PEER REVIEW IN SW

TRAINING FOR NEW ENGINEERS

24G TRAINING (REV 1.0)
ESW DEPARTMENT
RENESAS DESIGN VIETNAM
RENESAS ELECTRONICS CORPORATION

OUTLINE

INTRODUCTION

OBJECTIVE OF THIS COURSE

- Understand the role of REVIEW in software development
- Understand the importance of REVIEW
- Understand the characteristic of Fagan inspection
- Understand the evaluation process of PEER REVIEW

Peer review

NOT Beer review

EXPECTATION

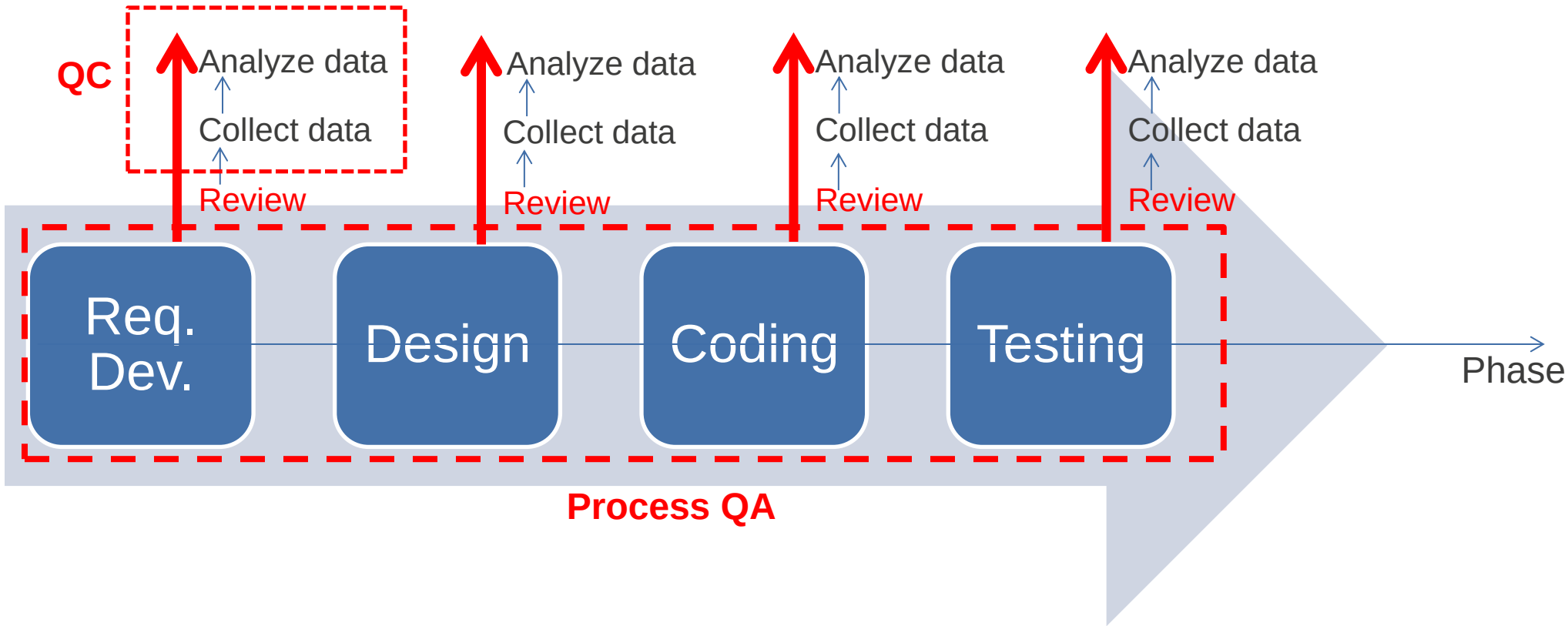
- Trainees can apply what are learnt into their actual projects
- Trainees can do proper peer review following RVC software development process
- Trainees can analyze the review performance

COURSE CONTENT

- Lecture: 3.5 hours
- Practice (team-work)
- Q & A after practice: 3.5 hours

RVC SW DEVELOPMENT PROCESS

RVC SW DEV. PROC. ACTIVITIES



- QA: ensure good development processes to ensure the quality
- QC: control the quality based on statistic methods

WHAT IS A REVIEW?

WHAT IS A REVIEW?

- In each development process (step), the person who is not developer investigates the deliverables such as specifications and source code in detail with developer, and makes feedback to developer whether the content satisfies specifications or requests, and whether there is any mistake and bug.
  - Execute document review for specifications and design documents
  - Execute code review for source code

TYPES OF REVIEW

- **Management review:** Review with high-level managers the important matters of project, like: product release, continuance/stop of project, resource change. Review with the purpose of solving issue.
- **Design review:** To see if tasks planned in the corresponding process could be executed as plan. Review with the stakeholder of project with the purpose of agreement/approval of moving to the next process.
- **Peer review:** Review between designers with the purpose of removing defect of deliverable in technical viewpoint.
- **Status review:** Project leader and members confirms entire project, like: confirming the delay or advance of progress of creating deliverables against predefined milestones, confirming the risk.

WHY IS REVIEW NECESSARY?

- **What is the importance in a development? What is a successful development?**
  - Target KPIs are set at the beginning and achieved at the end
  - KPIs: Quality – Cost – Delivery (QCD)
- **If there are many bugs found in later phases (testing) of a project:**
  - Needs to return to earlier phases (design, coding…) to find the originality of the bugs
  - Rework may be needed: delay, extra cost…
    - Quality may not be guaranteed (Q)
    - Cost may be much higher (C)
    - Delivery may get delay (D)

