S.E.

Chapter 1.

- Software should be easier, faster and less expensive to build and maintain.
- SE is a framework that encompasses a process, a set of methods, and an array of tools to build computer software
- Software is both the product as well as the vehicle that delivers it.

eg: product: facebook, (software
vehicle: browser or app (software) } delivers information
medium: network is also software

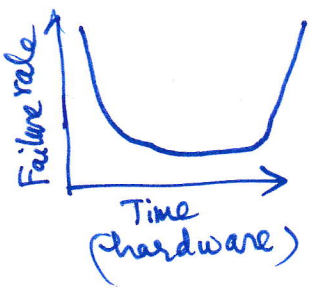
product: Microsoft word
vehicle: Windows / Linux

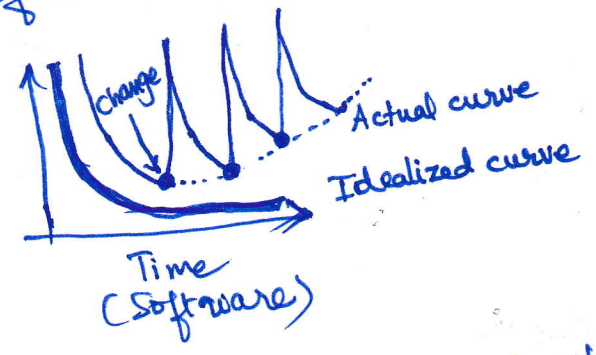
Characteristics of Software:

I Software is engineered (developed) not manufactured

Characteristics of software

2. Software does not wear-out but deteriorates
good design reduces software deterioration
deterioration is caused due to change
All failures in software indicates errors in design





3. Most software is custom-built rather than component-based
- reusability is low (via use of standard libraries)

2 of 5

(Seven) Categories of Computer software → (Application domains)

1. System software
2. Application software
3. Engineering/scientific software
4. Embedded software
5. Product-line software
6. Web Applications
7. AI software

New areas (categories)
- Open world computing
- Net sourcing
- Open source

(Find examples for each category) ?

Legacy software: they are characterized by logetirity and business criticality (~~indispensible~~ indispensable) and sometimes poor quality

Types of changes to legacy systems
- due to new computing environment
- due to new business requirements
- due to interoperability with other modern systems
- due to ~~no~~ viability in a network environment

Goal of SE: devise methodologies that accomodates continuous change (evolution) of software and cooperation among different softwares.

~~Ashok Kumar Reddy~~

Software Engineering

- Understand the problems before developing solutions. Listen to all stakeholders.
- Design is important as system have become very sophisticated
- Software should exhibit high quality
- Software should be maintainable

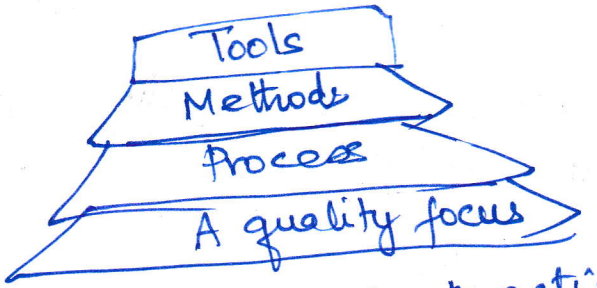
Defⁿ. IEEE of SE.

(1) The application of systematic, disciplined, quantifiable approach to development, operation and maintenance of software
i.e., Application of engineering to software

(2) Study of approaches as in (1)

Establishment of sound engineering principles in order to obtain economically software that is reliable and works efficiently on real machines

Layers



what? when? How?

Process is a collection of activities, actions & tasks that are carried to create a work product

Activity — broad objective (eg comm. with stakeholders)

~~Fast~~ Actions — creates a major work product (eg: architectural design)

Tasks — small but well defined objective (eg: unit test)

Action is a collection of tasks

Process Framework consists of framework activities and umbrella activities

Framework activities (for all projects)
- communication
- Planning
- Modeling
- Construction
- Deployment

Umbrella activities (activities throughout the process)
- software project tracking and control
- Risk mgmt.
- Software Quality Assurance
- Technical Reviews
- Measurement (process, project, product)
- Software configuration mgmt.
- Reusability mgmt.
- Work product preparation and production.

How do process models differ?
- Over all flow of activities, actions and tasks with interdependencies
- Degree to which actions & tasks are defined
- Degree to which work product are identified and required
- Overall degree of detail & rigor of process
- Degree to which customers & stakeholders are involved
- Level of autonomy given to software teams
- degree to which team organization & roles are prescribed

What are general principles of SE that apply to any process framework)
Essence of SE practice
- Understand the problem
- Plan a solution
- Carry out a plan
- Examine result for accuracy.

General Principles guiding SE-

1. → The reason it all exists
   → Keep it simple stupid!
   → Maintain the vision
   → What you produce, others will consume
   → Be open to the future
   → Plan ahead for reuse
   → Think!

Software Myths:

Management
- Standards and procedure have everything to build software
- Adding programmers can ensure timely delivery when behind schedule
- Outsourcing a project can solve my problems

Customer
- A general statement of objectives is sufficient to start programming
- Requirement change continually, but changes can be easily accomodated

Practitioner
- Once a program is written and deployed, our work is done
- Quality can only be assessed after implementation
- The only deliverable work product is a working program.