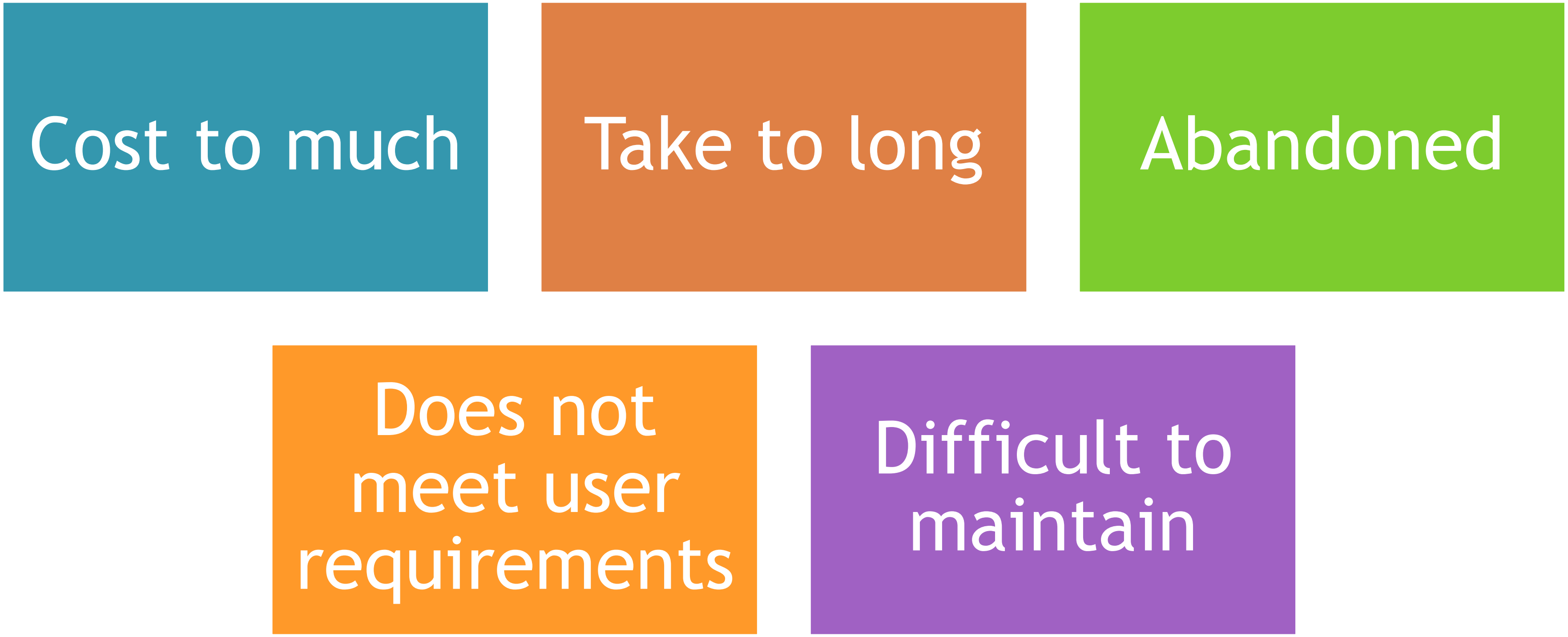
Software Development Introduction

Ridi Ferdiana
ridi@acm.org
Version 1.0.0

Software Crisis



Cost to much

Take to long

Abandoned

Does not meet user requirements

Difficult to maintain

Software Challenges

Creativity

Estimation

Intangible

Complex

Software Maintenance Issues

- Management get New Ideas
- Management proposes new benefits
- User request changes
- Business process changes
- 3rd Party Software make new releases
- Developers idea
- Hardware/Software updated
- Laws changes

