

05: Managing Risks and Uncertainty

Software Project Management
Philippe Kruchten

Jan. 2014

Copyright © 2005-14 by KESL

1

Module Outline

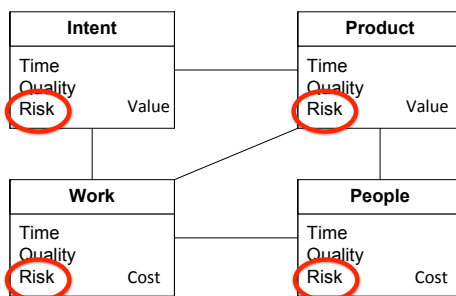
- Failure & success
- Uncertainty & risks
- Risk exposure
- Sources of risk in software projects
- Basic risk management strategies
- Iterative lifecycle, as risk mitigation
- Standards & tools

Jan. 2014

Copyright © 2005-14 by KESL

2

Risk and uncertainty



Jan. 2014

Copyright © 2005-14 by KESL

3

Success

- Success is meeting the entire set of all requirements and constraints held as project expectations by those in power. (RUP)
- Success vs. failure
 - degrees of success or failure

Jan. 2014

Copyright © 2005-14 by KESL

4

Uncertainty

- A state of the project that may have many different outcomes, answers, values... currently unknown
- Each outcome may have a probability of occurrence
 - There is a 60% chance that the number of users will be more than 1,000, and a 10% chance that we'll have more than 10,000, and about 0% chance that we'll have more than 1 billion
- Software projects tend to have more more uncertainties than other type of projects.

Jan. 2014

Copyright © 2005-14 by KESL

5

- Attendre d'en savoir assez pour agir en toute lumière, c'est se condamner à l'inaction.
Waiting to know enough to act in full light is to condemn oneself to permanent inaction.

Jean Rostand, French biologist, 1967

Jan. 2014

Copyright © 2005-14 by KESL

6

Risk Defined

- Risk: the possibility of a loss or injury (Webster)
- Risk is the possibility of suffering loss (SEI)
- A risk is an ongoing or impending concern that has a significant probability of adversely affecting the success of major milestones (RUP)

Jan. 2014

Copyright © 2005-14 by KESL

7

Risk Defined (cont.)

- Risk: The likelihood of an event, hazard, threat, or situation occurring and its undesirable consequences; a potential problem (IEEE Std 1540).
- A risk is whatever may stand in our way to success, and is currently unknown or uncertain.

Jan. 2014

Copyright © 2005-14 by KESL

8

Success and Failure: Chaos Report 2003

- Success: 34% (was 16% in 1994)
 - On time, on budget, with expected functionality
- Failure: 16% (was 31% in 1994)
 - Cancelled somewhere along the lifecycle, etc...
- “Challenged” projects: 51%
 - cost and schedule overrun, reduced functionality,
 - “restarts” is often the cause

Source: Standish Group

Jan. 2014

Copyright © 2005-14 by KESL

9

Risk Exposure

$$\text{Risk exposure (R)} = \text{Probability (R)} \times \text{Impact (R)}$$

Jan. 2014

Copyright © 2005-14 by KESL

10

Example

Risk	Probability	Impact	Exposure
Late by a week	10%	\$187,000	\$18,700
Show-stopper bug in User-Interface	30%	\$200,000	\$60,000
John leaves	2%	\$500,000	\$10,000
... etc.
Total exposure:			\$88,700

Jan. 2014

Copyright © 2005-14 by KESL

11

Risk Management

"The readiness is all."
- Hamlet V.ii:215

- Risk Assessment
 - Risk identification
 - Risk analysis
 - Risk prioritization
- Risk Control
 - Risk monitoring (“top 10 risks”)
 - Risk planning: avoidance, transfer, reduction
 - Risk resolution
 - prototypes, simulations, analyses, expertise, ...

Jan. 2014

Copyright © 2005-14 by KESL

12

Direct or Indirect Risks

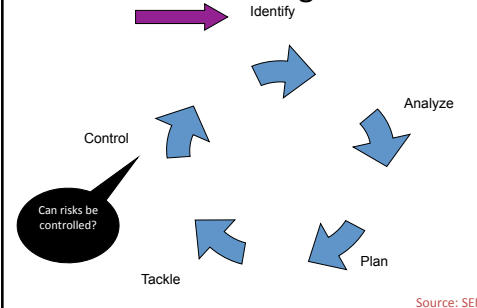
- Direct risk: project has a large degree of control
 - Examples:
 - Wrong workload estimates
 - Poor quality
- Indirect risk: project has little or no degree of control
 - Examples:
 - Loss of personnel
 - New competitor on the market

