ILLINOIS INSTITUTE OF TECHNOLOGY

Transforming Lives. Inventing the Future.

www.iit.edu

SOFTWARE QUALITY MANAGEMENT

CSP587

Prof. Dennis Hood

Computer Science

Lesson Overview

- Quality Culture
- Reading
  - Ch. 2 – Quality Culture
- Objectives
  - Analyze quality cost drivers and structures
  - Discuss the organizational structure of software quality functions
  - Analyze the impact of the organization on the delivery of quality
  - Examine the profession of software engineering and the maturity of software engineering organizations

Topics for Discussion

- Discuss the meaning of organizational culture and its role in delivering quality
- Identify cost drivers and structures associated with producing quality and making sound investments in the quality function
- Analyze the underlying principles which foster an organization's ability to deliver quality as well as the structural components of software systems projects
- Discuss the software engineering profession and supporting code of conduct

Week 3
Quality Culture

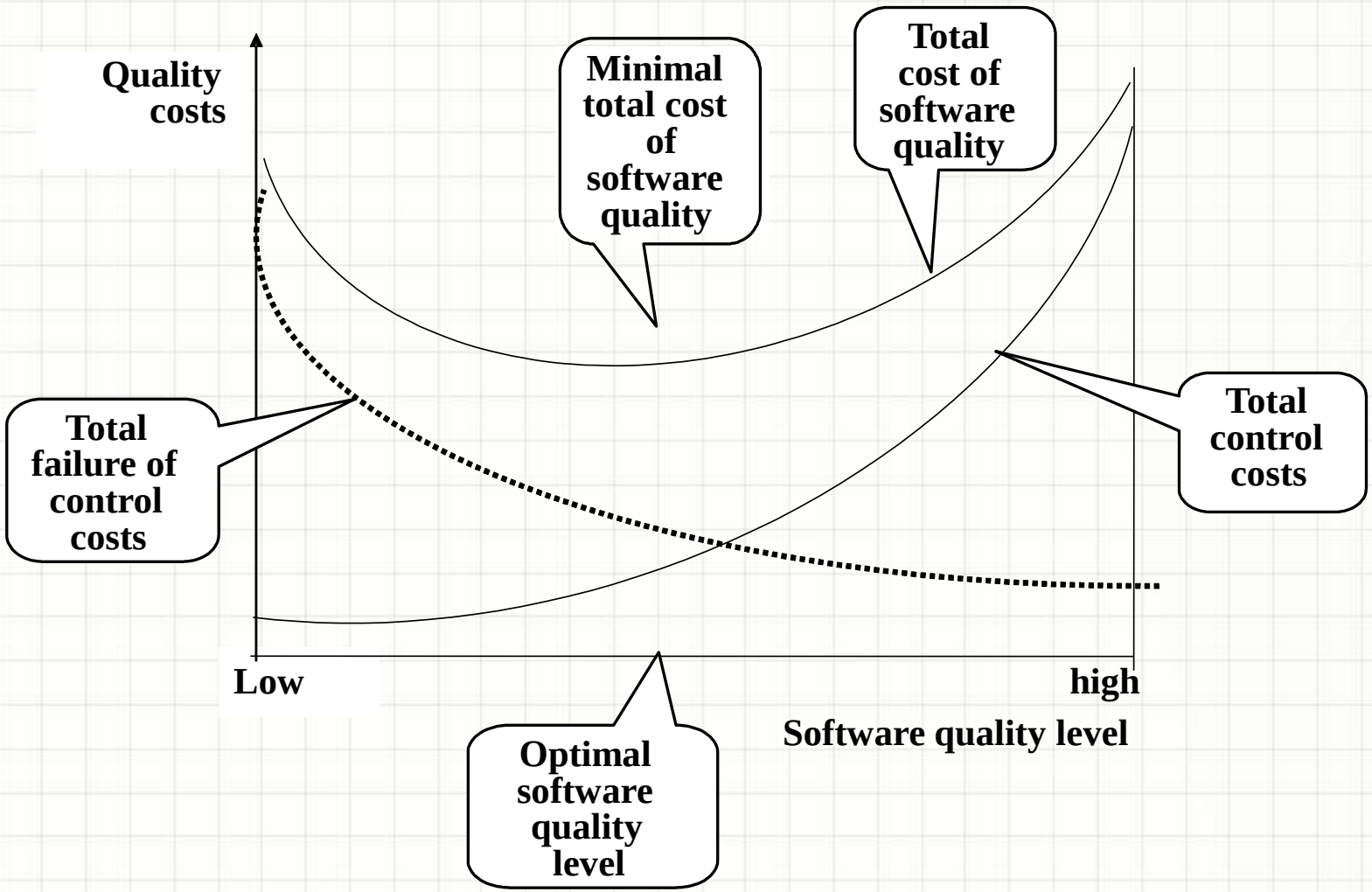
Case: Therac-25

- Life or death consequences (mission critical)
- Significant operator control (infinity?)
- Exception detection (user vs. oper. awareness)
  - Time lag 1: error occurs -> damage done
  - Time lag 2: error occurs -> human awareness
- User error vs. UX design error
- Lack of defensive design (restrict HCI)

Cost / Value

- The cost of quality
  - Create opportunity (hiring, purchasing, training)
  - Apply capability (people, tools, etc.)
  - Over time (time is money)
  - Test and fix
- The cost of a lack of quality (=value?)
  - Rework (haste makes waste)
  - Customer dissatisfaction
  - Legal liabilities (e.g., warranties, lawsuits, etc.)

Total Cost of Software Quality



Prevention vs. Removal

- In the abstract, you can either pay “in advance” to prevent a defect from being inserted, or “pay later” to remove it
- But these costs are not equivalent – and the cost of removal “snowballs” the longer the defect is present in the system under development
  - For example, the cost of hiring the best programmers and training them to use sophisticated tools will outweigh a coding error that is immediately removed
  - But a defective requirement that does not manifest until after the system is in production could have devasting costs to both “excise” from the depths of the design / code and placate the customer

Organizational Focus on Quality

- Mature organizations (a culture of quality) recognize these issues as very real, rather than as hypotheticals,
  - and commit to prevention as not only smart, but essential
- Culture is contagious - new employees tend to adopt
  - If everyone is serious about quality, they will be too
  - But be warned, if no one cares, they won't either

Cultural Principles (Wiegers)