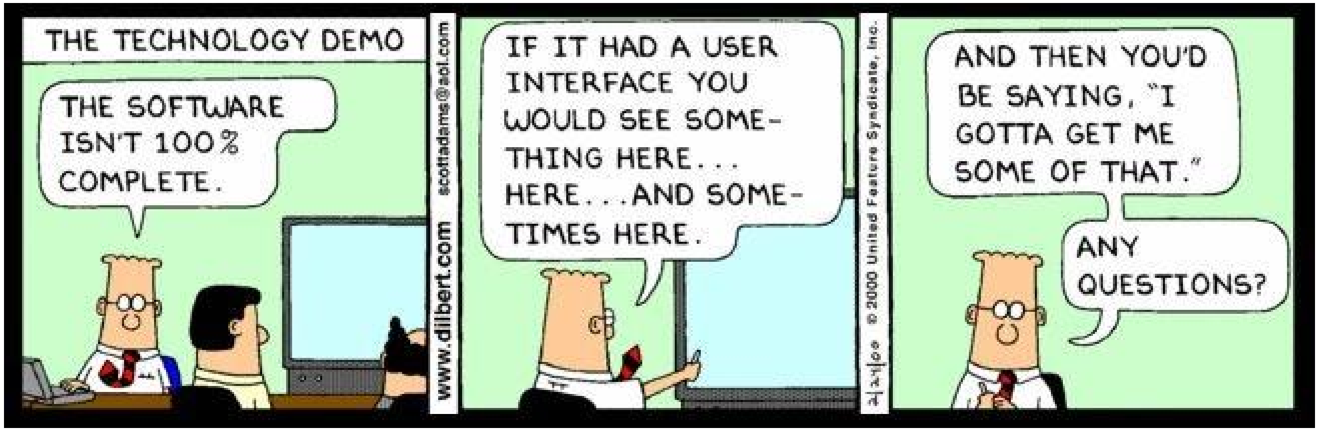
Software Suddent Dead Situation

- Tight schedule
- Staffing problem
- Requirement is too high
- Small Budget

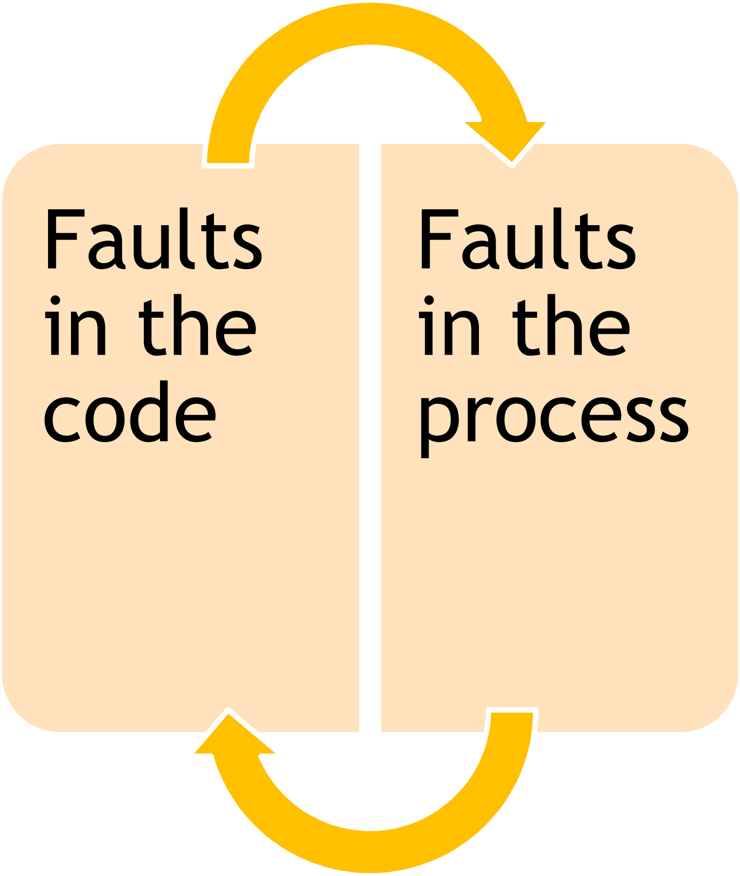
The failure of the software development
Is more that 50%, - Ed Yourdon

Software is wicked problem

“What should these system do”
Rittel & Webber (1973)


THE TECHNOLOGY DEMO
THE SOFTWARE ISN'T 100% COMPLETE.
scottadams@aol.com
www.dilbert.com
IF IT HAD A USER INTERFACE YOU WOULD SEE SOMETHING HERE... HERE...AND SOMETIMES HERE.
© 2000 United Feature Syndicate, Inc.
2/24/00
AND THEN YOU'D BE SAYING, "I GOTTA GET ME SOME OF THAT."
ANY QUESTIONS?

