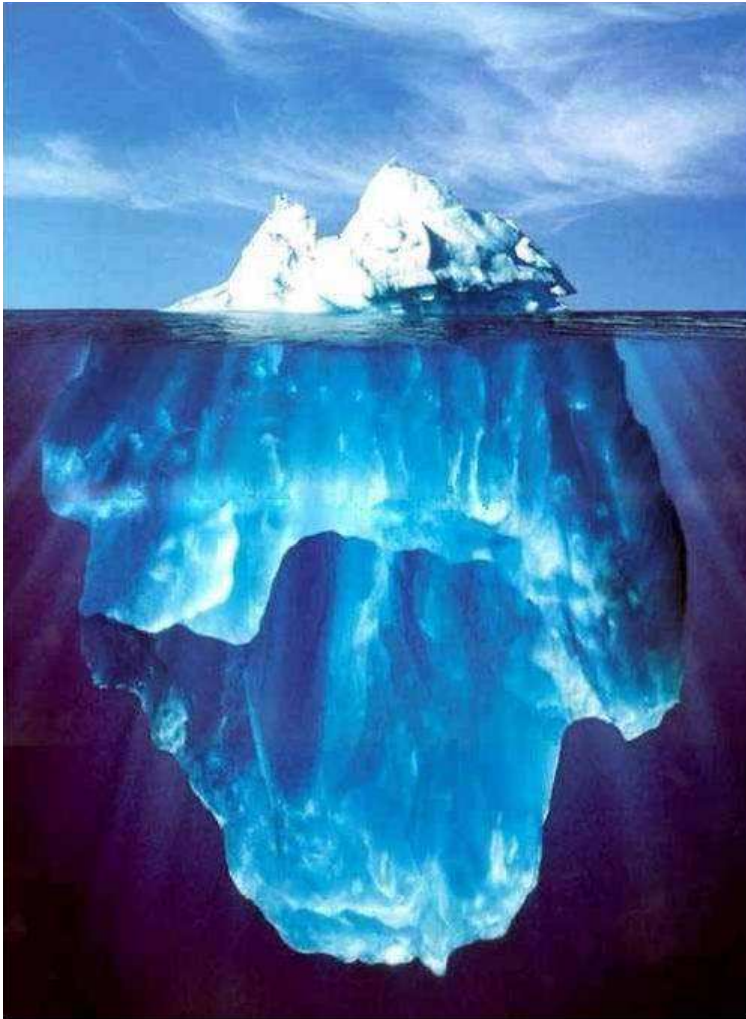


Topics

- Background
- Risk and Software Risk
- Risk Management
- Risk Management in Practice

SOFTWARE ENGINEERING IN PRACTICE #2



Introduction to

Risk Management

#1. Background

#2. Risk and Software Risk

#3. Risk Management

#4. Risk Management in Practice

Q. #1

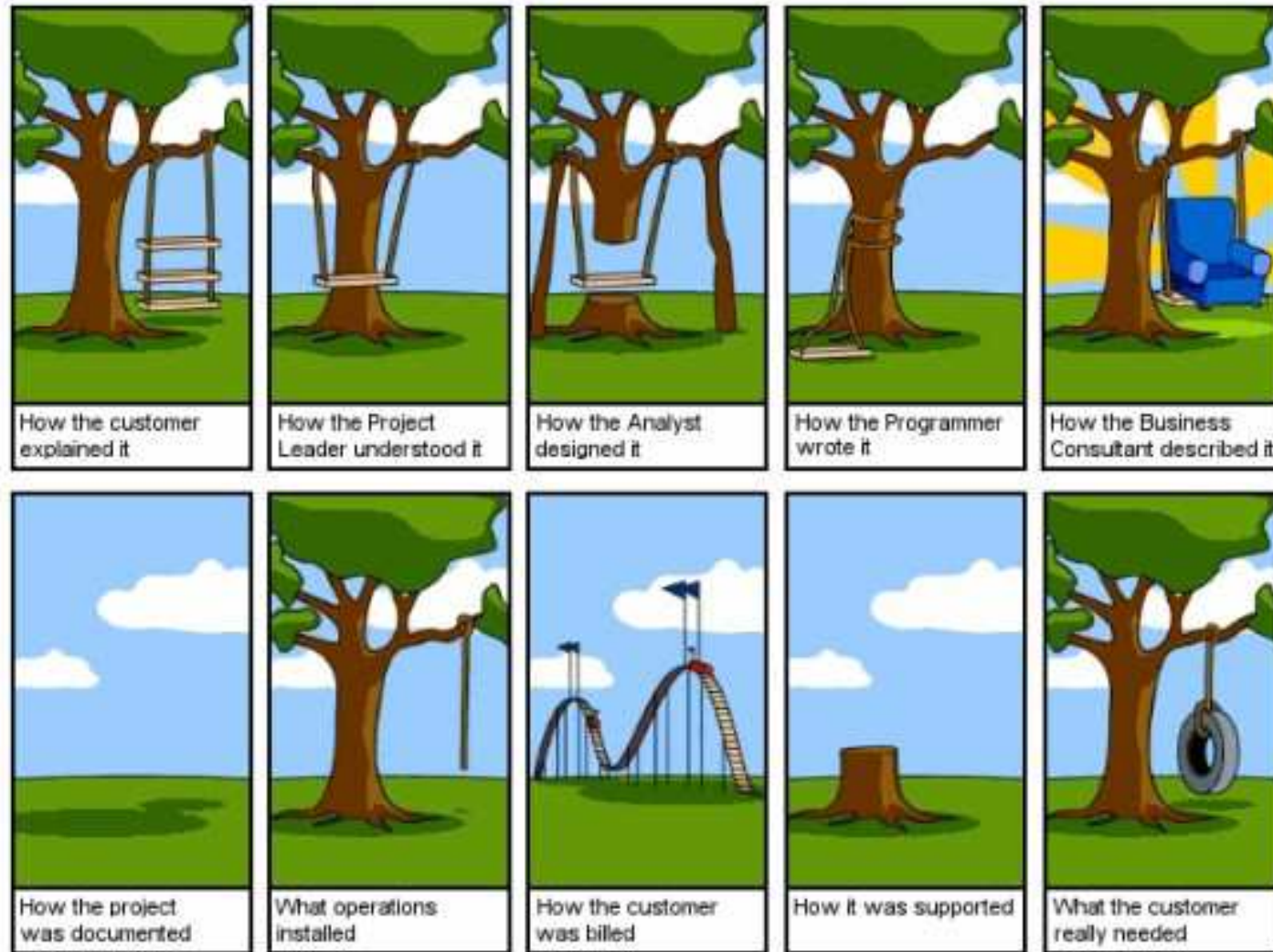
- **Are you sure** that the *software* you built will run properly in accordance with the *requirements*?

