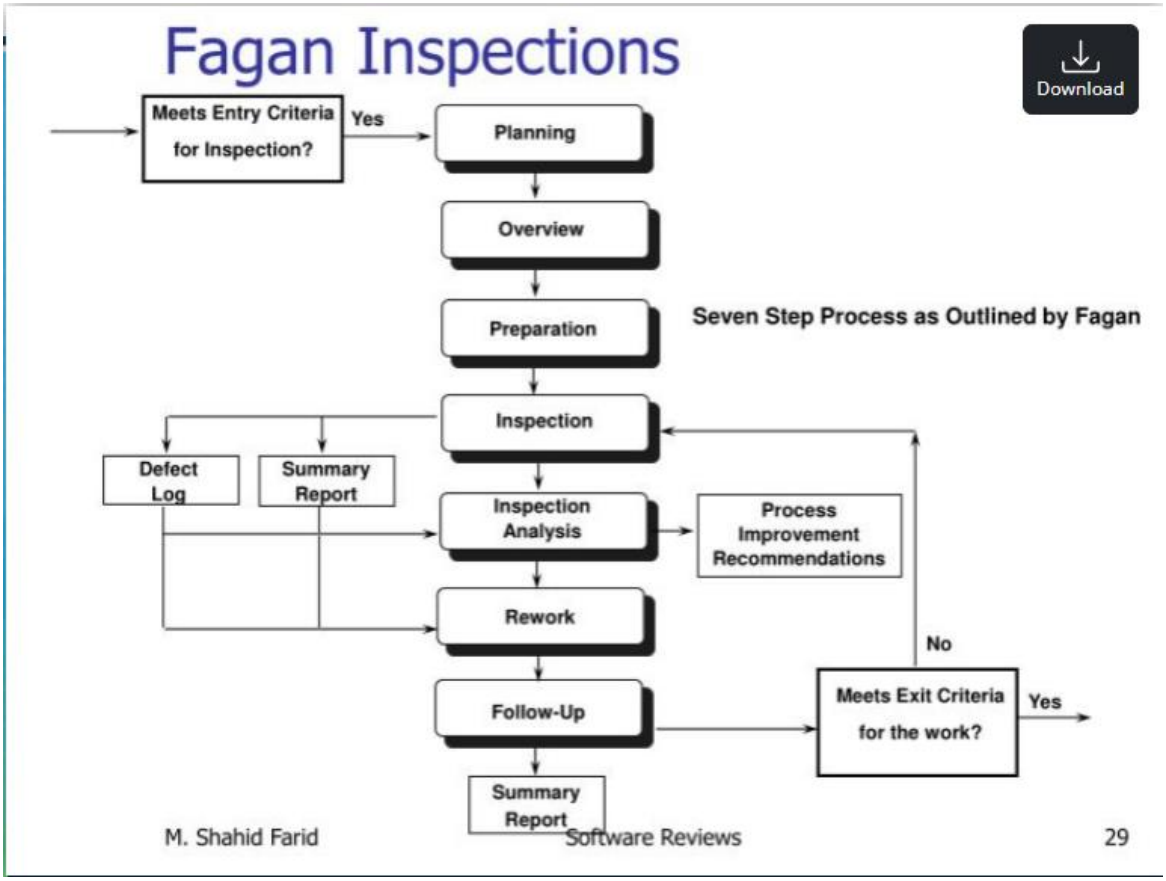
3. Preparation: The reader provides a neutral summary of the deliverable by paraphrasing it as an impartial expert. The group makes extensive preparations to Determine any possible flaws.
The group gets ready for the real examination here. This include putting together the inspection team, setting up the meeting, and making sure everyone is conversant with the procedure being examined.

4. Inspection: At the center of the procedure is a methodical review of the output. Specialists examine it from several perspectives, looking for contradictions, ambiguities, and departures from the norm.
the center of the operation! The group looks through the procedure or end item methodically. They spot possible problems, anomalies, or nonconformities. This stage involves conversations and observations.

5. Process Improvement: Efforts to improve the process are informed by the insights discovered during inspection. Teams improve their procedures to stop recurrence by learning from defects that are found.

