



CSE 321 Software Engineering

Review and Defect Analysis

Md. Shamsul Haque

WHY REVIEW ?

To err is human.

- Technical work needs reviewing for the same reason that pencils need erasers
- we need technical reviews is that although people are good at catching some of their own errors, large classes of errors escape the originator more easily than they escape anyone else

WHAT ARE REVIEWS?

- A MEETING CONDUCTED BY TECHNICAL PEOPLE FOR TECHNICAL PEOPLE
- A TECHNICAL ASSESSMENT OF A WORK PRODUCT CREATED DURING THE SOFTWARE ENGINEERING PROCESS
- A SOFTWARE QUALITY ASSURANCE MECHANISM

WHAT REVIEWS ARE NOT

- A PROJECT SUMMARY OR PROGRESS ASSESSMENT
- A MEETING INTENDED SOLELY TO IMPART INFORMATION
- A MECHANISM FOR POLITICAL OR PERSONAL REPRISAL!

SO WHAT IT IS ?

- Reviews are applied at various points during software engineering and serve to uncover errors and defects that can then be removed.
- Software reviews “purify” soft-ware engineering work products, including requirements and design models, code, and testing data.

WHAT DO WE LOOK FOR

- Errors and defects
- We make this distinction because errors and defects have very different economic, business, psychological, and human impact
- However, the temporal distinction made between errors and defects in this book is *not* mainstream thinking

REVIEW TECHNIQUES : INFORMAL, FORMAL

- Informal reviews include:
 - a simple desk check of a software engineering work product with a colleague
 - a casual meeting (involving more than 2 people) for the purpose of reviewing a work product, or
 - the review-oriented aspects of pair programming
- *pair programming* encourages continuous review as a work product (design or code) is created.
 - The benefit is immediate discovery of errors and better work product quality as a consequence.

REVIEW TECHNIQUES : INFORMAL, FORMAL

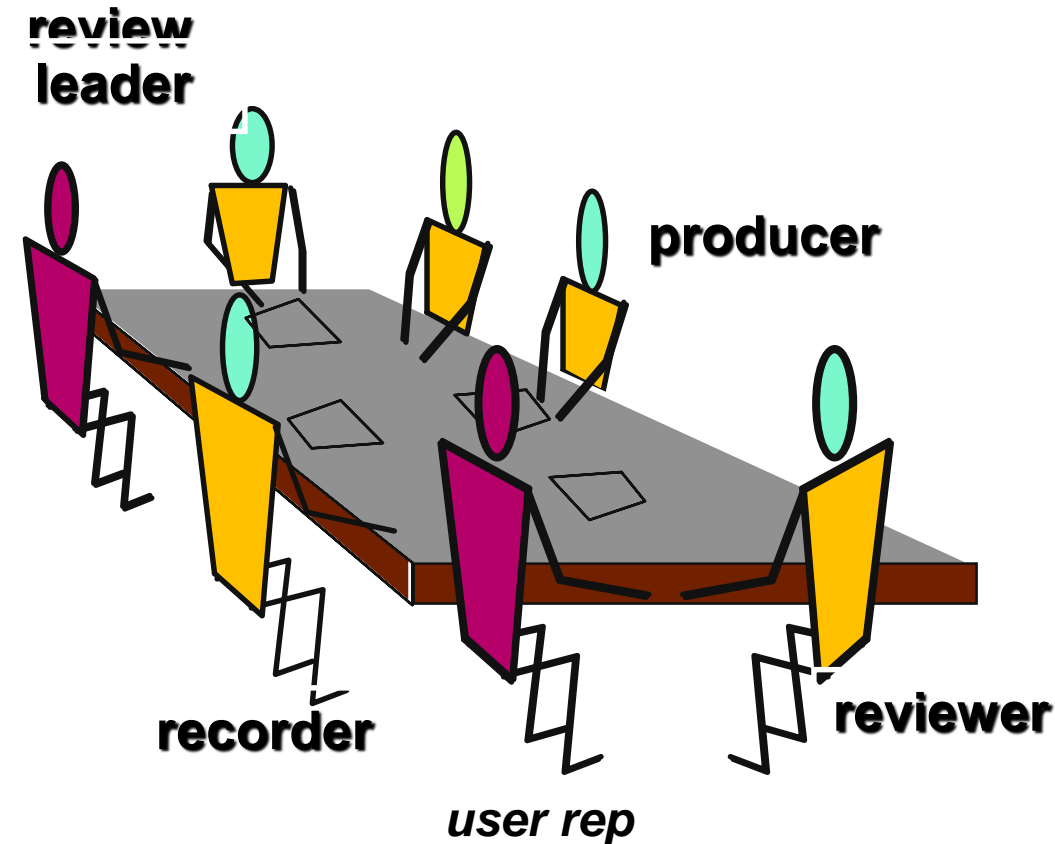
- The objectives of an FTR are:
 - to uncover errors in function, logic, or implementation for any representation of the software
 - to verify that the software under review meets its requirements
 - to ensure that the software has been represented according to predefined standards
 - to achieve software that is developed in a uniform manner
 - to make projects more manageable
- The FTR is actually a class of reviews that includes *walkthroughs* and *inspections*.