a. Yes, off-course

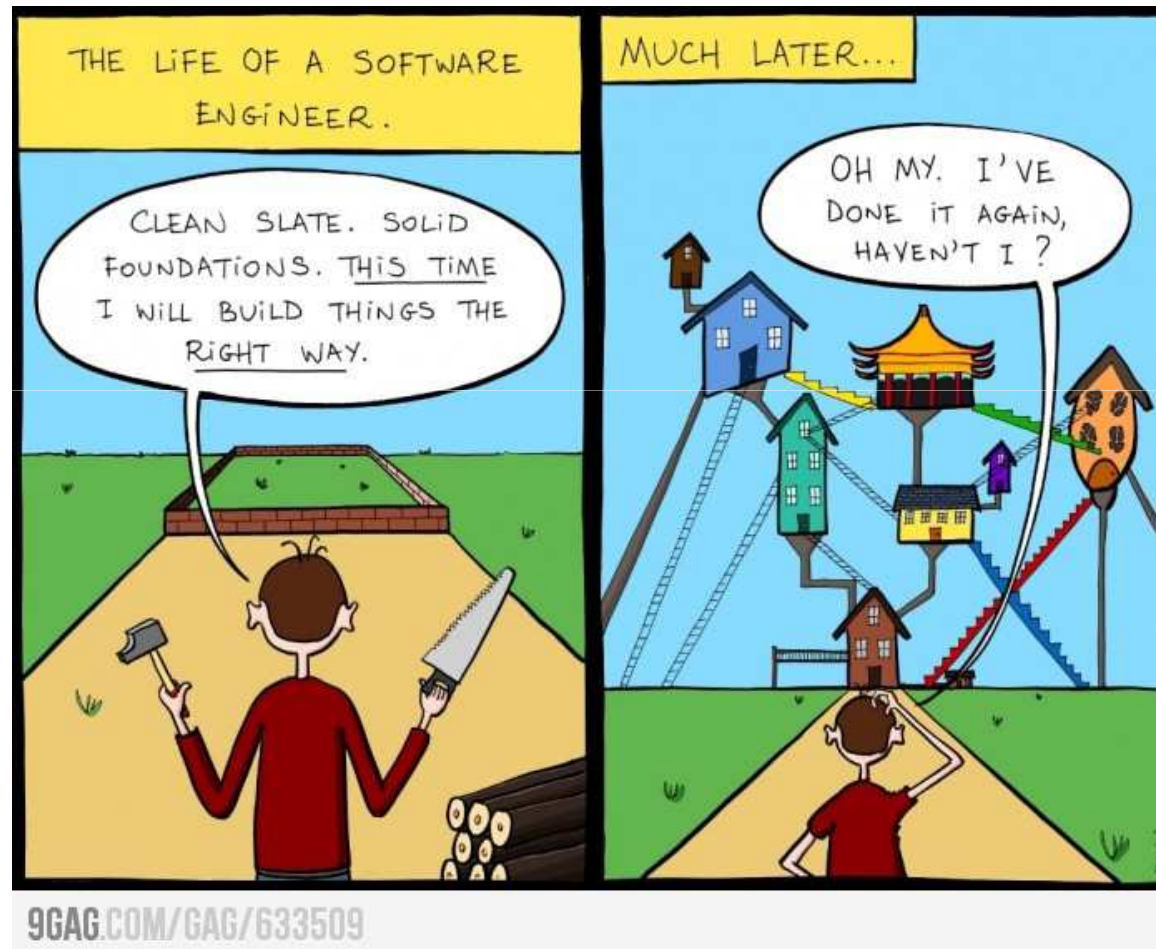
b. Not sure

c. Don't know

The “Rock” Problem



The “Rock” Problem

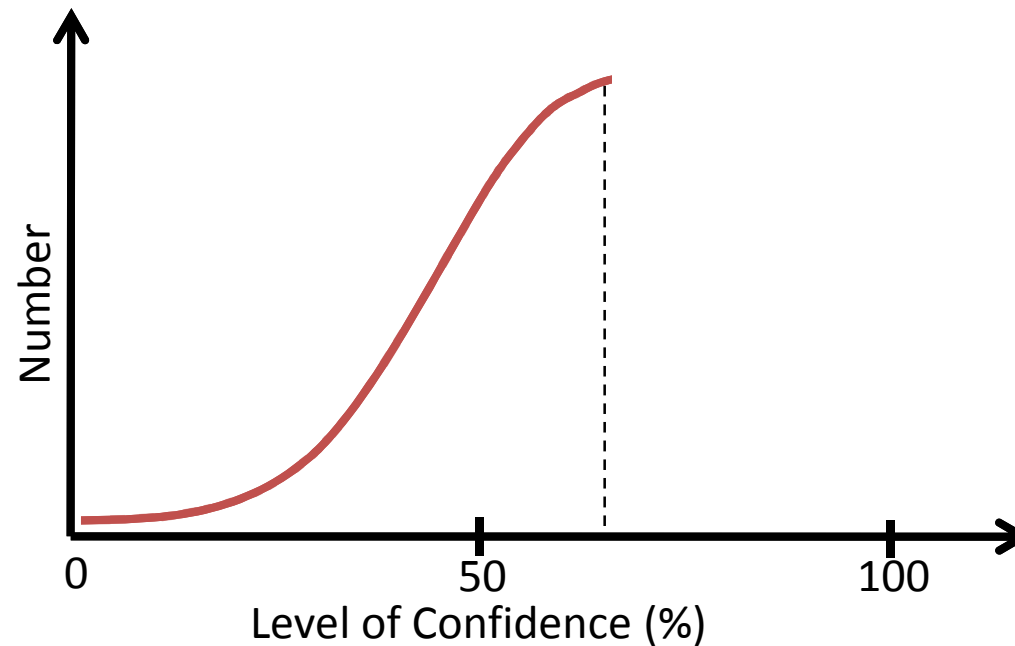


The “Rock” Problem

- The “Rock” Problem gives us a perspective that it is almost impossible to catch and equate the requirement definition to all stakeholders.
- It means: software development will build a software that is never completed at the eyes of stakeholders.
- Software development is never finished. It is a long-life development cycle.

Q. #2

- **How confident** are you that the *software* you built will run properly in accordance with the *requirements*?



Success Rate *)

Based on Chaos Report of Standish Group, @1995

- The Standish Group research shows a staggering 31.1% of projects will be cancelled before they ever get completed.
- Further results indicate 52.7% of projects will overcost 189% of their original cost estimates.
- The average overrun is 222% of the original time estimate.
- The success rate was only 16.2%.

*) Success means “on time, on budget, on spec.”

Success Rate *)

Based on Ambysoft.com Survey on Dec, 2008

- TIME: Which is more important?
 - Delivering on time according to the schedule
 - Delivering when the system is ready to be shipped
 - It does'n matter to me

39,5%

57,5%

3,0%

Success Rate *)

Based on Ambysoft.com Survey on Dec, 2008

- **FUNCTIONALITY: Which is more important?**

- Building the system to the specification
- Meeting the actual needs of stakeholders
- It does'n matter to me

15,5%

83,4%

1,1%

Success Rate *)

Based on Ambysoft.com Survey on Dec, 2008

- QUALITY: Which is more important?
 - Delivering systems on time and on budget
 - Delivering high quality, easy to maintain systems
 - It doesn't matter to me