1. Resist pressure (from your boss and client) to rush, cut corners, etc.
2. People must feel that their work is appreciated
3. Continuing education is the responsibility of each team member
4. Involve the client proactively
5. Share your vision of the final product with the client
6. Embrace continuous improvement
7. Procedures establish a common culture of best practices
8. Quality is job one – and leads to long-term success
9. Ensure that defects are discovered internally, not by clients
10. Repeatedly go through all phases – except coding for which the goal should be to do it only once
11. Controlling error reports and change requests is essential to quality and maintenance
12. Measure what you do – to achieve measurable improvement
13. Focus on what is feasible
14. Prioritize, then act with urgency

Project Dimensions (Scope Triangle)

- Boundaries
  - Capacity / capability to produce output (staff / tools)
  - Schedule - available time
  - Budget (cost) - available funds
- Output
  - Scope – the volume and complexity of the system's features (utility)
  - Quality – the degree of adherence to those requirements
- Scope Triangle
  - These are interconnected – equilibrium is maintained
  - Extending or contracting boundaries adjusts the value of the output accordingly (the area of the triangle)

SW Engineering Code of Conduct

- A set of rules set by a “society” to govern the actions of its “citizens” (e.g., do no harm)
- Software Engineers must:
  - Act consistently with the public interest
  - Act in the best interest of client and employer
  - Meet the highest professional standards possible
  - Maintain integrity and independence in judgement
  - (Managers and leaders) must promote an ethical approach
  - Champion the integrity and reputation of the profession
  - Be fair and supportive of their colleagues
  - Participate in lifelong learning and promote an ethical approach
- A code of conduct has practical, tangible benefits:
  - Basis for training, education, and certification
  - Builds trust in, and credibility for, the society
  - Protects “whistle blowing”, which is the epitome of organizational maturity – the welcoming of criticism as an opportunity to grow

The Quality Organization

- Management structure
  - Senior management
  - Middle management
  - Project management
- The QA structure
  - QA management
  - Testers
  - SQA professionals and interested practitioners
    - Trustees, committee members, etc.

Management's Role in SQA

- Senior management
  - Own software quality policy
  - Assure adherence
  - Fund the SQA function
- Middle management
  - Manage SQA tasks
  - Allocate QA resources
  - Control and improve performance
- Project management
  - Develop plans
  - Determine project requirements
  - Schedule SQA tasks
  - Manage test process