6. Rework: The author updates the deliverable if errors are found. This step's iterative design guarantees ongoing improvement.
This process entails fixing any flaws or inconsistencies that are discovered. To fulfill quality requirements, it can include changing the product or revising some steps in the process.

7. Follow-Up Activity: The group examines the updated deliverable to make sure that all problems have been fixed. Insights from the past guide upcoming examinations.
Until corrective measures are implemented, the inspection remains incomplete. This entails monitoring development, making sure that corrective measures were carried out, and guaranteeing ongoing improvement.



TYPICAL OPERATIONS

1. Planning: The team gets ready for the inspection at this first stage. They specify the prerequisites for entrance, making sure the paperwork satisfies them before moving forward. Exit criteria are specified in the high-level document and act as the benchmark for assessing the results of operations that follow.

2. Material Preparation: Participants go over the supporting documentation and the item (such as the code or specs) that will be inspected. Any queries or possible flaws are noted.
3. Selecting Attendees: The team brings together all pertinent parties, such as moderators, authors, and reviewers.

4. Calling the Conference Place: The inspection meeting will take place at an appropriate location.

5. Overview: A summary of the contents being reviewed is given to the participants. This stage guarantees that everyone is aware of the inspection's context and goal.

6. Group Education: To enlighten one another about the subject matter, participants hold a group discussion. They dispel misconceptions, exchange knowledge, and come to a consensus.

7. Role Assignment: Particular roles are assigned:
- Author:
The individual in charge of the examined document.
The person who wrote the low-level document

- Moderator:
Aids in the process of scrutiny.
responsible for the inspection session, functions as a coach

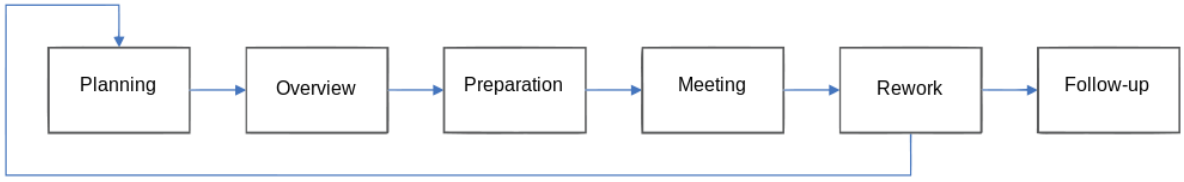
- Reviewer(s):
Check for errors in the document.
reviews the low-level document from a testing standpoint

8. Preparation: Each participant gets ready on their own by reading the content and figuring out any possible problems.

9. Inspection Meeting: To discuss the document, the team gets together. They actively look for errors by comparing the output to the high-level document's specified exit criteria.

10. Real Finding of Defects: Throughout the meeting, flaws are found, classified as major or small, and recorded.

11. Rework: The writer, designer, or programmer fixes the flaws that were discovered throughout the examination. Fast corrections are made.
12. Follow-Up: Any problems that still exist are fixed after the inspection, and any required modifications are done.



WHY FAGAN INSPECTION MATTER?

Fagan inspections are a useful tool for identifying flaws in papers used in the development of systems. It has been discovered that Group Support Systems (GSSs) can greatly improve error identification in the context of Fagan inspections.

1. Requirements for Entry and Exit: Fagan inspections follow entry and exit criteria that have been pre-established. The conditions needed to enter a certain process are outlined in the entry criteria. For example, both high-level and low-level documents need to meet certain entrance requirements prior to official inspections. Conversely, exit criteria outline the requirements needed to finish a process. For instance, in order to move on to phase 2, low-level papers must meet the departure criteria outlined in the high-level document

2. Early Defect Detection: It is critical to find flaws as soon as possible once they are inserted. When compared to the maintenance phase, early operations incur far lower repair costs for problems. Fagan inspections aid in the early detection of problems by comparing each operation's output to the exit criteria.

3. Operations Involved: Several operations are often involved in a Fagan inspection.
-Planning: Organizing the procedure for the inspection.
-Preparation of Materials: Compiling the files for examination.

- Assembling Participants: Putting together the inspection group.
- Overview: Informing attendees about the resources being examined.
-Assignment of Roles: Assigning responsibilities to members of the inspection team.