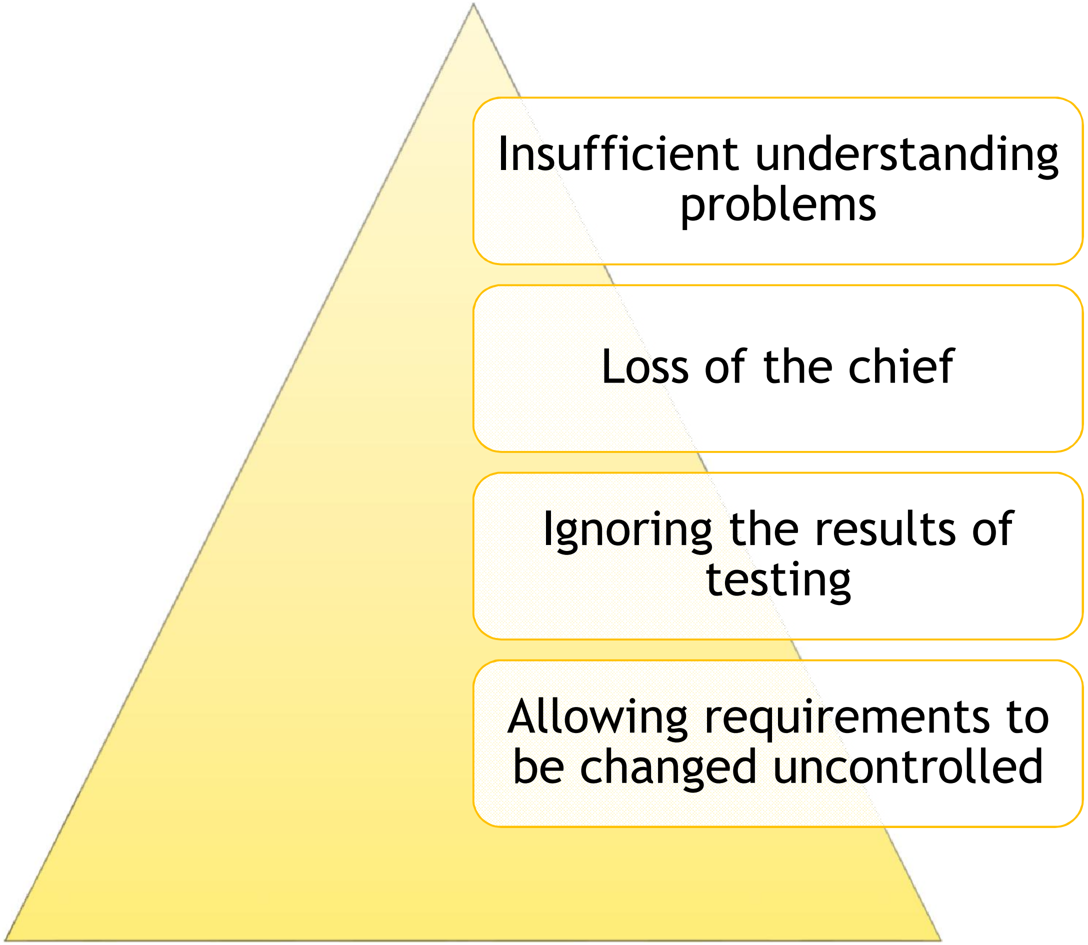
Software failure types



Faults in the code

Faults in the process

Main Failures Reason



Insufficient understanding problems

Loss of the chief

Ignoring the results of testing

Allowing requirements to be changed uncontrolled

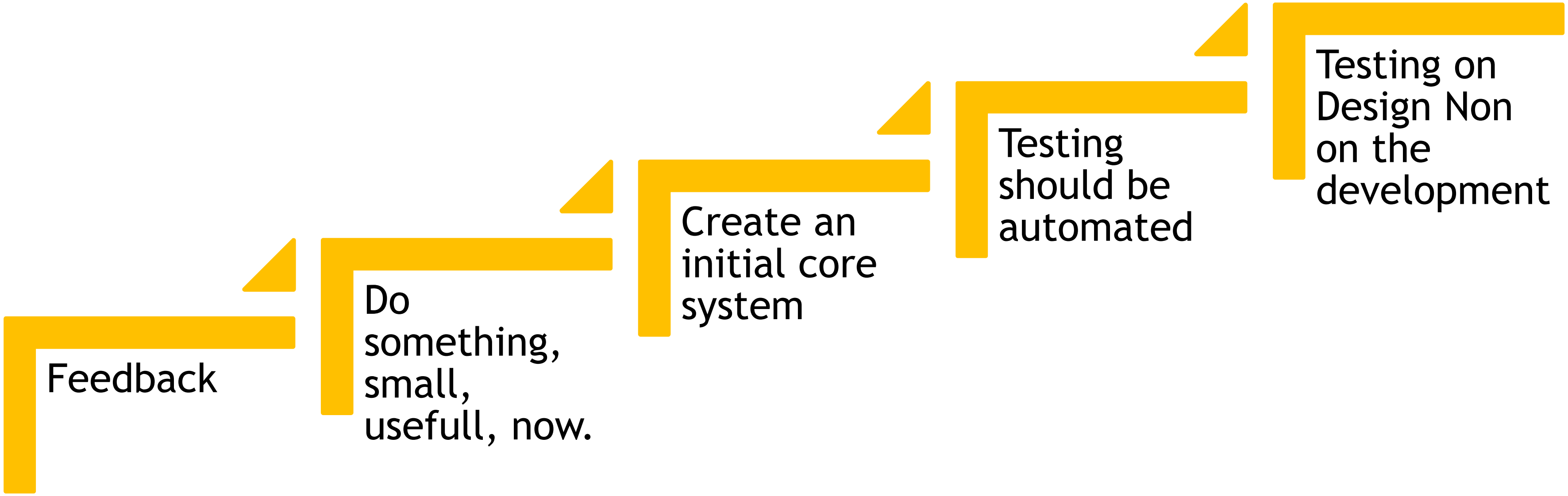
Software Development Issues

- Lacked science and engineering practice
- Ad hoc and verbal specification
- Design method were inadequate
- Testing were ignored
- Too much decoration
- Unable to say no ‘to’ client
- Past technology for future implementation

Part of Software Engineering

- Hamilton said first on NATO Conference in 1968
- 1972, Djikstra said about software crisis. A difficulty to build a software on a large scale
- Software crisis era
  - 90’ limited tools and technical difficulty
  - 2K’ user and usage expectation difficulty

Software Crisis ‘solution’