16,6%

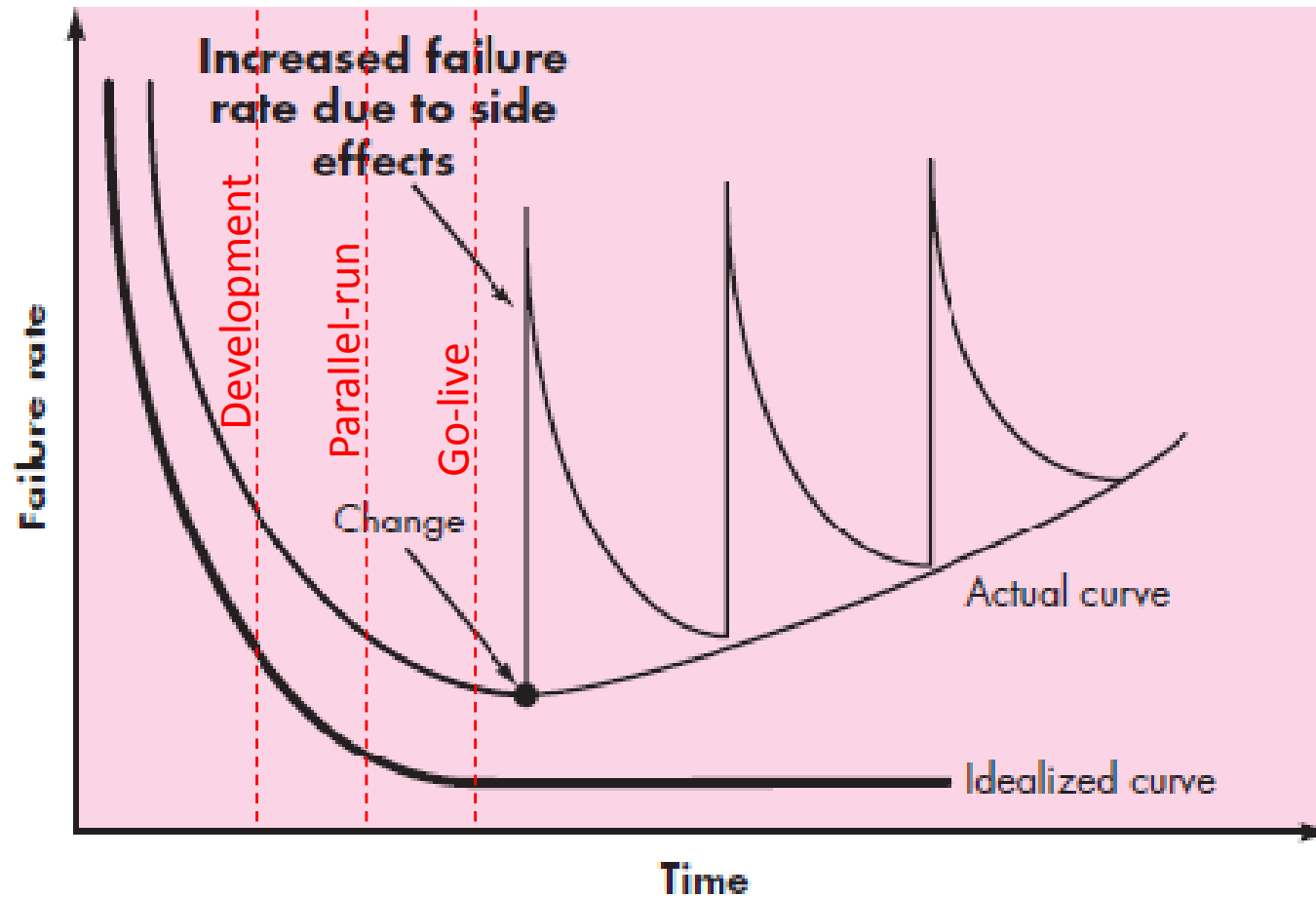
81,9%

1,5%

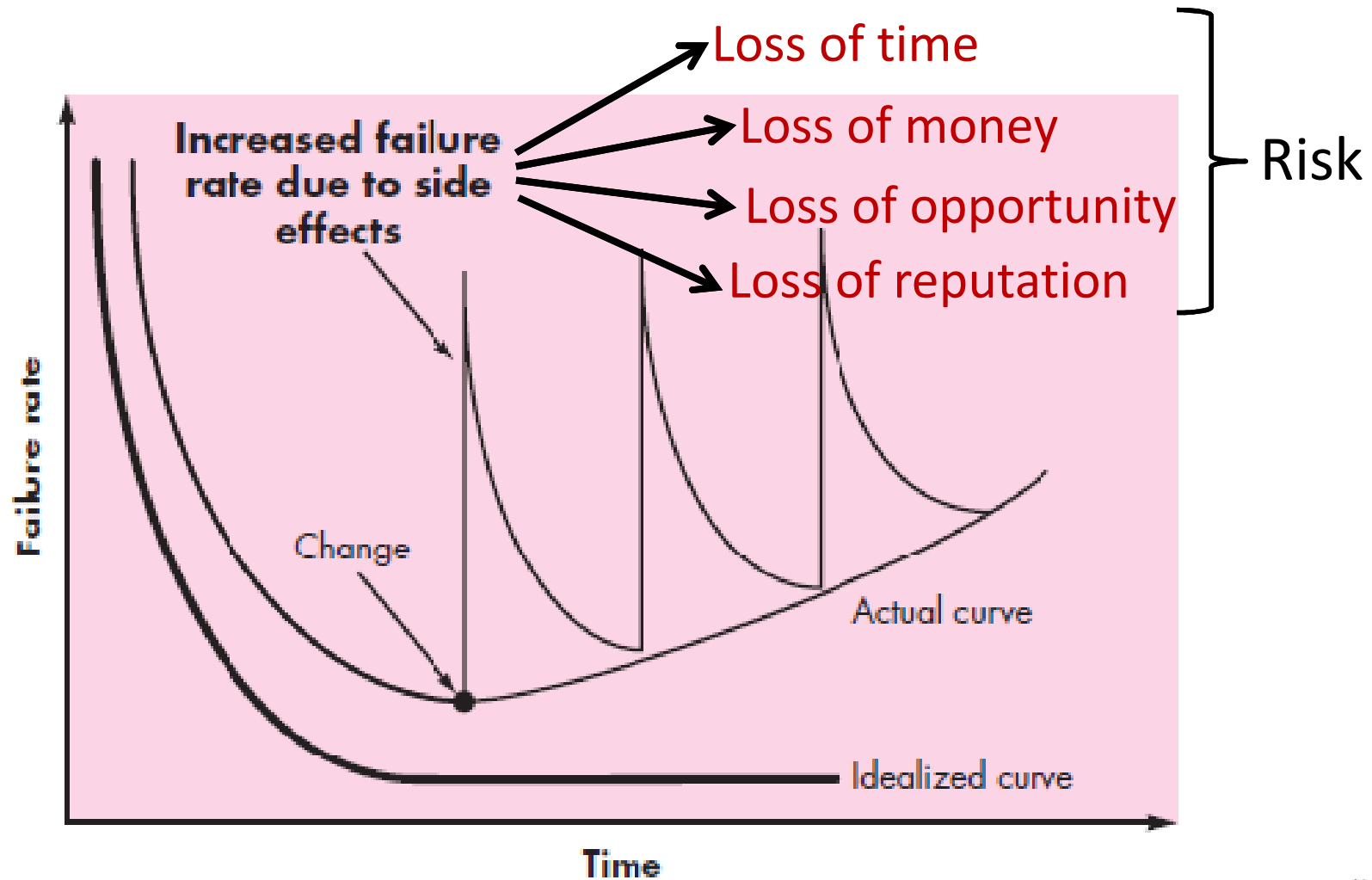
Success Factor

- Very important to meet and match the success factors between all stakeholders in developing system.

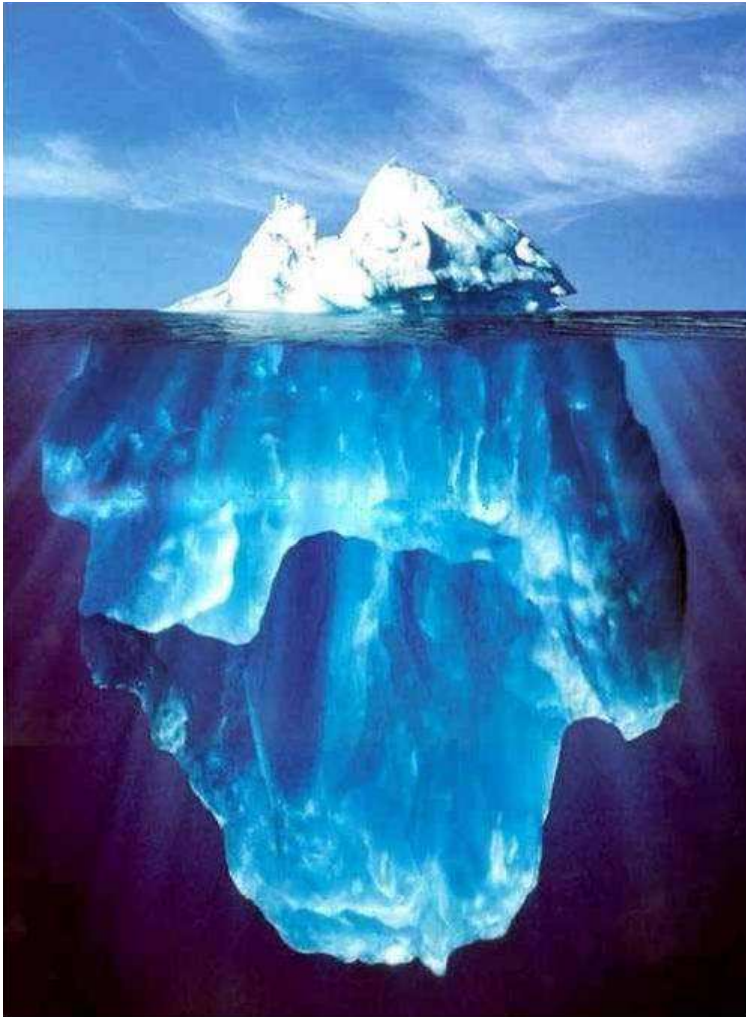
S/W Failure Curve



Generating “Risk”



SOFTWARE ENGINEERING IN PRACTICE #2



Introduction to

Risk Management

#1. Background

#2. Risk and Software Risk

#3. Risk Management

#4. Risk Management in Practice

What is “Risk”?

- Risk is a probability or threat of damage, injury, liability, loss, or any other negative occurrence that is caused by external or internal vulnerabilities, and that may be avoided through pre-emptive action.