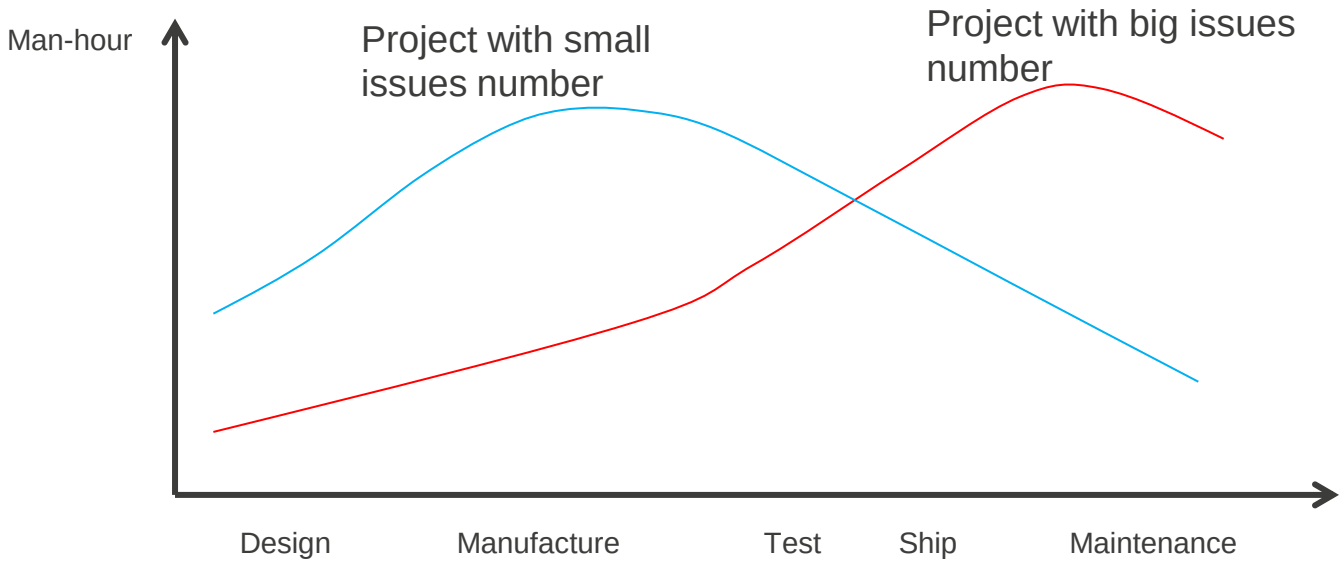
Lose customer satisfaction!

Detecting/Removing defects at early phases is important to achieve QCD.

➔ **Execute review (peer review), improve quality.**

REVIEW EFFECTIVENESS

- **Does review really take lots of time? Is it really effective?**

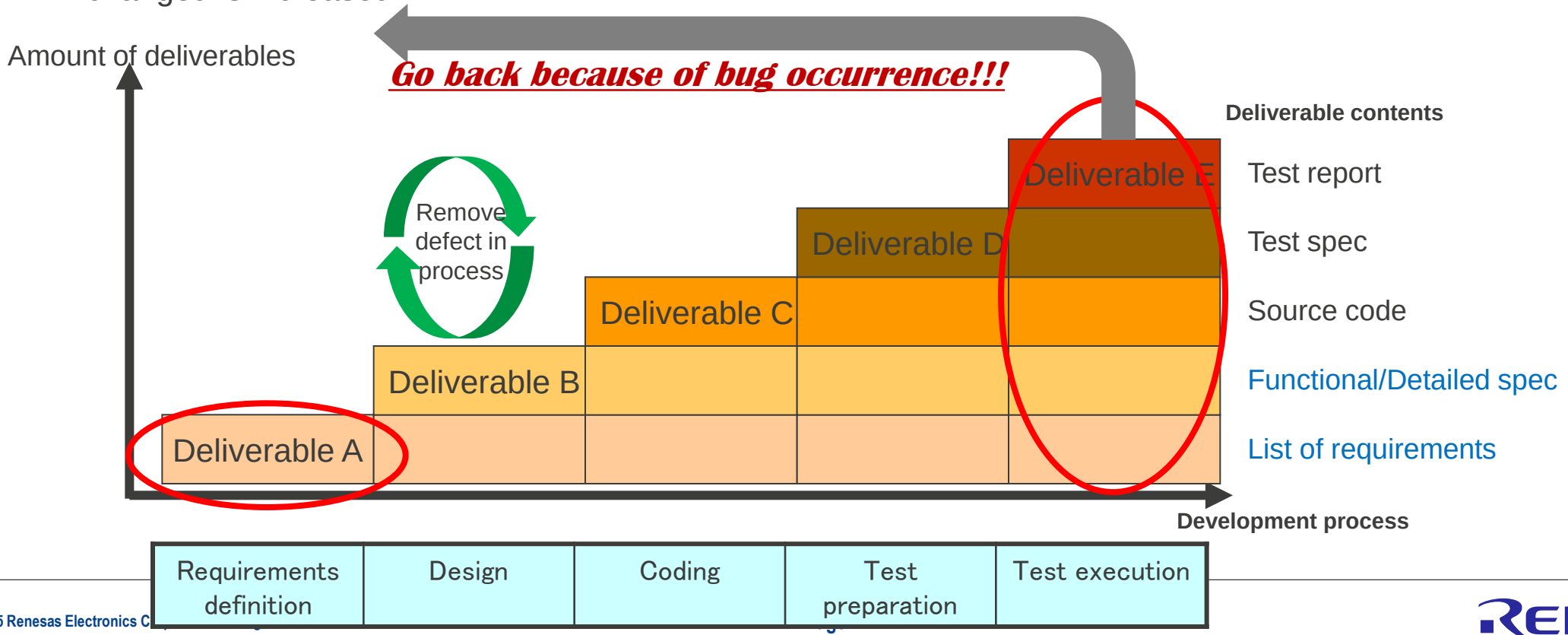


(1) Cost of externally detected bug: for bug found outside company
(2) Cost of internally detected bug: for bug found inside company
(3) Cost for quality evaluation: for review, test, quality analysis
(4) Cost for defect preparation: for training, improving engineering skills, quality process

(1) Cost of externally detected bug + (2) cost of internally detected bug > (3) Cost for quality evaluation + (4) cost for defect preparation

DEVELOPMENT PROCESS & DELIVERABLE VOLUME

- The more the process is going to later phases, the more the deliverable volume is increased.
- If defect is detected later, rework will more costly because the deliverable volume that needs to be changed is increased.



TO ACHIEVE PROJECT GOAL (QCD)

- Detect/correct bug at early phases, do not let it flow into later phases
  - First, create your output with no bug
  - Next, detect and correct bug as soon as it is detected

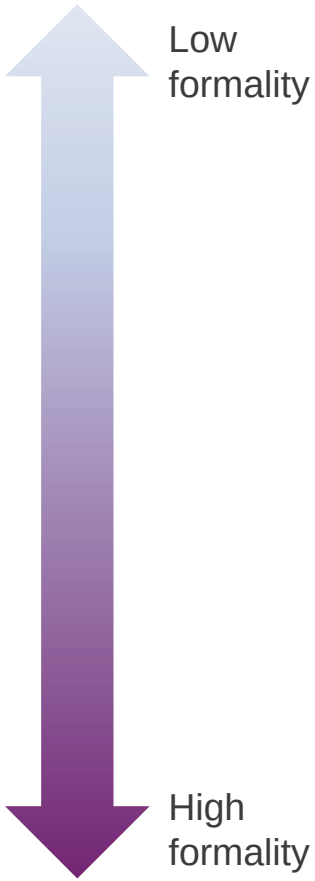
➔ **Conduct review during design phase**

Peer review is executed with the purpose of detecting bugs.