Feedback

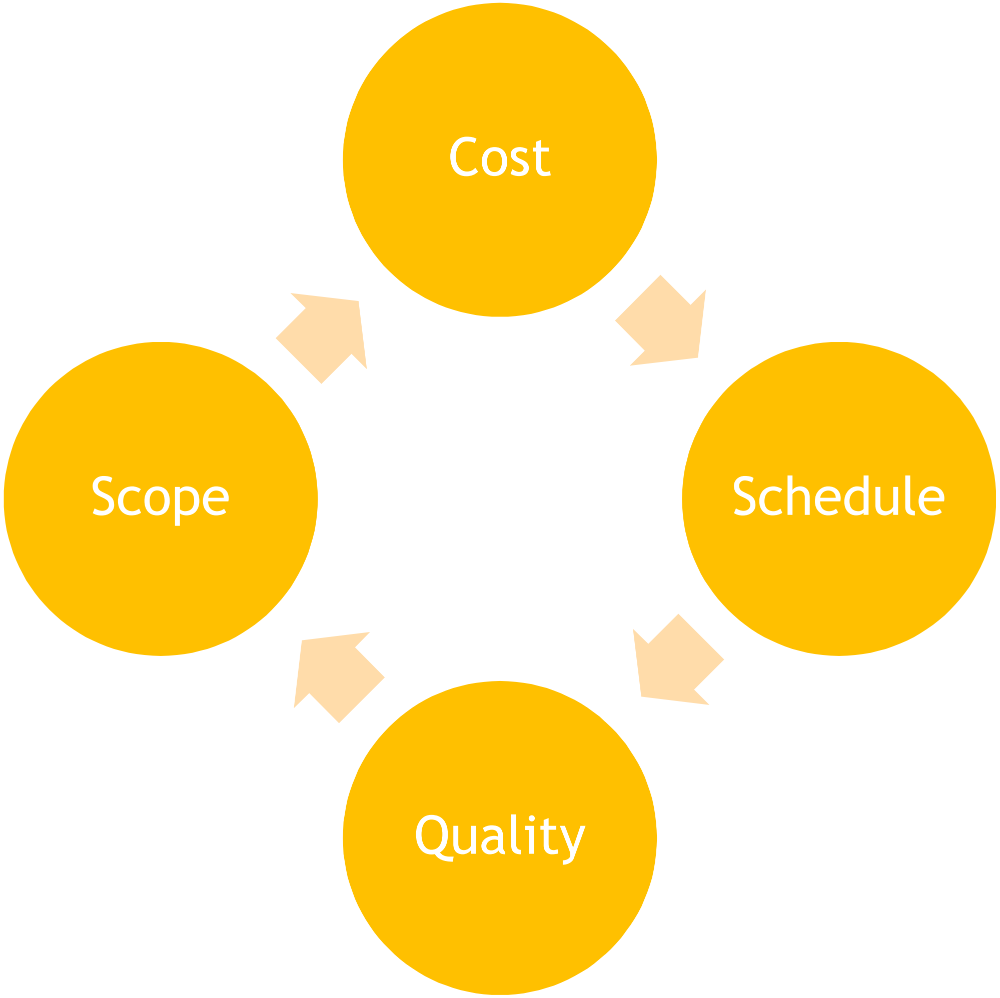
Do something, small, usefull, now.

Create an initial core system

Testing should be automated

Testing on Design Non on the development

Four Variables on Software Development



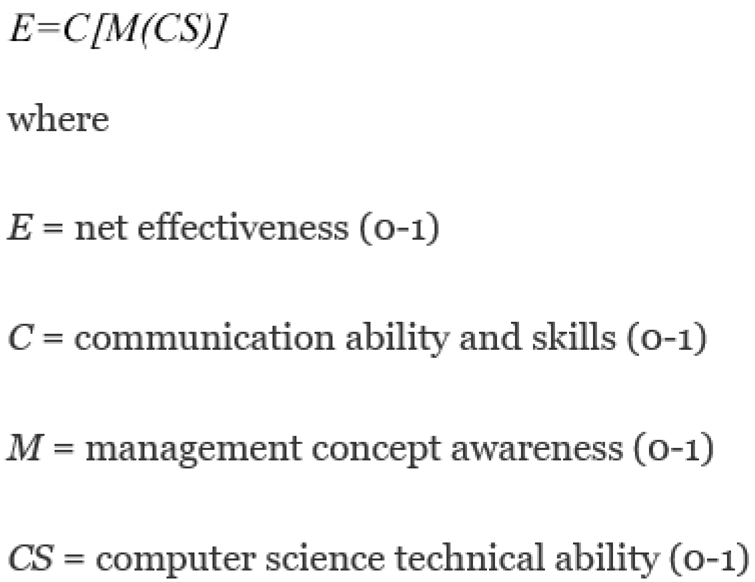
Cost

Schedule

Quality

Scope

Effectiveness on software development



$E=C[M(CS)]$

where

E = net effectiveness (0-1)

C = communication ability and skills (0-1)

M = management concept awareness (0-1)

CS = computer science technical ability (0-1)

Software Development Organization

- Startup / Smaller Company
  - Multiple roles on a peoples
  - Sporadic management
  - Development survival model
  - Revenue based software
- Enterprise
  - Specific roles
  - Comply with law or regulation
  - Maintenance survival model
  - Internal use only software

Key points

- Software development as a part of software engineering
- Software development similar names software construction; programming; coding + design + testing
- Software crisis can be solved through technical way or management way

References

- Crookshanks, E. 2015. Practical Software Development Techniques: Tools and Techniques for Building Enterprise Software. Apress
- Parsons, D. 2015. Rapid and Agile Development. IGI Global
- Jensen, W.R. 2014. Improving Software Development Productivity: Effective Leadership and Quantitative Methods in Software Management. Prentice Hall