4. Outcomes and Benefits:
- Quality Assurance: Fagan inspections guarantee accurate, understandable, and compliant paperwork. Test cases, code, specs, and test plans are all maintained with rigor using this quality control process.
- Early Error Detection: Fagan inspections stop problems from spreading later on by identifying flaws at an early stage.
- Enhanced Team Dynamics: Collaborative assessments promote enhanced dialogue and comprehension among members of the team.
- Cultivating a 'Quality' Culture: Regular inspections encourage an excellent culture.

ADVANTAGE AND DISADVANTAGE IN FAGAN INSPECTION

Advantages:

1. Defect Detection: Early in the development phase, flaws can be found with the aid of Fagan inspections. Teams can lower the cost of resolving defects by finding problems early on and addressing them before they spread to later stages by analyzing documentation at each level.
2. Formal Process: A methodical and structured approach is used during Fagan inspections. This protocol guarantees uniformity and comprehensiveness in identifying flaws.

3. Decreased Maintenance Costs: It is far less expensive to find and correct flaws in the development stage than it is to fix them during maintenance. Overall cost reductions are facilitated by Fagan inspections.

4. Quality Improvement: Fagan inspections improve the total quality of software artifacts, including requirements, design, and code, by identifying flaws at an early stage.

5. Validation: For every action, the procedure verifies that the result meets the predetermined exit conditions. The validation process guarantees compliance with quality requirements.

Disadvantages

1. Resource-Intensive: Participant time and effort are needed for Fagan inspections. It can take a lot of resources to prepare documents, schedule meetings, and carry out reviews.

2. Human mistakes: Human mistake can still happen even in the formal process. Reviewers could overlook flaws or interpret criteria incorrectly.

3. Logistical Challenges: Organizing team members for inspections can be difficult, particularly if they are dispersed geographically or working on several projects at once.

4. Not Widely Used: Although Fagan inspections were common in the past, current research indicates that they are no longer often used.

GILB AND GRAHAM INSPECTION

INTRODUCTION

- Gilb Inspections, named for the visionary Tom Gilb, are designed to instill quality from the very beginning in software products. Inspections find flaws early and stop them from getting worse, unlike retroactive quality adjustments. Research indicates that when defects are discovered later in the development cycle, the cost of fixing them rises.
As software development is a complicated process, producing results of the highest caliber is crucial. Gilb Inspections provide an organized method for attaining software product quality. They were created by Tom Gilb and improved over several years.

Now let's examine the main facets of Gilb Inspections:

1. Effectiveness and Objectivity:
- Software inspections' main goal is to embed quality directly into the product while it is being developed. Early defect detection is more economical than late defect correction.
- Research indicates that the longer a flaw goes undiscovered, the more expensive it is to fix. Proactive quality practices are therefore essential.

2. Informal and Formal Approaches- Depending on the requirements of the organization, software inspections can be either informal or formal.
- Informal inspections entail having someone other than the author go through documentation or code. These conversations take place in a meeting room or at the author's desk.
- Formal methodologies offer an organized procedure for inspections, including the well-known Fagan inspection and the Gilb methodology.

3. The Fagan Inspection technique: - The Fagan inspection technique, which was created by

Michael Fagan at IBM in the middle of the 1970s, places a strong emphasis on rigor.
- It consists of pre-inspection tasks, an inspection conference, and follow-up after the inspection.
- Authors, inspectors, testers, and moderators are typical roles.
The Gilb methodology was created by Tom Gilb and serves as a supplement to the Fagan approach.
- Organizations base their formality level decisions on the specifics of their company.
- Project schedules specifically include formal inspections for essential systems (such as telecom software).

5. Business Impact and Formality: - Defects can have a serious negative effect on businesses. One-line code changes have caused telecommunications failures, which emphasizes the importance of thorough inspections.
- Businesses customize their inspection procedures to meet their unique requirements, taking into account the influence on customers and company culture.

INSPECTION PURPOSE

In order to address the shortcomings and inefficiencies of traditional formal technical reviews and encourage their implementation in software companies, Gilb and Graham (1993) presented a thorough inspection technique.