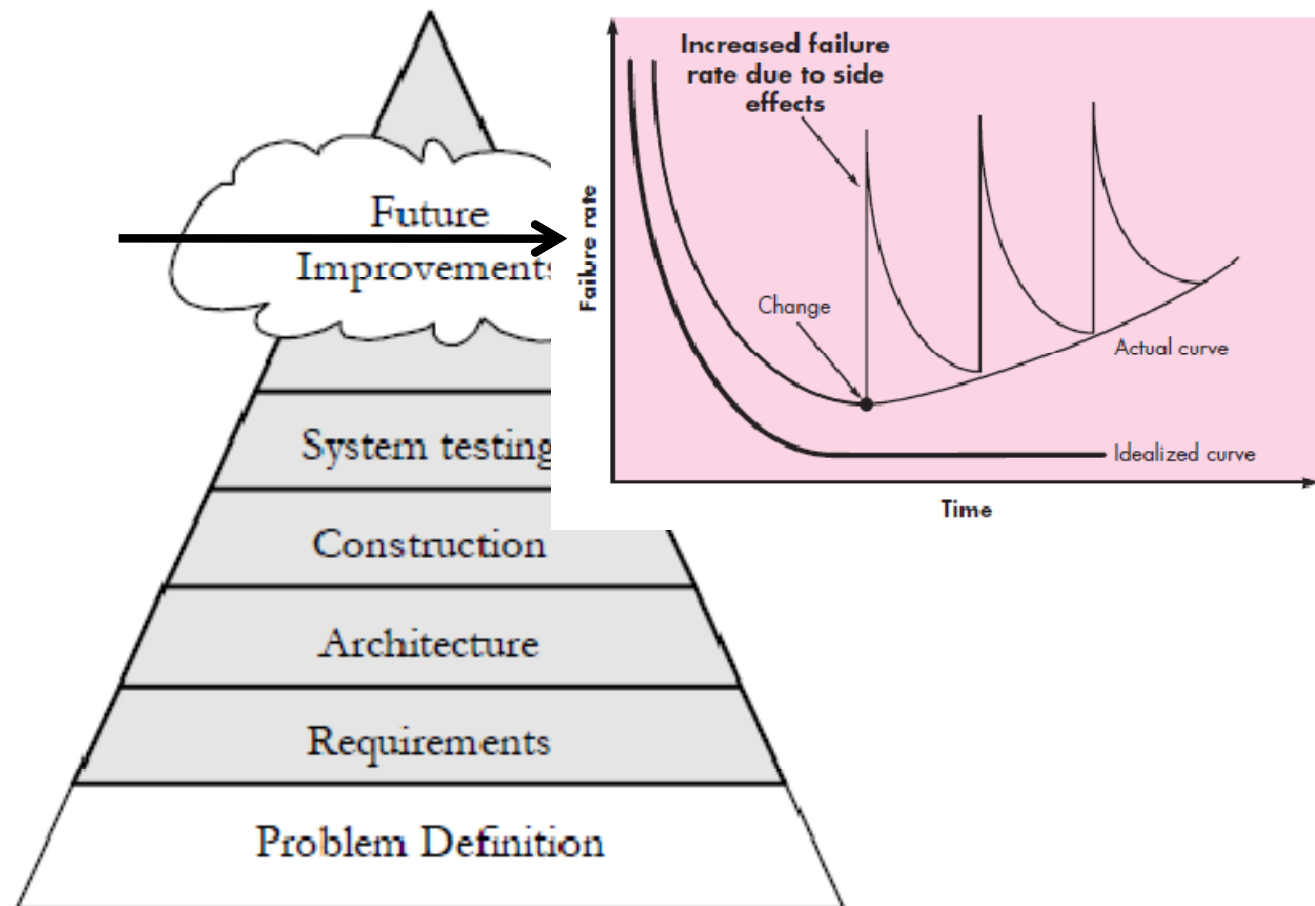
(businessdictionary.com).

More about “Risk”

- Risk concerns future happenings.
 - Yesterday and today are already happened.
 - We are already reaping what was previously sowed by our past actions.
- Risk involves choice, and the uncertainty that choice itself entails.
- Risk involves change
 - By changing our actions today, we can create a better situation tomorrow.

(Robert Charette)

The Foundation



Key #1



Robert N. Charette
Risk management
consultant

Thus paradoxically, risk, like death and taxes, is one of the few certainties of life.

(Robert Charette)

Charette's Concept

Charette's Three Conceptual Underpinnings

- The future is our concern
 - What risks might cause the software project to go awry
- Change is our concern
 - How will changes in any entity connected to the project affect overall success
- Choice is our concern. Means we must grapple with choice.
 - What methods, tools should be used, how many people should be involved, etc?

Risk: Negative or Positive?

- People have a negative connotation when thinking about Risk.
- Risk itself is not really a bad thing.
- Taking risks can even have a positive impact.
 - You may get a discount from a vendor because of past order.
 - A change request from customer allows project six more weeks of development time.
- Positive risks are called *Risk Opportunities*.

Risk: Negative or Positive?

- Let see the interesting video of Mario Teguh regarding the “Risk”.