PEER REVIEW

PEER REVIEW METHODS & CHARACTERISTICS

- **Ad-hoc review:** Instant review that members confirm and give comments if necessary.
- **Pass-around:** Send necessary material by email, reviewee receives comment from reviewer separately. Review is in a form of written style.
- **Walk-through:** Review in meeting style where reviewer invites reviewee (person who receives review). Review material is distributed at the meeting, reviewee explains it and receives comment from reviewer.
- **Team review:** Review in meeting style periodically with time and member planned beforehand. Like deliver material beforehand, rule exists to some extent.
- **Inspection:** This is the most systematical review in “written style + meeting style”, where each participant is given a role; material is distributed beforehand; each reviewer executes on desk review and collect comments. After that, a member other than the creator of deliverable leads the meeting.



COMPARISON 1: PEER REVIEW PROCESS & ROLE

Item		Walk-through	Team review	Inspection
Before review				
	Understand contents beforehand	×	○	○ (must)
	Collect comments beforehand	×	△ (optional)	○
During review				
	Moderator	×	×	○
	Contents reader	Creator	Creator	Not creator
	Minutes recorder	Creator	Creator	Not creator
	Discuss modification method	Freely	Freely	×
After review				
	Process evaluation	×	×	○
	Cause analysis of comments	×	×	○

COMPARISON 2: CHARACTERISTICS OF WALK-THROUGH & INSPECTION

Comparison item	Walk-through	Inspection
Efficiency of bug detection in design process	○	◎
Improve the quality of deliverable, simplicity of modification	○	◎
Consider and discuss the deliverable content	◎	○
Training, technical transfer	◎	○
Defect recurrence prevention based on cause analysis of comments	△	◎

EXAMPLE IN OTHER COMPANIES

- **AT & T Bell Lab:** Detection of errors by inspection reduces cost to 1/10 and quality is 10 times improved.
- **IBM:** Defects could be found by Fagan inspection (*). If defects leaked to other company, man-hour cost would be 100 times higher.

Inspection is an effective peer method for bugs removal!

(*) Fagan inspection: This is the inspection method Michael Fagan applied at IBM in 1970 to improve programming quality & productivity.

GOOD “CULTURE” WHEN DOING PEER REVIEW

- **Welcoming bug detection:** When there are lots of bug detected and reported to managers, then:
  - Creator, reviewee think their bosses will get angry because of many bugs
  - Creator, reviewee seem to hide bugs
  - Make all minor comments for bugs

➔ **Wrong thinking!**
➔ **Bugs may still remain.**

GOOD “CULTURE” WHEN DOING PEER REVIEW

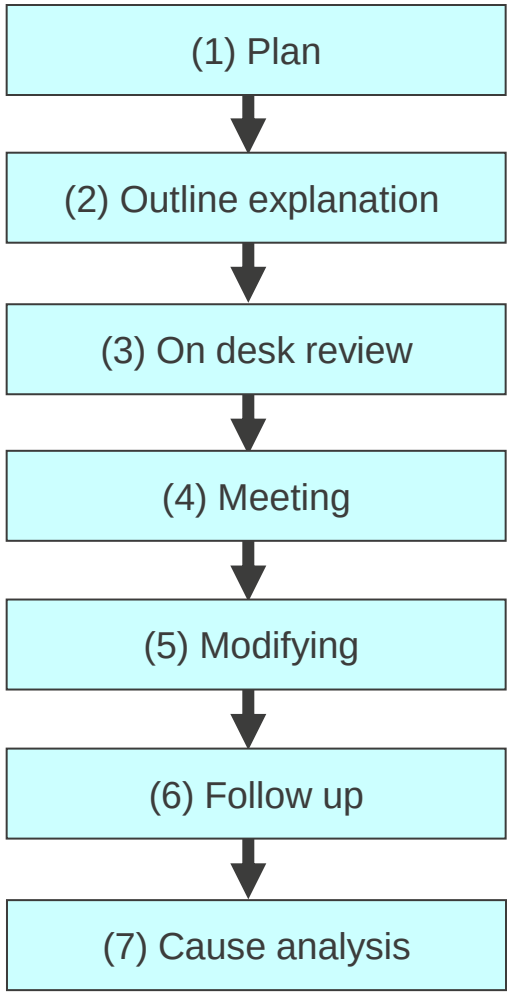
- **Welcoming bug detection:** When there are lots of bug detected and reported to managers, then:
  - Creator, reviewee think they bosses will get angry because of many bugs
  - Creator, reviewee seem to hide bugs
  - Make all minor comments for bugs

➔ Bug occurs because human made mistake. If bug can be detected, we will be happy because we can stop letting it flow into next processes. Keeping this recognition in mind is important!

FAGAN INSPECTION

FAGAN INSPECTION FLOW

- Steps are defined clearly
- Regulate what to execute at each step

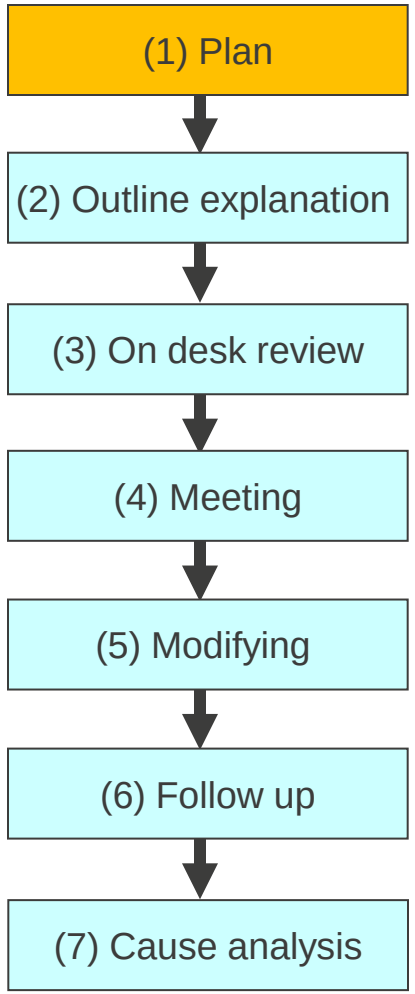


ROLES OF PARTICIPANTS IN FAGAN INSPECTION

- **Moderator:** Control the review so that it is executed in good direction
- **Inspector (reviewer):** Review deliverable contents
- **Creator (reviewee):** Create deliverables which are the review target objects
- **Reader:** Explain deliverables' contents during the review
- **Recorder:** Records discussion during the review