THE REVIEW MEETING

- Between three and five people (typically) should be involved in the review.
- Advance preparation should occur but should require no more than two hours of work for each person.
- The duration of the review meeting should be less than two hours.
- Focus is on a work product (e.g., a portion of a requirements model, a detailed component design, source code for a component)

REVIEW MEETING PARTICIPANTS

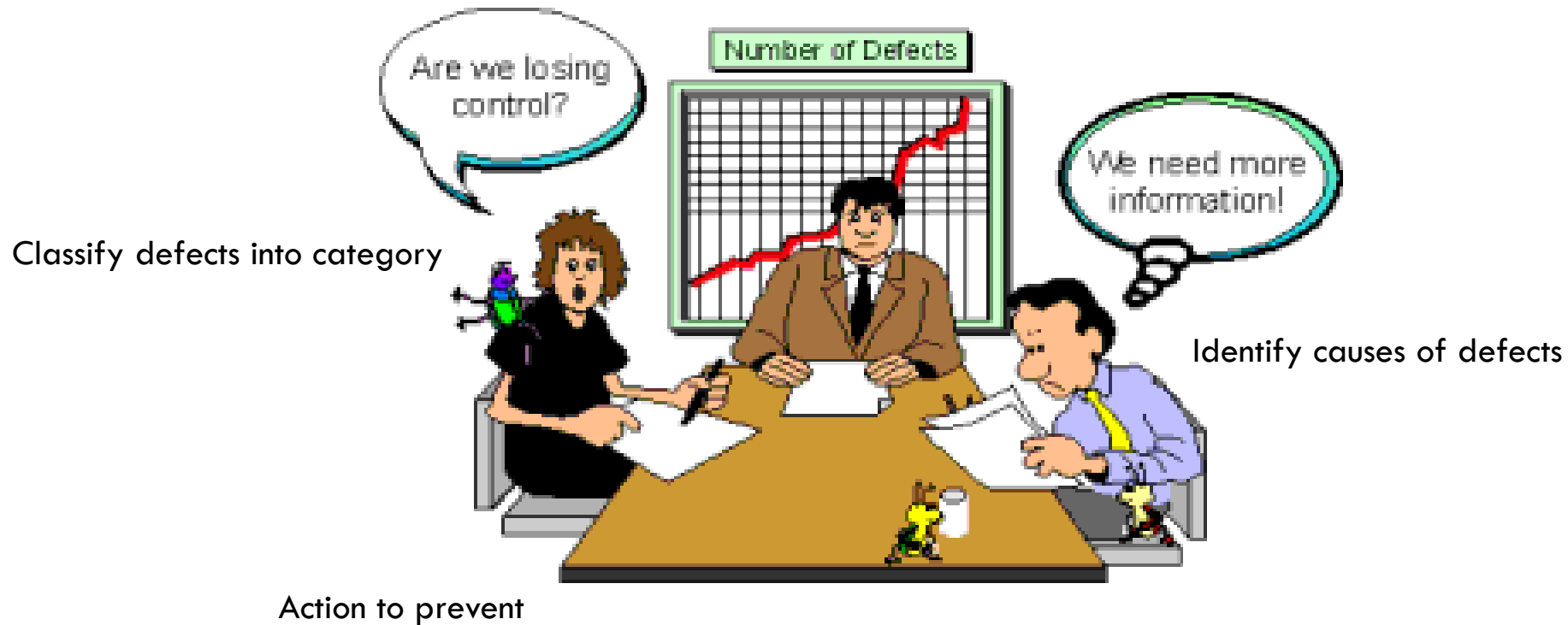
- *Producer*—the individual who has developed the work product
 - informs the project leader that the work product is complete and that a review is required
- *Review leader*—evaluates the product for readiness, generates copies of product materials, and distributes them to two or three *reviewers* for advance preparation.
- *Reviewer(s)*—expected to spend between one and two hours reviewing the product, making notes, and otherwise becoming familiar with the work.
- *Recorder*—reviewer who records (in writing) all important issues raised during the review.