Jan. 2014

Copyright © 2005-14 by KESL

13

Risk Management



Jan. 2014

Copyright © 2005-14 by KESL

14

Risk Identification

- Produce and maintain a list of risks specific to the project
- Start from know lists and taxonomies:
 - SEI, Caper Jones's book
 - Brainstorming sessions:
 - scenarios about what can go wrong
 - no finger pointing, don't shoot the messenger!
 - no attempt to fix anything

Jan. 2014

Copyright © 2005-14 by KESL

15

Examples of "Unsatisfactory Outcome"

- Budget and cost overrun
- Product with the wrong functionality or incomplete
- User-interface shortfalls
- Performance shortfalls
- Lack of reliability
- Software impossible to evolve or maintain
- Shortcomings: Management or Technical?

Jan. 2014

Copyright © 2005-14 by KESL

16

Risk Analysis

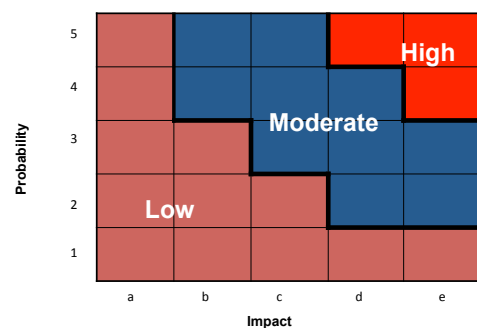
- What is affected, what is the cause?
- One or many risks?
- Determine Exposure
- Quantitative (\$\$\$) ? or qualitative?
 - make it simple and effective
 - Impact: low, medium, high (or 1 to 5)
 - Probability: a percentage (or 1 to 5)

Jan. 2014

Copyright © 2005-14 by KESL

17

Alternative: Using a simple matrix



Copyright © 2005-14 by KESL

18

Prioritize Risks

- You can't tackle everything!
- Put aside the ones you will have to live with
- Bring out a list of the "top ten"
- Make it visible to all involved (external stakeholders and upper management included)

Jan. 2014

Copyright © 2005-14 by KESL

19

Example of a Starting List

1. Goals not defined properly
2. Goals defined, but changes out of control
3. No proper planning
4. No leadership
5. OK plan, but lack resources to match
6. No contingency in plan
7. Expectations not properly managed
8. Progress not monitored properly
9. No reporting
10. Naïve view of fixing problems (e.g.: "add more resources")

Jan. 2014

Copyright © 2005-14 by KESL

20

Example of Actual Risks

Risk	Prob	Imp.	Exp	Actions	Warning signs
Burnout	3	3	9		
Scope creep	1	3	3		
Bad mngmt	2	3	6		
People leave	3	3	9		
Late subcontract	2	2	4		
Charles leaves	1	3	3		
... etc.					

Copyright © 2005-14 by KESL

Source: O'Connell 2001

21

Risk Treatment

What do we do with the risks?

- **Risk acceptance:** live with it!
- **Risk avoidance:** reorganize to eliminate the risk
- **Risk contingency:** Develop a "plan B"
- **Risk mitigation:** reduce the probability or the impact
- **Risk transfer:** push the risk onto another stakeholder

Jan. 2014

Copyright © 2005-14 by KESL

22

Example of actual risks (continued)

Risk	P	I	E	Actions	Warning signs
Burnout	3	3	9	??	absences
Scope Creep	1	3	3		
Bad mngmt	2	3	6		delays
People leave	3	3	9	incentives	resumes
Late subcontrac	2	2	4	Xfer >John	delays
Charles leaves	1	3	3	incentive	resignation
Infeasibility	1	3	3		

Copyright © 2005-14 by KESL

23

Risk Management: A Continuous Activity

- New risks pop up
- Impact & likelihood vary
- Update, then sort the list
 - tip: Use an Excel spreadsheet
- Make risk a global concern
 - Publish weekly "top ten"
 - Do something about them (where possible)
- Encourage risk identification
 - No "instant punishment" on the discoverer

Jan. 2014

Copyright © 2005-14 by KESL

24

Risk Management: State of Practice

- 20% of organization claim doing it
- 10% are really doing more than risk identification and then problem management
- Issue: is the payoff worth the effort?