FAGAN INSPECTION FLOW - PLAN

- **What is “plan”:**
  - Prepare for starting the inspection. It is not the entire plan of the project, but just a plan executing each inspection.
- **Main activities:**
  - (Creator) Inform to Project Leader if deliverables (review objects) are ready. And, confirm if:
    - are starting conditions satisfied? Or, is it possible to satisfy all starting conditions and prepare enough documents until the next step (“Outline explanation”)?
  - (Project Leader) Assign moderator and inspector
    - Also, decide other roles (contents reader, minutes recorder).
    - Inspector needs to have appropriate skills.
  - (Creator): Confirm if the starting conditions are satisfied? And prepare the set of materials to be reviewed.
  - (Moderator): Create an inspection execution plan.

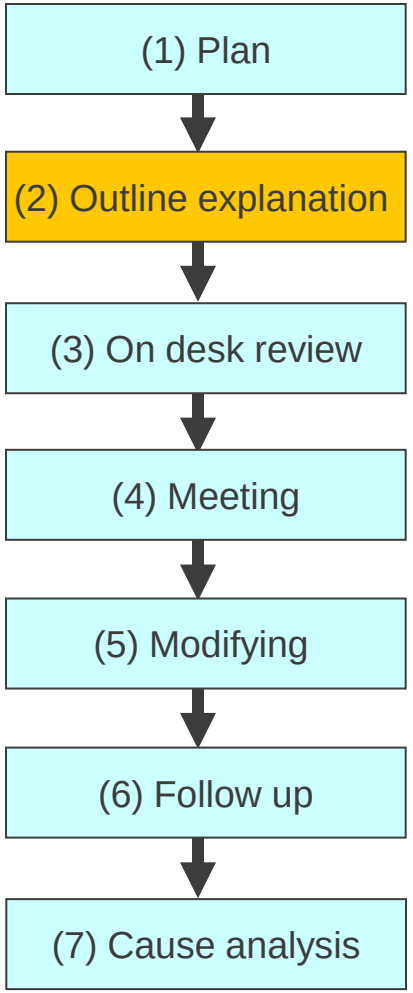


FAGAN INSPECTION FLOW - PLAN

- **Example of starting conditions:**
  - Source code is compiled successfully without error.
  - Statistic analysis is already executed without issue found.
  - All typo errors are checked and corrected.
  - Purpose & venue of the review are clarified
- **General inputs of a review:**
  - Deliverables for which the inspection will be done
  - Top level document of deliverable
  - Result of static analysis
  - Checklists
- **Inspector:**
  - His/her technical ability should be evaluated with reliable evidence (e.g., working experience…)
  - He/she must do careful on-desk review before the meeting

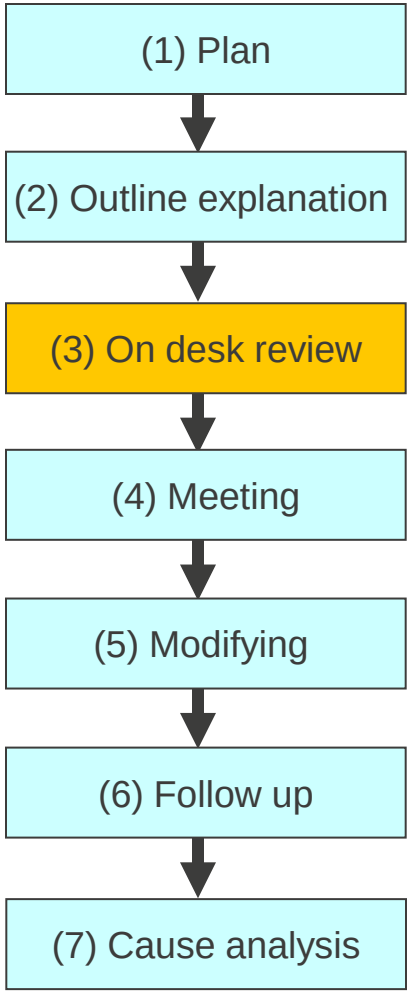
FAGAN INSPECTION FLOW – OUTLINE EXPLANATION

- **What is “outline explanation”:**
  - Provide to the inspector with necessary information for finding issues by on-desk review.
  - It can be omitted if all inspectors know well about the review materials.
- **Main activities:**
  - (Creator) Send the review materials and explain review targets, check viewpoints (also can be shared by inspector), expectations…
  - (Creator) Explain the materials and answer questions.
  - In case there are many comments received, the moderator can decide to stop the inspection.



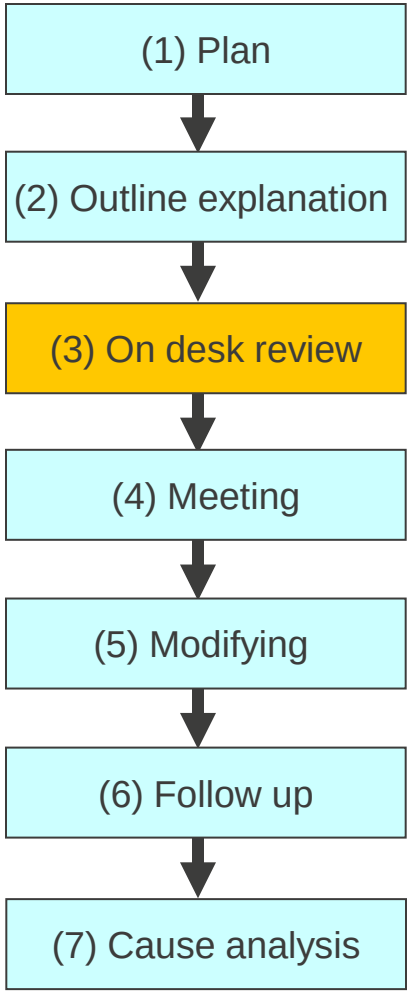
FAGAN INSPECTION FLOW – ON-DESK REVIEW

- **What is “on-desk review”:**
  - All inspectors investigate (“inspect”) the deliverables, find “comments”, “potential” issues, or issues
- **Inspection is effective in detecting bugs in 2 steps:**
  - Detecting bugs by on-desk review
    - Inspectors can find issues from different personal viewpoints and experiences. So, the bugs found ratio is high.
  - Detecting bugs in meeting
    - Inspectors can share ideas in meeting for discussion. So, new bugs can be found.
    - However, the effectiveness is not high if their preparations are not enough.



