Introduction to Software Engineering (SENG 300 W2020)

Lecture 02

Sohaib Shahid Bajwa

Acknowledgement

- Some of the contents are adapted from material by Manzil-e-Maqsood

- Software Engineering – A practitioner's approach by Roger S. Pressman and Bruce. R. Maxim
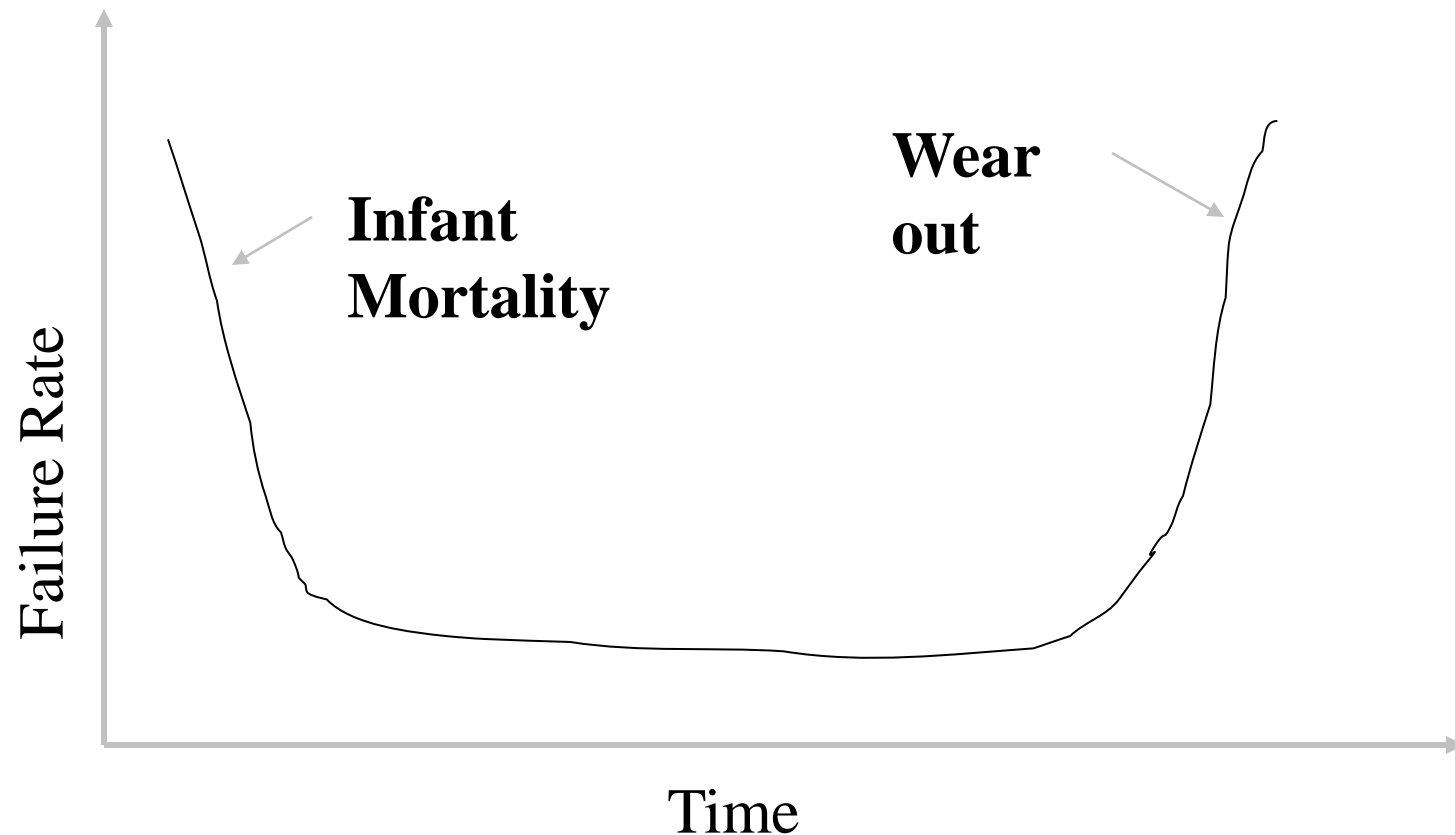
Hardware Vs. Software
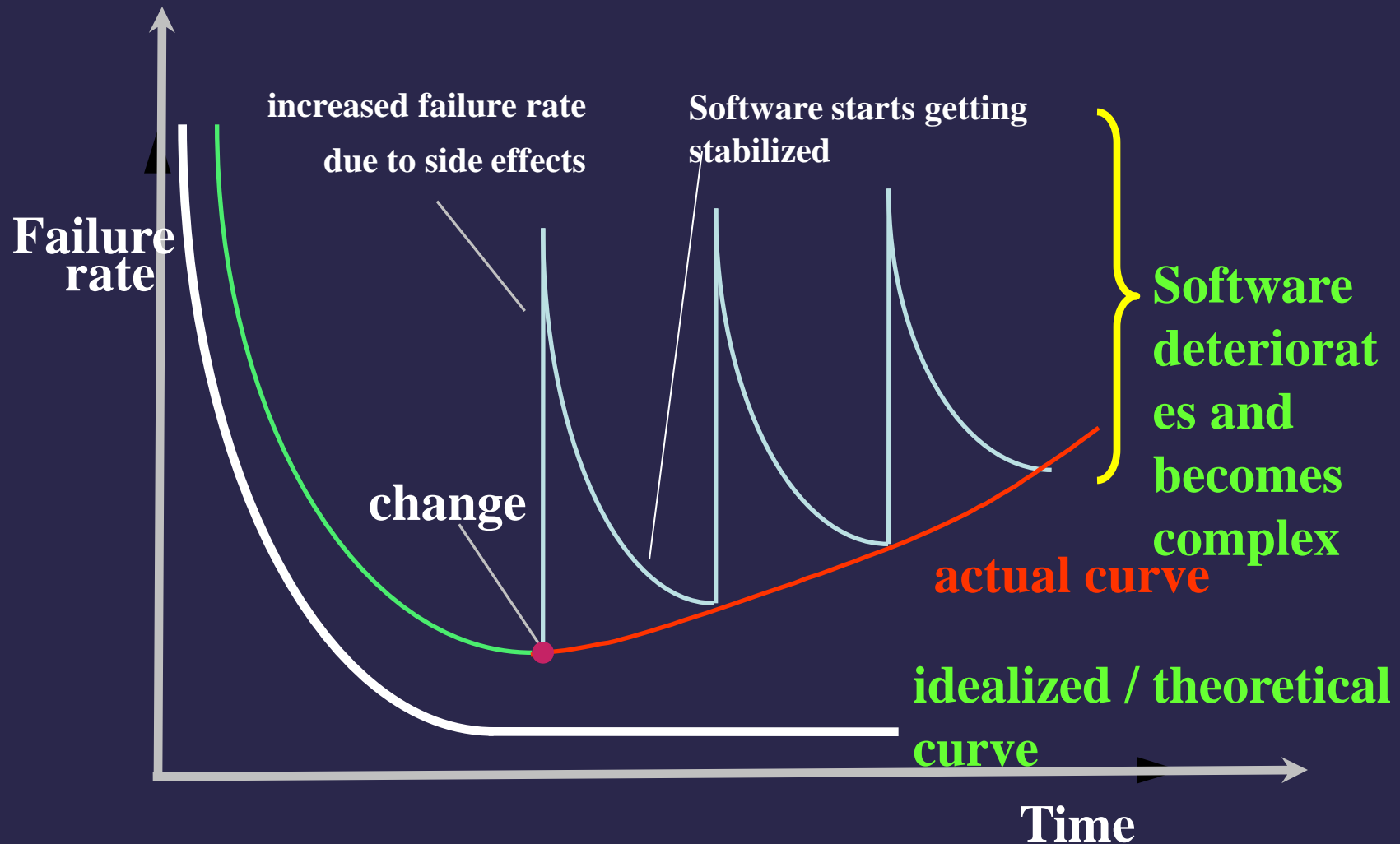
Hardware