Software Risks

- Important goals of software projects:
 - Deliver the software to customer at the agreed time
 - Keep overall costs within the budget
 - Deliver software that meets customer's expectations
 - Maintain a happy and well-functioning development team.
 -

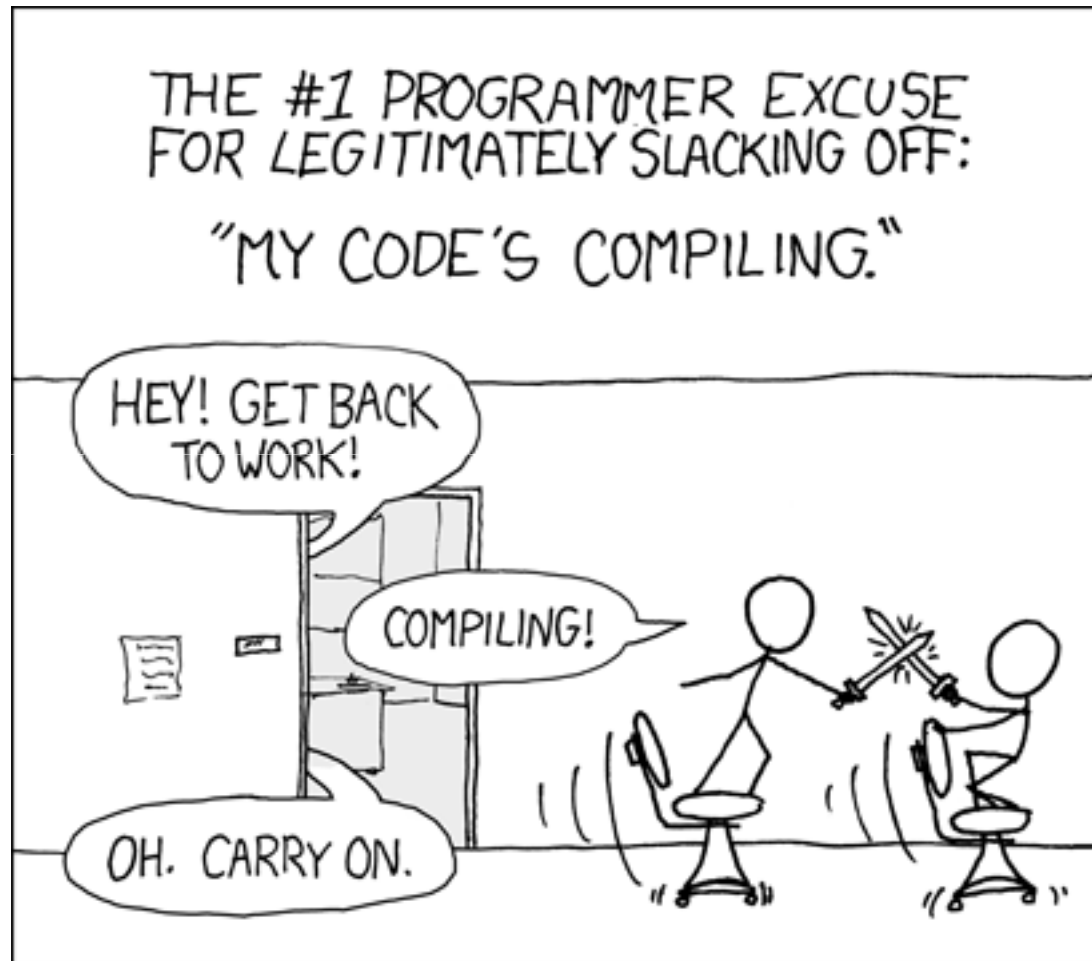
Software Risks

- Some differences of the software:
 - The product is intangible
 - Large software projects are often “one-off” projects.
 - Software processes are variable and organization-specific.
- It is not surprising that some software projects are late, over budget, and behind schedule.

Software Risks

- Any risk that is related to software engineering.
- In software engineering, risks are *rampant*.
- All projects deal with risks that are usually so far off the risk radar.
 - The company might go out of business.
 - An asteroid could crash into the office building
 - Thunder in a rainy day damages all devices, included data storage.
 - A thief steals your production computer. There is no back-up of your codes.
 - Senior programmer might be move to Hawaii.

Software Risk



Software Risk Types

- Risk types of software engineering
 - Product Risk: related to software development
 - Project Risk: related to project management.
- There is another risk that is closed to software engineering.
 - Business Risk: related to organization developing or procuring the software.
 - Pure Risk: related to any bad stuff that nobody likes.

Risk management

- Risk management is concerned with identifying risks and drawing up plans to minimise their effect on a project.
- A risk is a probability that some adverse circumstance will occur
 - Project risks affect schedule or resources;
 - Product risks affect the quality or performance of the software being developed;
 - Business risks affect the organisation developing or procuring the software.

Examples of common project, product, and business risks

Risk	Affects	Description
Staff turnover	Project	Experienced staff will leave the project before it is finished.
Management change	Project	There will be a change of organizational management with different priorities.
Hardware unavailability	Project	Hardware that is essential for the project will not be delivered on schedule.
Requirements change	Project and product	There will be a larger number of changes to the requirements than anticipated.
Specification delays	Project and product	Specifications of essential interfaces are not available on schedule.
Size underestimate	Project and product	The size of the system has been underestimated.
CASE tool underperformance	Product	CASE tools, which support the project, do not perform as anticipated.
Technology change	Business	The underlying technology on which the system is built is superseded by new technology.
Product competition	Business	A competitive product is marketed before the system is completed.