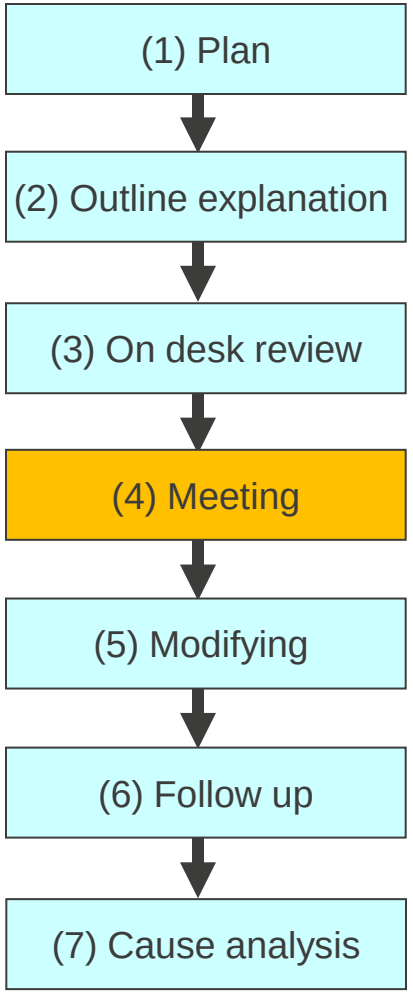
FAGAN INSPECTION FLOW – ON-DESK REVIEW

- **Main activities:**
  - (All inspectors) Check the deliverables
    - All deliverables must be checked → This is a prerequisite for implementing a meeting.
    - If there is no special predefined checking method → Use checklists.
    - In case of receiving some viewpoint from the moderator to check/confirm some important things in advance → focus on confirming those important things at first; then also check other scopes/viewpoints later.
  - (Reader) Many to read the deliverable contents during on-desk review
  - (All inspectors) In principles, do not ask questions to the author. Just summarize all comments, pointed out issues, unclear points, simple mistakes... (Simple mistakes will not be discussed in the meeting.)
  - If on-desk review is also conducted by the creator → The creator is also an inspector.
  - In case there are many comments received, the moderator can decide to stop the inspection.



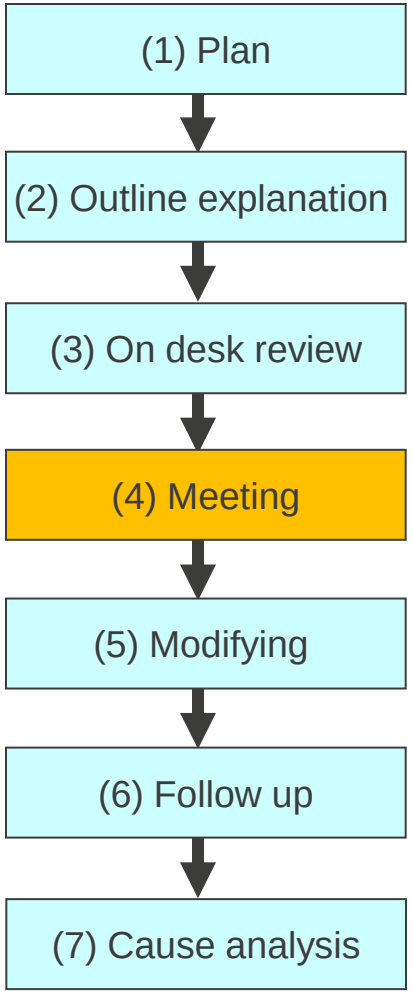
FAGAN INSPECTION FLOW – MEETING

- **What is “meeting”:**
  - Collect all comments found in the previous steps and discuss if there is any issue.
  - Find new comments by discussion: New bugs found at meeting is said to be 4-25% of the total.
- **Main activities (Opening):**
  - (Moderator) Confirm the preparation status of all instructors (on-desk review is already done?!). And, confirm if all starting conditions are satisfied.
  - (Moderator) Take appropriate countermeasure in case there is issue on insufficient preparation:
  - (Moderator) Confirm if there is any question and comment.



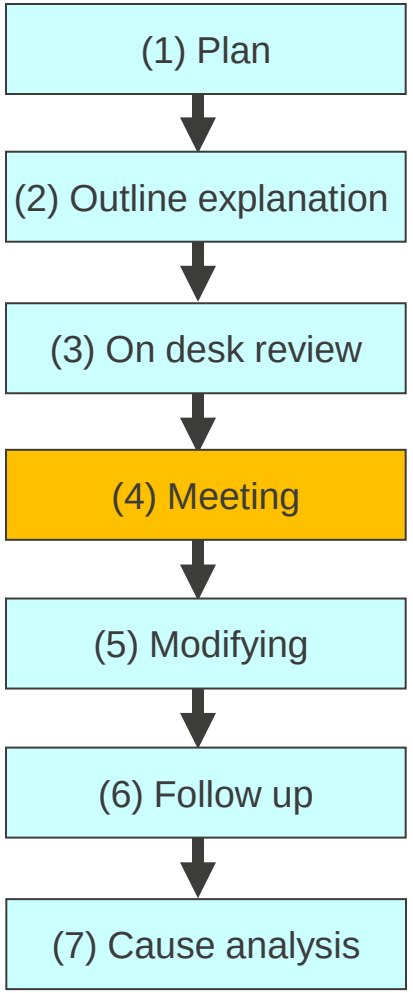
FAGAN INSPECTION FLOW – MEETING

- **Main activities (During meeting):**
  - (Reader) Explain the deliverables by his/her own words.
  - (Reader) Confirm if there is any comment or question.
  - (Inspector) Make comments and questions for the explained parts.
    - Must not blame on the creator, deliverables and other inspectors
    - Must not raise simple mistakes in the meeting
  - (Moderator) If a comment is confirmed as an issue, move to the next comment
    - Do not let the discussion falls into finding corrective action/method
  - (Recorder) Record the issue content into an “issue log”
    - Detail descriptions should be recorded
  - (Creator) Verify the deliverables with subjective eyes.