1. Software Inspections' Objective:
- Embedding quality into the software product from the beginning is the main objective of software inspections, as opposed to trying to retrofit quality later in the development cycle. It is more economical to identify and fix flaws early in the process rather than waiting till later.
- Research indicates that the cost of fixing a fault increases the longer it goes undiscovered. As a result, proactive quality control via inspections is highlighted.

2. Differences in Inspection Methodologies:
- Informal Approach: This entails having someone other than the author undertakes a quick walkthrough of the document or code. Usually held in a meeting room or at the author's desk, the meeting is informal.
The author and reviewer have an informal discussion on the code or text.
Standard Procedures: There are two established formal inspection methodologies:
Methodology for Fagan Inspection: Pre-inspection activities, an inspection meeting, and post-inspection follow-up are all part of this strategy, which Michael Fagan developed at IBM in the mid-1970s. Several roles are involved in the process: moderator, tester, inspector, and author.
- The Gilb Method: This methodology, created by Tom Gilb, places a strong emphasis on meticulous examinations of high-level designs, detailed code, and requirement documentation.

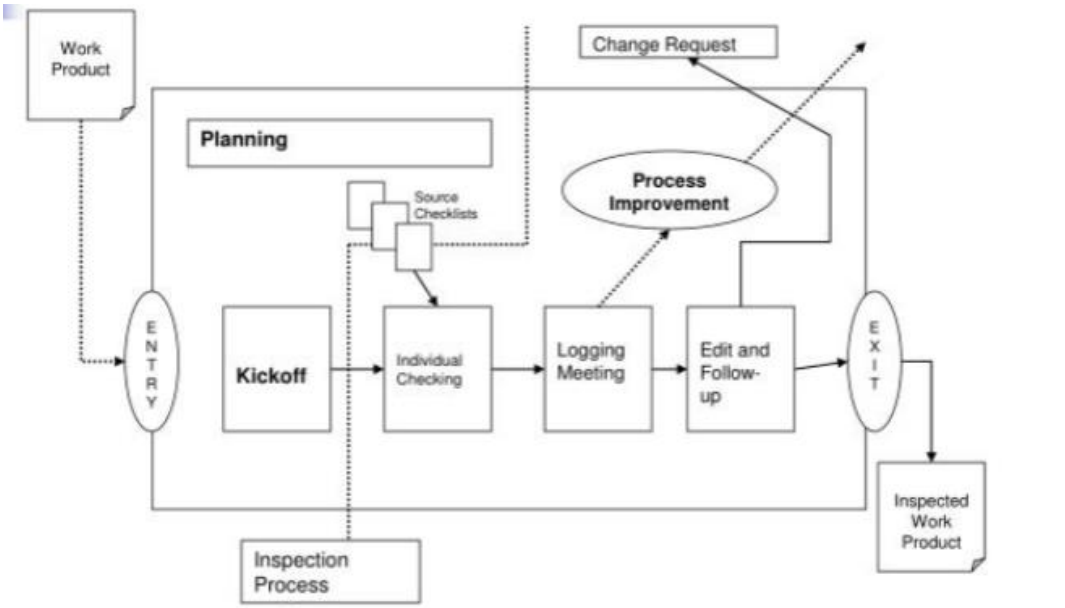
This strategy is adopted by organizations based on their business needs and the possible impact of flaws on their consumers.

3. Formality and Impact: - The organizational culture, business model, and possible repercussions of software flaws all influence how formal inspection procedures should be.
- For example, a telecommunications corporation is likely to adopt a formal inspection process because even a minor software flaw might result in significant interruptions. Examining requirement documents, design documents, and code that is specifically scheduled in the project schedule are all part of this process.
- Organizations customize their inspection procedures, whether official (as in the case of Fagan or Gilb) or informal, to meet their unique requirements.

4. Participants in the Inspection and Their Roles: A formal inspection process involves several key roles. These positions include:
- Author: The creator of the code or document under examination.
- Inspector: The person who checks the code or document for errors, consistency issues, and standard compliance.
- Tester: In charge of making sure the design and code adhere to the requirements and that the requirements can be tested.
- Moderator: Oversees the inspection meeting, making sure it proceeds as planned, adheres to the agenda, and fosters a positive environment. Every role has a part in making the inspection process more effective overall.