The risk management process

- Risk identification
 - Identify project, product and business risks;
- Risk analysis
 - Assess the likelihood and consequences of these risks;
- Risk planning
 - Draw up plans to avoid or minimise the effects of the risk;
- Risk monitoring
 - Monitor the risks throughout the project;

The risk management process



Risk identification

- May be a team activities or based on the individual project manager's experience.
- A checklist of common risks may be used to identify risks in a project
 - Technology risks.
 - People risks.
 - Organisational risks.
 - Requirements risks.
 - Estimation risks.

Examples of different risk types

Risk type	Possible risks
Technology	The database used in the system cannot process as many transactions per second as expected. (1) Reusable software components contain defects that mean they cannot be reused as planned. (2)
People	It is impossible to recruit staff with the skills required. (3) Key staff are ill and unavailable at critical times. (4) Required training for staff is not available. (5)
Organizational	The organization is restructured so that different management are responsible for the project. (6) Organizational financial problems force reductions in the project budget. (7)
Tools	The code generated by software code generation tools is inefficient. (8) Software tools cannot work together in an integrated way. (9)
Requirements	Changes to requirements that require major design rework are proposed. (10) Customers fail to understand the impact of requirements changes. (11)
Estimation	The time required to develop the software is underestimated. (12) The rate of defect repair is underestimated. (13) The size of the software is underestimated. (14)

Risk analysis

- Assess probability and seriousness of each risk.
- Probability may be very low, low, moderate, high or very high.
- Risk consequences might be catastrophic, serious, tolerable or insignificant.

Risk types and examples

Risk	Probability	Effects
Organizational financial problems force reductions in the project budget (7).	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project (3).	High	Catastrophic
Key staff are ill at critical times in the project (4).	Moderate	Serious
Faults in reusable software components have to be repaired before these components are reused. (2).	Moderate	Serious
Changes to requirements that require major design rework are proposed (10).	Moderate	Serious
The organization is restructured so that different management are responsible for the project (6).	High	Serious
The database used in the system cannot process as many transactions per second as expected (1).	Moderate	Serious

Risk types and examples

Risk	Probability	Effects
The time required to develop the software is underestimated (12).	High	Serious
Software tools cannot be integrated (9).	High	Tolerable
Customers fail to understand the impact of requirements changes (11).	Moderate	Tolerable
Required training for staff is not available (5).	Moderate	Tolerable
The rate of defect repair is underestimated (13).	Moderate	Tolerable
The size of the software is underestimated (14).	High	Tolerable
Code generated by code generation tools is inefficient (8).	Moderate	Insignificant

Risk planning

- Consider each risk and develop a strategy to manage that risk.
- Avoidance strategies
 - The probability that the risk will arise is reduced;
- Minimisation strategies
 - The impact of the risk on the project or product will be reduced;
- Contingency plans
 - If the risk arises, contingency plans are plans to deal with that risk;

Strategies to help manage risk

Risk	Strategy
Organizational financial problems	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business and presenting reasons why cuts to the project budget would not be cost-effective.
Recruitment problems	Alert customer to potential difficulties and the possibility of delays; investigate buying-in components.
Staff illness	Reorganize team so that there is more overlap of work and people therefore understand each other's jobs.
Defective components	Replace potentially defective components with bought-in components of known reliability.
Requirements changes	Derive traceability information to assess requirements change impact; maximize information hiding in the design.

Strategies to help manage risk

Risk	Strategy
Organizational restructuring	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business.
Database performance	Investigate the possibility of buying a higher-performance database.
Underestimated development time	Investigate buying-in components; investigate use of a program generator.

Risk monitoring

- Assess each identified risks regularly to decide whether or not it is becoming less or more probable.
- Also assess whether the effects of the risk have changed.
- Each key risk should be discussed at management progress meetings.

Risk indicators

Risk type	Potential indicators
Technology	Late delivery of hardware or support software; many reported technology problems.
People	Poor staff morale; poor relationships amongst team members; high staff turnover.
Organizational	Organizational gossip; lack of action by senior management.
Tools	Reluctance by team members to use tools; complaints about CASE tools; demands for higher-powered workstations.
Requirements	Many requirements change requests; customer complaints.
Estimation	Failure to meet agreed schedule; failure to clear reported defects.

Key points

- Good project management is essential if software engineering projects are to be developed on schedule and within budget.
- Software management is distinct from other engineering management. Software is intangible. Projects may be novel or innovative with no body of experience to guide their management. Software processes are not as mature as traditional engineering processes.
- Risk management is now recognized as one of the most important project management tasks.
- Risk management involves identifying and assessing project risks to establish the probability that they will occur and the consequences for the project if that risk does arise. You should make plans to avoid, manage or deal with likely risks if or when they arise.