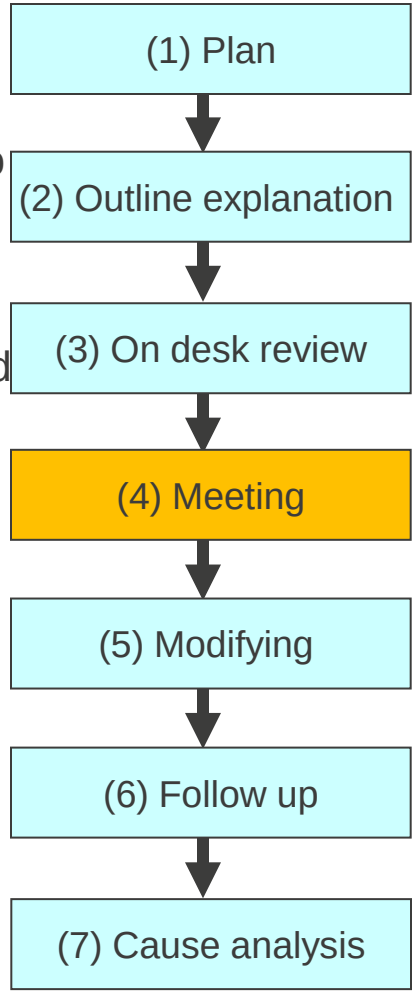
FAGAN INSPECTION FLOW – MEETING

- **Main activities (Closing):**
  - (All inspectors) Evaluate the deliverables:
    - Need to follow-up: Need to revise the deliverables and follow-up after that
    - Need to re-inspect:
      - In case big modifications for the deliverables are needed
      - Inspection must be redone from the beginning for the updated deliverables
    - No need to follow-up: Might need some little modification for the deliverables. However, it is possible to skip cause analysis and follow-up process
  - (All inspectors) Evaluate the review process
    - Evaluate the review performance based on, e.g., statistical methods
  - (Moderator) Create the meeting minutes and inspection report



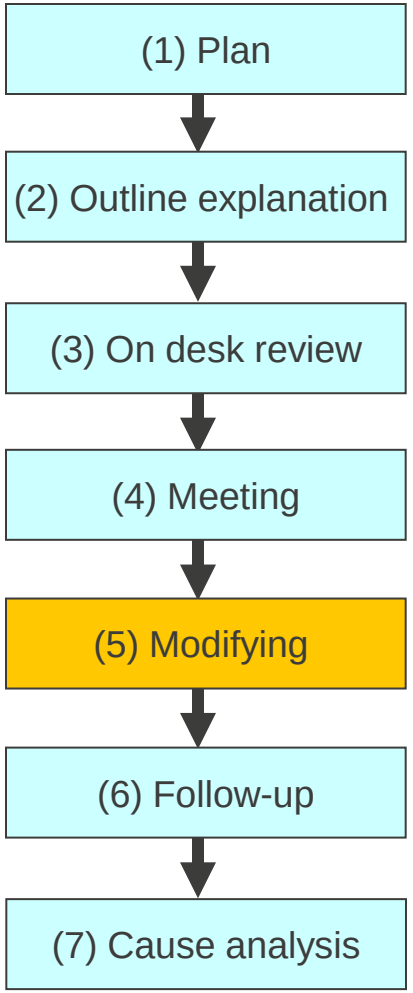
FAGAN INSPECTION FLOW – MEETING

- **Effect of using a reader:**
  - Readers can also find “potential” issues: Impossible to explain well the deliverables’ contents, to answer questions, or the answers are different with the ones of the creators: Possibly, the deliverables’ contents are too complicated, incomprehensible.
  - Note: If the reader does not read using his/her own words, the review efficiency may be reduced remarkably → The importance is the reader reads the deliverables’ contents with the understanding of him-/her-self and by using his/her own words.



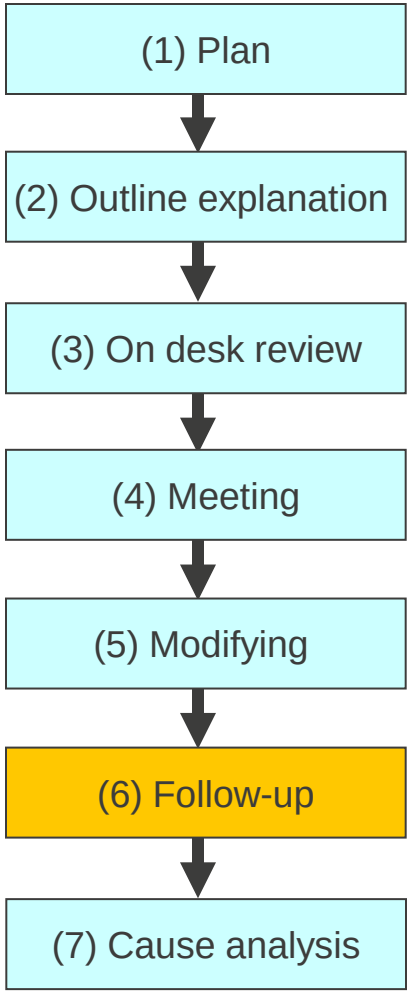
FAGAN INSPECTION FLOW – MODIFYING

- **What is “modifying”:**
  - Deliverables need to updated/modified based on the list of issues and simple mistakes
- **Main activities:**
  - (Creator) Consider each item of issue log and which item should be modified.
  - (Creator) If a new issue (except simple mistake) is found when during modifying, add it into the issue log.
  - (Creator) Modify items which should be modified based on simple mistake list and issue log.
  - (Confirmer) Follow-up the modification progress status
  - (Creator) Consider if the modify cause degradation (e.g., degraded bug)
  - (Creator) Update the issue log
- Pay attention not only to “modifying” but also to “recording”: It takes time to record the supplementary information, but it will be useful later.