CONDUCTING THE REVIEW

- *Review the product, not the producer.*
- *Set an agenda and maintain it.*
- *Limit debate and rebuttal.*
- *Enunciate problem areas, but don't attempt to solve every problem noted.*
- *Take written notes.*
- *Limit the number of participants and insist upon advance preparation.*
- *Develop a checklist for each product that is likely to be reviewed.*
- *Allocate resources and schedule time for FTRs.*
- *Conduct meaningful training for all reviewers.*
- *¹⁰ Review your early reviews.*

DEFECT ANALYSIS:: WHAT



Definition:

Defect analysis generally seeks to classify defects into categories and identify possible causes of defects so that they can be prevented or detected earlier

DEFECT ANALYSIS:: WHY

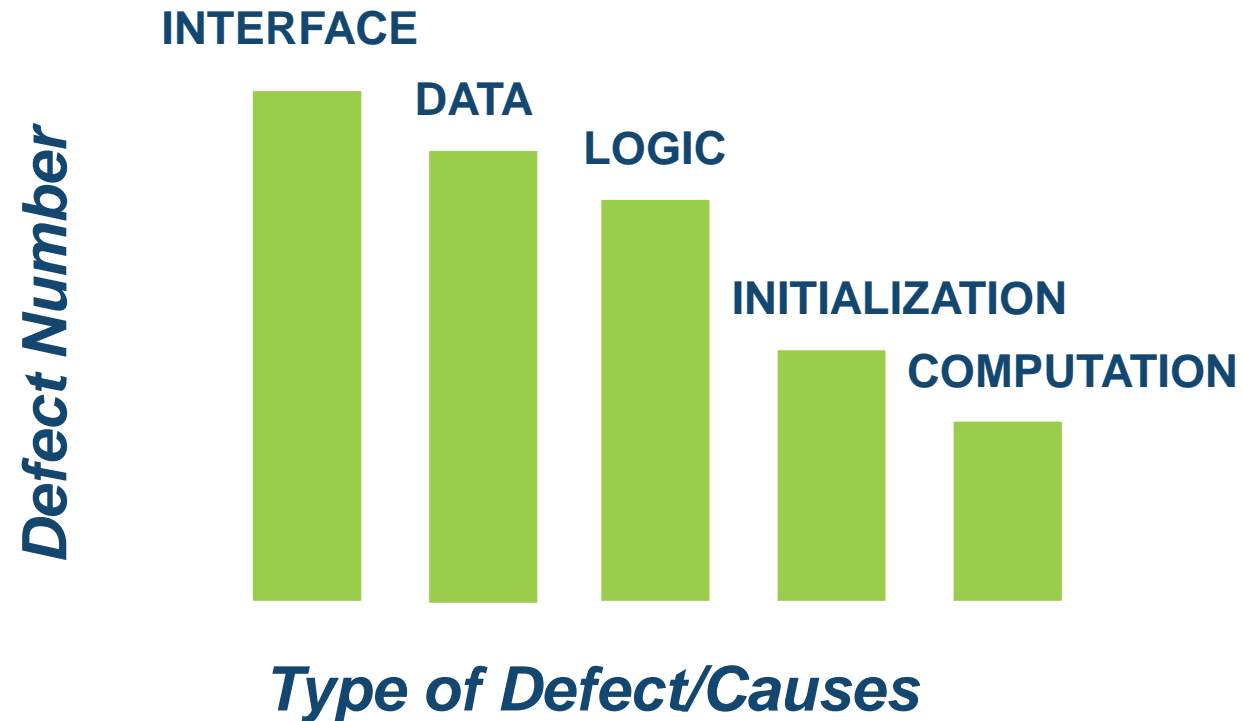


DEFECT ANALYSIS:: HOW

Analysis to identify the main defect types Analysis to identify the causes of defects Identify solutions to attack the causes

CLASSIFY CAUSES OF DEFECTS

Problem/Errors/Defects
Causes

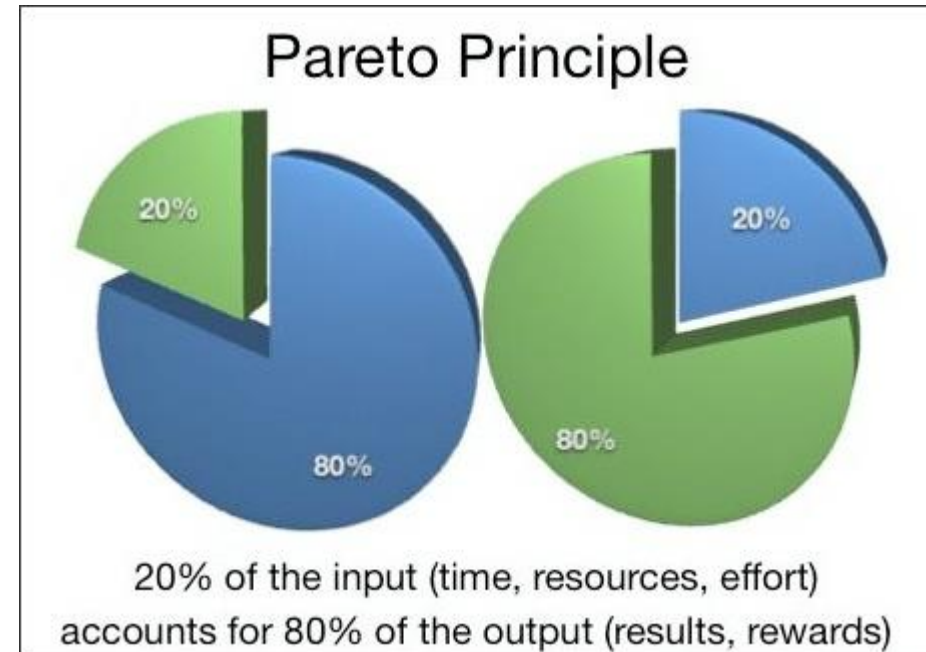


PARETO ANALYSIS

Statistical technique in decision-making used for the selection of a limited number of tasks that produce significant overall effect.



A large majority of problems (80%) are produced by a few key causes (20%).