- Wears out
- Replaceable spare parts

Software

- Does not wear out
- No spare parts

Hardware Vs. Software …

Infant Mortality

Wear out

Failure Rate

Time

Hardware Failure Curve

Wear Vs. Deterioration

increased failure rate
due to side effects

Software starts getting
stabilized

Failure
rate

change

Software
deteriorat
es and
becomes
complex

actual curve

idealized / theoretical
curve

Time

No Silver Bullet

- Fred Brooks, "The Mythical Man-month"
- There is no short cut to software development
- Software can be disastrous to managers in the form of:
  - Schedule slippages
  - High costs
  - Delivery of unreliable systems

Begin upright and upfront with a systematic approach to building software

Class Activity 1

- You are leading a small software development team responsible for overhauling the D2L system at Univ. of Calgary for course management.

- What are the activities your team will be involved with?

- Instructions:
  - Make a team of 5 people
  - Time: 5 minutes

Effort Breakdown
Breakdown of activities involved in software development
(Caper Jones)

- Project Management
- Requirements
- Design
- Coding
- SQA
- SCM
- Integration
- Miscellaneous

Effort Breakdown

Breakdown of activities involved in software development
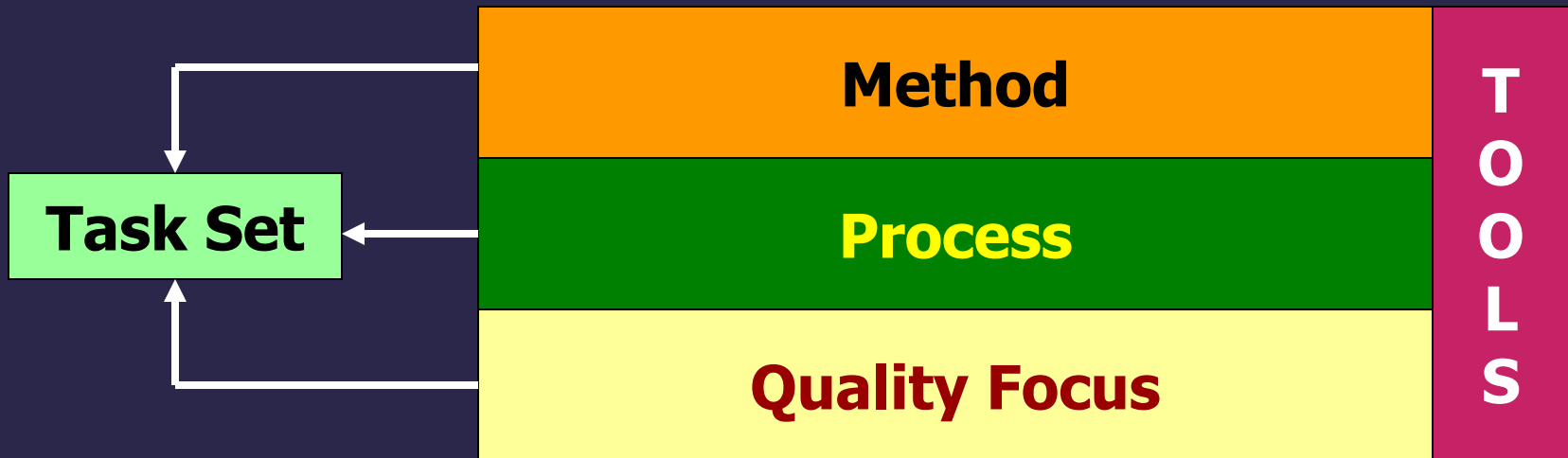(Caper Jones)

- Project Management   8.08%
- Requirements   14.43%
- Design   11.36%
- **Coding**   **13.50%**
- SQA   30.64%
- SCM   13.02%
- Integration   6.54%
- Miscellaneous   ~3%

SE is NOT just Coding !!

Software Engineering Layers

- **Software Engineering is a layered technology consisting of the following layers:**
  - **Tools**
  - **Methods**
  - **Process(es)**
  - **Quality Focus**

Software Engineering Layers

Quality focus

- Bedrock that supports software engineering

- Organizational commitment to quality

- Demands that processes be defined for rational and timely development of software

Process

- Foundation of software engineering and holds all the other layers together
- **Defines the tasks** to be performed **and the order** in which they are to be performed along with deliverables and milestones to be produced
- Ensures
  - Application of technical methods - tasks
  - Production of work products
  - Establishment of milestones
  - Management of change
  - Quality focus
- Example: Waterfall process

14

Task Set

- A collection of SE work tasks, project milestones, work products etc.

- Defined by process

- Technically described in detail by methods

- Ensured by quality control activities

- Examples:

     Requirements Analysis; Design; Coding; Testing

15

Method

- Provide **technical how-to's** to carry out the tasks defined by the process
- Methods encompass tasks like:
  - Requirements Analysis
  - Design
  - Program construction
  - Testing
  - Support

Examples:

Use Case Model for Requirements Analysis Task, Class Diagram and GUI Style Guide for Design Task, Coding Conventions for Coding Task, Test plan and Test case execution for Testing Task

16

Tools

- SE tools provide automated or semi-automated support for the process and the methods
- CASE tools
  - A system for the support of software development
  - A set of integrated tools in which information created by one tool can be used by another
  - Examples: Rational Development Suite, MS Visio, MS Project , BugZilla,etc.

Difference between a tool and a method

- Standard deviation is a **method** to estimate and measure dispersion of data
- Excel is a **tool** that can be used to calculate standard deviation

Software Process

- A process is a collection of activities, actions and tasks that are performed when some work product is to be created.

- Process should not be considered as a rigid process in the context of software engineering

Software Engineering Framework…

- A common process framework is established by defining a small number of framework activities that are applicable to **all** software projects, regardless of their size or complexity.

- Framework activities are managed and controlled through the use of SQA, SCM and measurement.

The Process Framework

- Establishes framework activities that are applicable to all software project

- Communication
- Planning
- Modeling
  – Analysis of requirement
  – Design
- Construction
  – Code generation
  – Testing
- Deployment

Umbrella Activities

- Software project tracking and control
- Risk management
- Software quality assurance
- Technical reviews
- Measurement
- Software configuration management

Software Myths

- Affect managers, customers (and other non-technical stakeholders) and practitioners
- Are believable because they often have elements of truth,

but …

- Invariably lead to bad decisions,

therefore …

- Insist on reality as you navigate your way through software engineering

Class Activity 2

- Software and Software Myths

- See handouts

- Instructions:
  - Spend first 10 minutes solving it as an individual
  - Then Discuss with your neighbor and agree on an answer (Time: 5 minutes)

What we learn today

- Hardware vs. Software

- Software development is a multi-activity process. It is not simply coding

- Difference between Framework activities and umbrella activities

- Software Myths vs Reality

SENG 300 W2020 TopHat

- **SENG 300 - Software Engineering - Winter 2020**

- Join Code: **405187**

Next

- Modeling
- Understanding Requirements