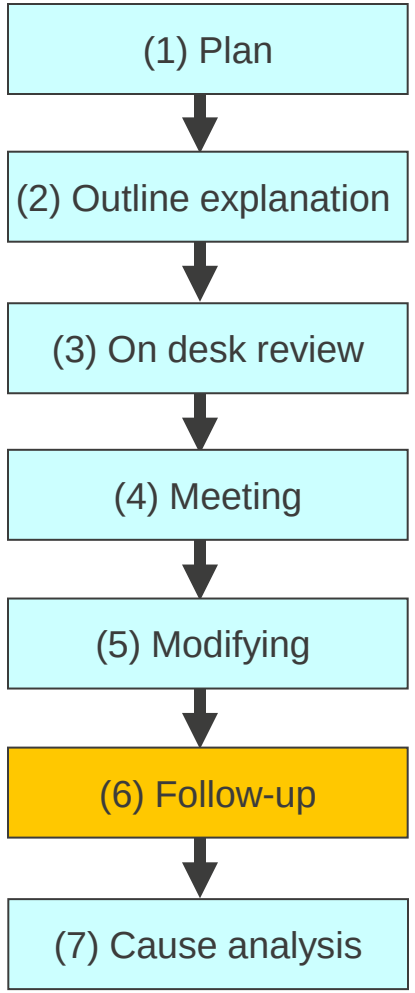
FAGAN INSPECTION FLOW – FOLLOW-UP

- **What is “follow-up”:**
  - Confirm the deliverables after modification
  - In some cases, it is possible to skip follow-up meeting
- **Main activities:**
  - (Confirmer & Creator) In the meeting, investigate the deliverables after modification; confirm if the modification is correct
  - (Confirmer) In case the re-investigation is judged to be needed, propose the re-inspection
  - (Confirmer) Approve if modification is confirmed to be complete
  - (Confirmer) Report the modification completion and modification man-hour to moderator



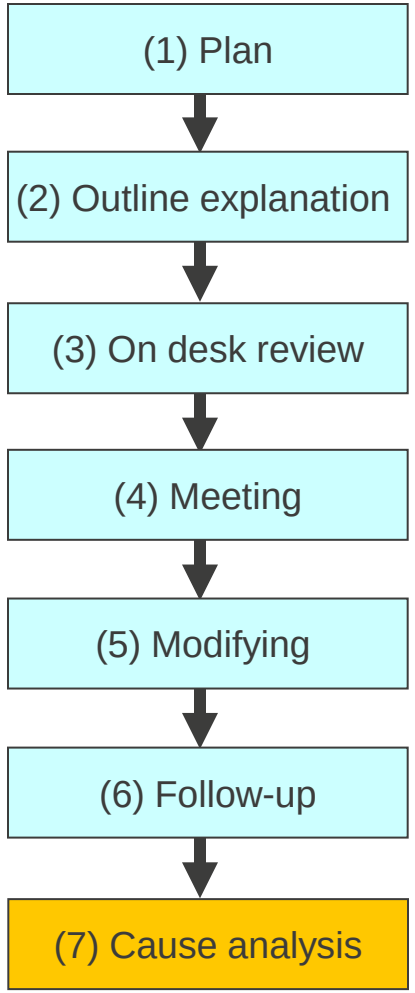
FAGAN INSPECTION FLOW – FOLLOW-UP

- **Who can be the confirmer:**
  - Usually, the approver, moderator can take this duty
  - Other inspectors or several inspectors can be, too
- Items that the confirmer should check:
  - Modified contents are correct?
  - Is there any degradation?
  - Comments which are not modified are OK or not?



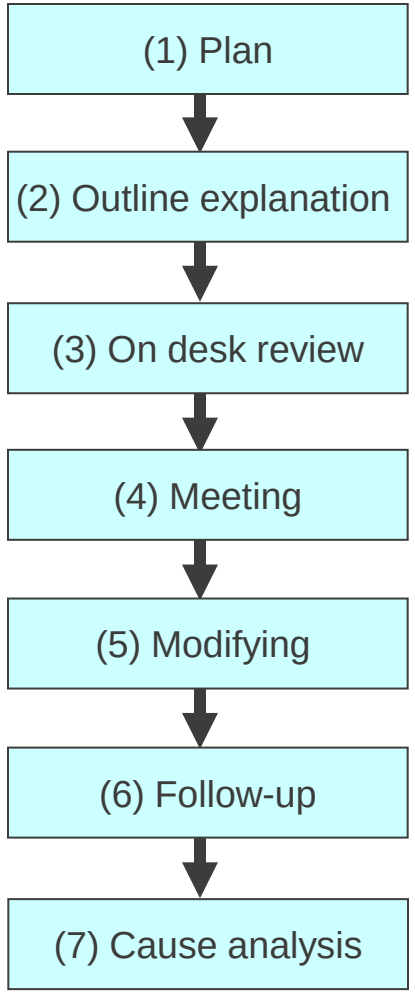
FAGAN INSPECTION FLOW – CAUSE ANALYSIS

- **What is “cause analysis”:**
  - Prevent bugs and improve the process by analyzing the comments, issues…
- **Main activities:**
  - (Moderator) Decide the important issues which should be solved (e.g., trend analysis)
  - (All inspectors) Analyze the motivational cause and propose countermeasures
  - (Moderator) Summarize them as proposal for process improvement
- Cause analysis is an advantage of inspection:
  - Possible to prevent bugs in future
  - As long as there is bug, there is room for improvement
  - Reveal the weaknesses in recognition/judgement, find preventive measures
  - Can become a core process of improvement activity



FAGAN INSPECTION - SUMMARY

- **Characteristics:**
  - Person who enough skills who is not the creator can be the moderator and leads the review.
  - Focus on the on-desk review process
  - Top priority is efficiency of bug detection
    - Do not discuss the modification method at meeting
    - Creator also joins in bug detection as inspector
  - Process improvement is included in that concept
- **Whether to use inspection method for all deliverables?**
  - No organization can inspect all deliverables.
  - Narrow down to the important target objects which should be inspected
    - Conduct inspection for the important functions of development system.
    - Conduct inspection for the processes which have high possibility of making bug