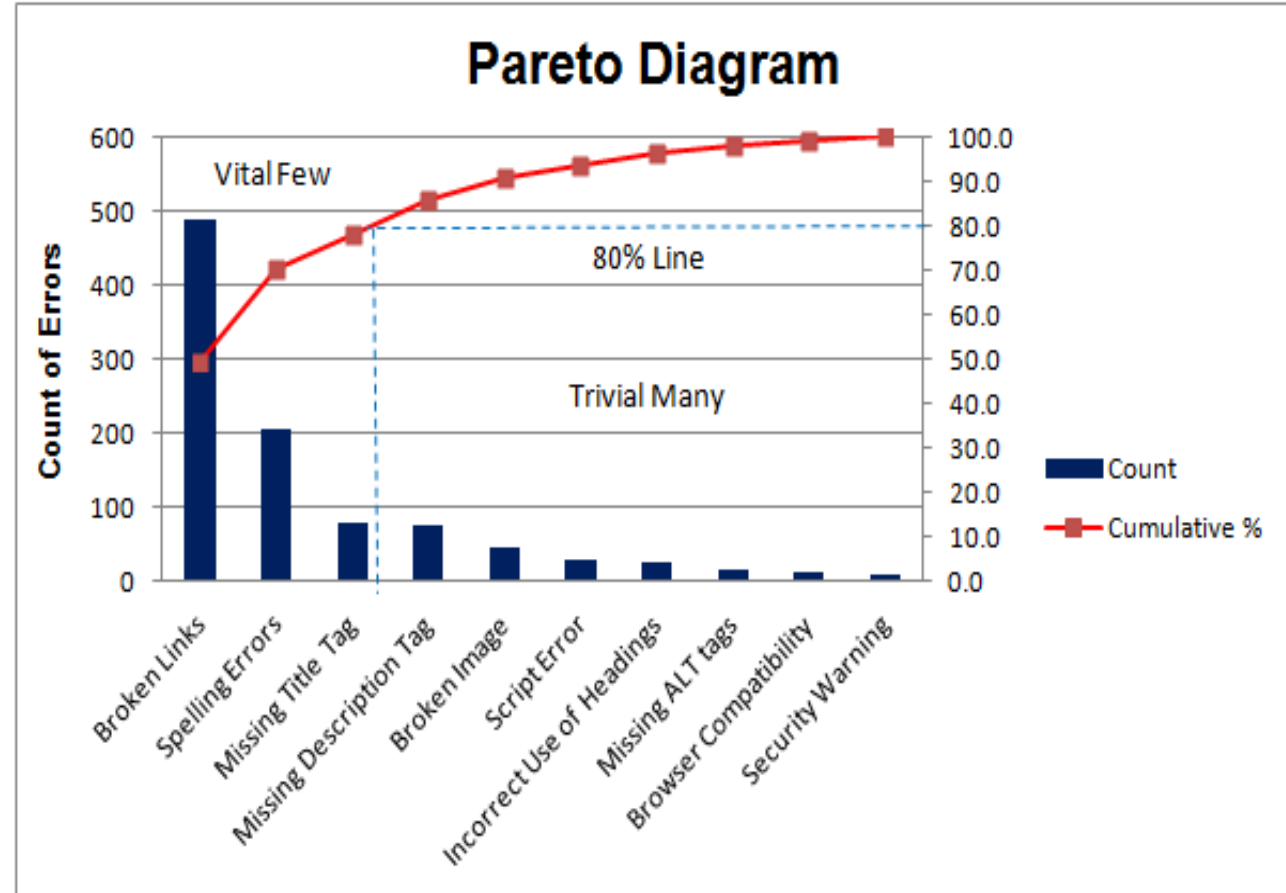


Example:

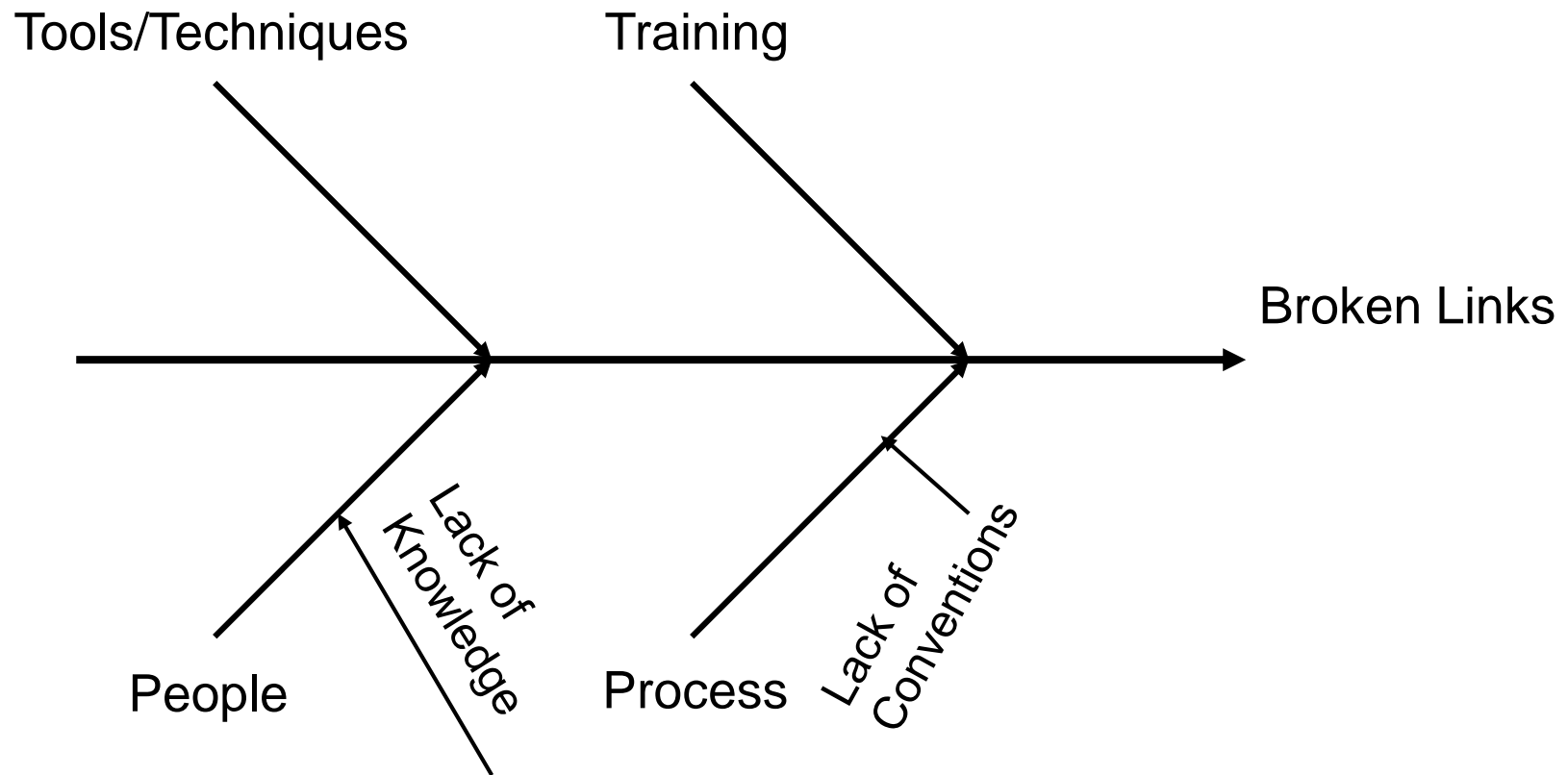
1. 80% of customer complaints arise from 20% of your products and services.
2. 20% of a systems defects cause 80% of its problems.

PARETO ANALYSIS

Error (Cause) /Defect type	Count
Broken Links	349
Spelling Errors	169
Missing Title Tag	79
Missing Description Tag	77
Broken Image	45
Script Error	30
Incorrect Use of Headings	15
Missing ALT tags	14
Browser Compatibility	12
Security Warning	9



CAUSE IDENTIFICATION



Thank You