INSPECTION PROCESS

Tom Gilb and Dorothy Graham have described an organized method for inspecting software. Now let's examine each of these procedures:



1. Entry: Finding the document or artifact that has to be inspected is the first step in this process. It prepares the groundwork for the next actions.
The primary purpose of this stage is to ensure that inspection time is not wasted on artifacts that contain defects which the author should rightly have found

2. Planning: In this phase, the inspection team establishes the goals, parameters, and scope of the examination. They choose the parts of the document to concentrate on and set procedures for the procedure. The moderator determines the practical aspects of the inspection

3. Kickoff Meeting: To formally start the inspection, the team gets together. They talk about each participant's role, expectations, and goal. This is a chance to get everyone in sync and establish the atmosphere for the examination.
The author of the artifact may be required to give a quick walkthrough of the artifact to be inspected and its relation to the other documentation

4. Individual Checking: Every team member examines the document on their own. They carefully examine it to make sure it adheres to requirements and is free of flaws. This process guarantees comprehensive reporting from a range of angles. The reviewer should record any issues found.

The majority of defects found in inspection processes are found in the individual checking stage.

5. Logging Meeting: The group gets together again to discuss their results. They record observations, problems, and flaws. Differences are discussed cooperatively and settled. Transparency and knowledge exchange are encouraged by this action.
A planned and moderated meeting with the primary purpose of logging the issues found by the reviewers.

6. Edit: The document is revised in light of the issues found. Improvements are put into practice, and mistakes are fixed. Improving the content's accuracy and quality is the aim. The editor is responsible for addressing all logged issues in the inspected artifact. The editor decides if something is a defect or not

7. Follow Up: Following editing, the group looks over the updated file. They confirm that the flaws found have been fixed. All outstanding problems have been fixed, and if more iterations are required, they might happen. Moderator must also ensure that any defects found in s source document during inspection are forwarded to the owner of that document for correction

8. Exit: A last review brings the inspection to an end. The document is evaluated according to the specified standards and is either accepted or rejected. Documenting lessons learnt helps to enhance processes. An inspection will be exit when predefined set of inspection exit criteria have been satisfied.

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

1. Defect Detection: Early in the development phase, flaws can be found with the aid of Fagan inspections. Teams can lower the cost of resolving defects by finding problems early on and addressing them before they spread to later stages by analyzing documentation at each level.

2. Formal Process: A methodical and structured approach is used during Fagan inspections. This protocol guarantees uniformity and comprehensiveness in identifying flaws.

3. Decreased Maintenance Costs: It is far less expensive to find and correct flaws in the development stage than it is to fix them during maintenance. Overall cost reductions are facilitated by Fagan inspections.

4. Quality Improvement: Fagan inspections improve the total quality of software artifacts, including requirements, design, and code, by identifying flaws at an early stage.

5. Validation: For every action, the procedure verifies that the result meets the predetermined exit conditions. The validation process guarantees compliance with quality requirements.

6. Cost-Effectiveness: It is less expensive to fix flaws at the inspection stage than it is to do so later in the development cycle. Gilb inspections assist in lowering the price of fixing flaws.

DISADVANTAGES

1. Resource-Intensive: Participants must devote time and energy to the Fagan inspection process. It can require a lot of resources to plan meetings, prepare materials, and carry out reviews.

2. Human Error: In spite of the formal procedure, mistakes can still happen. Reviewers could overlook flaws or misunderstand the specifications.
Coordinating team members for inspections can present logistical challenges, particularly if they are geographically dispersed or working on several projects at once.

4. Not Widely Used: Although Fagan inspections were common in the past, current research indicates that their use is no longer widespread.

5. Rigidity: Some teams may find the organized format of Gilb inspections to be inflexible. It might not be appropriate for all projects, particularly those where the implications of software failure are less severe.

6. Entry and Exit Criteria: It can be difficult to follow the entry and exit criteria. These standards specify the start and finish dates of inspections. Deviating from these recommendations could reduce productivity.