EVALUATION OF PEER REVIEW PROCESS

PERFORMANCE INDEXES

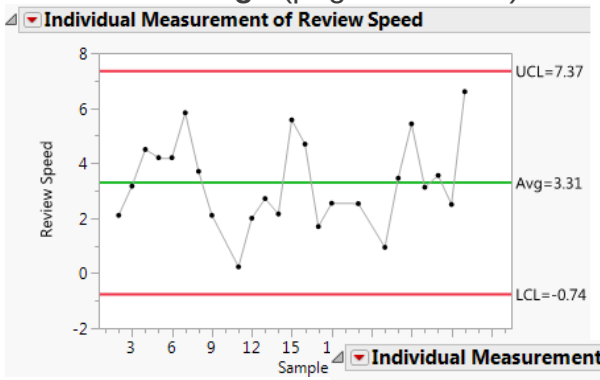
- **Peer review speed:**
  - It can be used to evaluate if the review time is compatible with the scale of the review objects
    - Whether the review is too fast?
    - Whether the review is too slow?
    - Whether the review rate is moderate, within standard range?
- **Peer review defect density**
  - It can be used to evaluate if the number of defects is compatible with the scale of the review objects.
- **Peer review efficiency**
  - It can be used to evaluate if the resource (cost) used for the review is compatible with the scale of the review objects.
  - It can be seen as the inverse of peer review speed

PEER REVIEW SPEED

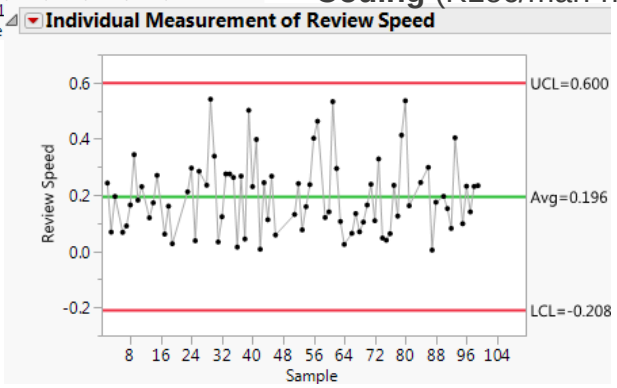
- **Definition:**
  - Peer review speed = Peer review scale/Peer review time
- **Peer review scale examples:**
  - Design document: Number of pages (A4-size)
  - Source code: Number of lines of code (LoC)
- **Tips:**
  - Evaluation/calculation can be done separately for on-desk review and meeting.

RVC SW Dept. data

Basic Design (page/man-hour)



Coding (KLoc/man-hour)



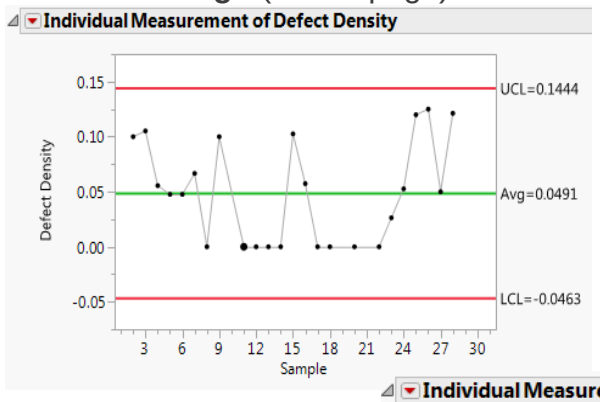
DEFECT DENSITY

- **Definition:**
  - Defect density = Number of defects/Peer review scale
- **Number of defects:**
  - Count all defects(*) in the deliverables, work-products
- **Tips:**
  - Evaluation/calculation can be done separately

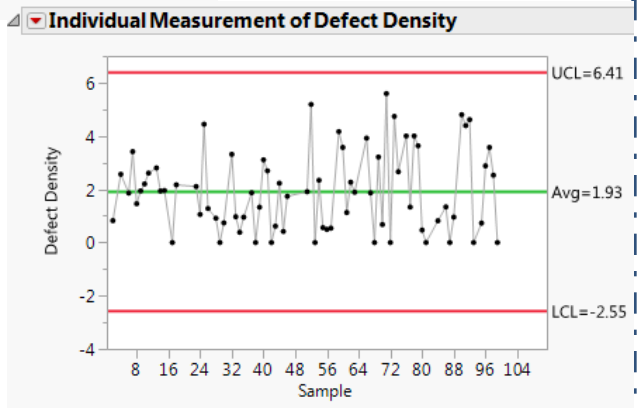
  for on-desk review and meeting.

RVC SW Dept. data

Basic Design (defect/page)



Coding (defect/KLoC)

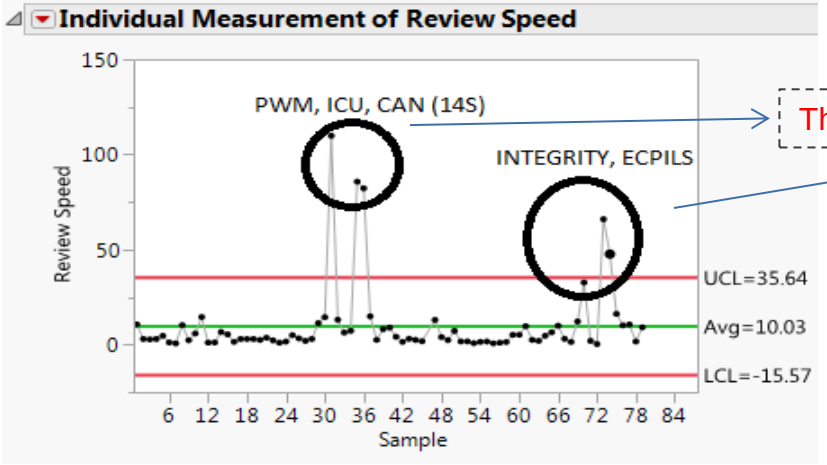


(*) What is a defect: the definition of defect is described in the standard development process. It can be:
- Bug in source code
- Wrong description in design document

etc.

CONTROL CHART FOR QUALITY CONTROL

- **Control chart:**
  - It is one of the 7 tools for Quality Control
  - By collecting the past data, we can determine (calculate) acceptance ranges.
- **Acceptance ranges:**
  - It is the range (Min-Average-Max) in which we expect that the result of a new project/process will fall. It is calculated subjectively based on statistical methods: It provides subjective data for the performance of the process.



The review in these projects might be not good!

SUMMARY

SUMMARY

- SW quality must be improved from early phases (design)
- Peer review is an effective method for bug detection
- Fagan inspection is good for peer review. But we need to apply it to important review target objects only, due to its high effort. Which part of deliverables can apply:
  - New technology, technique, tool
  - Complicated part
  - Part with lots of interface, core of product
  - Important functions
- We can use walkthrough method for other less important parts
- It is necessary and important to evaluate the results of peer review quantitatively.

REVISION HISTORY

Rev #	Creator	Changed contents
1.0 (Aug-2016)	Quynh Tran	Newly created

www.renesas.com

END.