Software Modeling and Analysis

Reading Log

‘Software’s Chronic Crisis’

by W. Wayt Gibbs

from Scientific American

경영정보 04 조대현

Overview

- Software's Chronic Crisis
    - Encountered Problems during Developing Software, Systems
    - Examples
    - Issues of Industrial growth

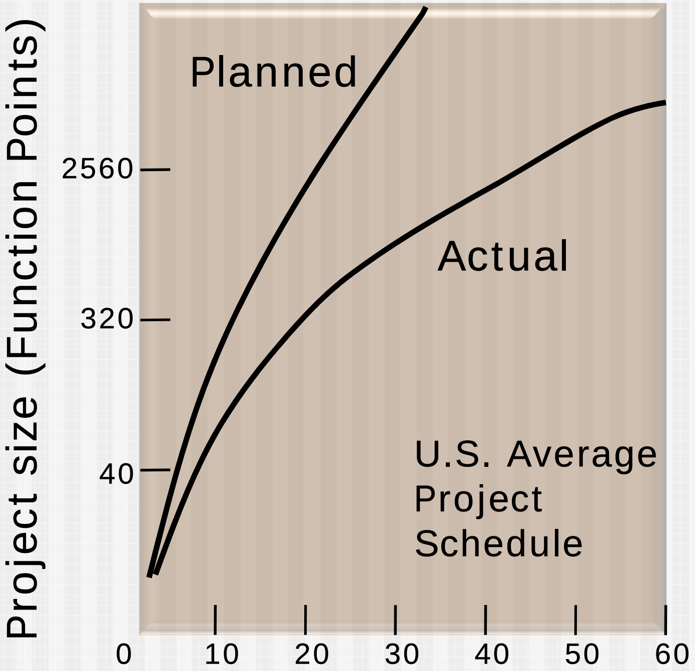
Measurements of the software

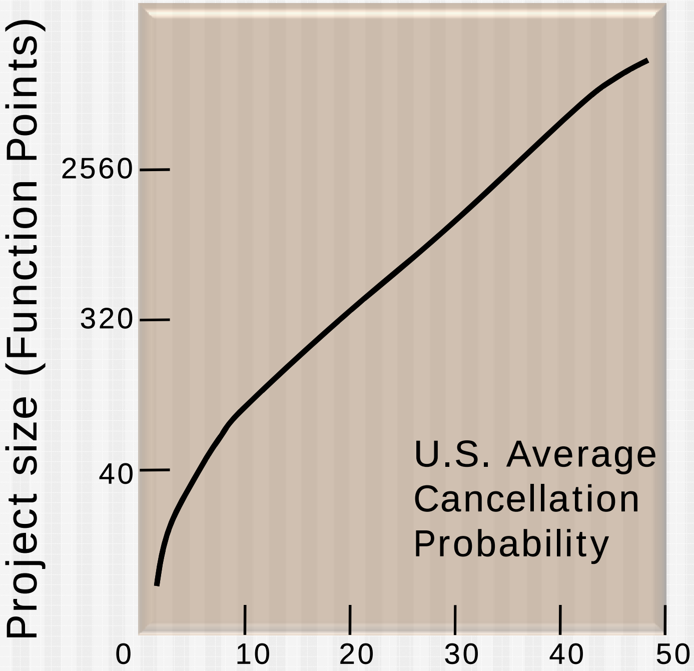
- Measuring Quality of Software
- Guarantee of Consistency
- Predict budget and time expend

Example :
Denver airport

- Baggage-handling system
  - 10 times bigger than Heathrow
  - 20 different airlines
- Failure!
  - Couldn’t perdict when it’s done
  - Overcosts, Overdue.

Study shows





SOURCE: Software Productivity Research

Study shows

- 33% drawback rate..
  - 6 large-scale software are put into operation, while 2 are canceled
- 50% overdue rate..
- 75% operation failures

Solution: Set a Goal

- **“Software Engineering”**
  - “the application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software.”
  - 1968, NATO Science Committee

Advance: Software Engineering

- Measurements of quality of software
  - Mathematical solutions
- New Found Needs
  - Interchangeable, reusable software parts

Advanced Industrial

- Bigger
  - Doubled the code
- More Elaborate
  - ‘Clementine’ – DOD
  - Safty-Critical Software

Advanced Industrial (2)

- “distributed systems”
  - Programs that run cooperatively on networked computers
- System integration
  - Reusability, Compatibility

Examples
Distributed System

- California's Driver and Vehicle Registration System
  - Simple ?

Examples
Distributed System (2)

- American Airlines
  - ‘SABRE’
    - $2-billion flight reservation system
  - Flight, Hotel, Car integrated System
    - Marriott, Hilton and Budget.

IBM Research

- 24 Leading companies
  - Developed large distributed systems

  - 55% Cost more than expected
  - 68% Overran schedules
  - 88% Redesigned

Tough to build Distributed System

- Complexity and Fragility
- Growing complexity
- “You can’t build skyscrapers using carpenters.”

  - Bill Curtis

IBM Case

- Advanced Automation System (FAA)
- Great challenge of 90’s

CMM

- Capability Maturity Model (CMM)
  - Software Engineering Institute 1991
  - Five-level scale
  - 261 organization rated
  - 75% - level 1
  - Mandated level 3 (U.S. Air Force, NASA)

Solutions

- Beta Test
  - Microsoft, Windows
- Proto Type
- Mathematical Formal method
  - Praxis

Solutions (2)

- Mathematical formal methods(2)
  - GEC – Train System in France

- “Clean-room approach”
  - Ericsson Telecom

Further Issues

- Need More Supports

- Developing Software Componentes
  - Profitability ?
    - Brad Cox

Further Issues

- Varity uses of Software Parts
- Real Programmers should do..
- Software development undergo Industrial evolution.

Farewell!

- Thank you For Listening.