Jan. 2014

Copyright © 2005-14 by KESL

25

Standard on Risk Management

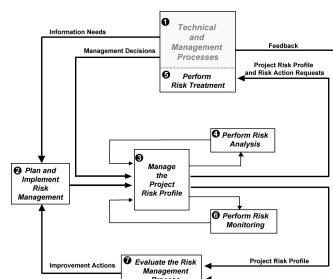
- IEEE standard 1540: 2001
Leads to a rather heavy processes
- ISO guide 73:2009 Risk Management-Vocabulary
- ISO 31000:2009 Principles and Guidelines for Risk Management

Jan. 2014

Copyright © 2005-14 by KESL

26

IEEE 1540



Jan. 2014

Copyright © 2005-14 by KESL

27

Risk Management Tools

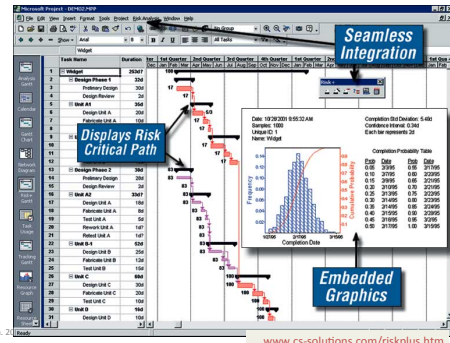
- Spreadsheet (e.g., Excel)
- CS Solutions has a MS Project add-on for schedule-related risks
– www.cs-solutions.com/riskplus.htm

Jan. 2014

Copyright © 2005-14 by KESL

28

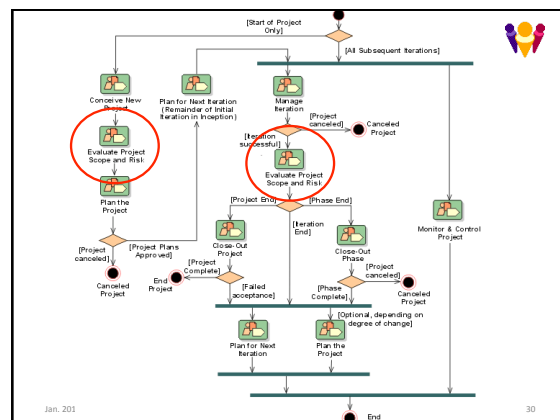
CS Solutions: Risk+ 2.0



Jan. 2014

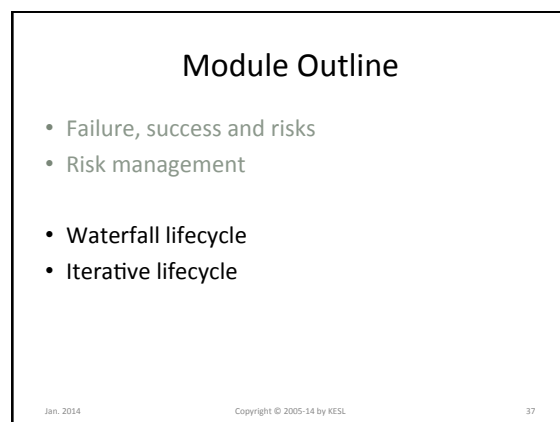
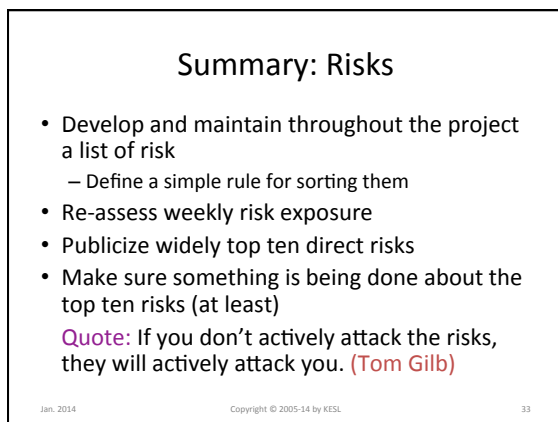
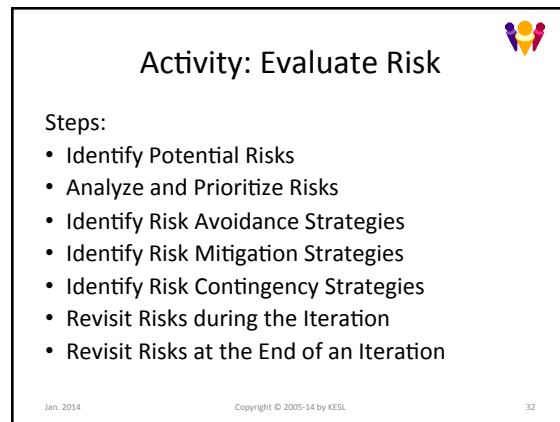
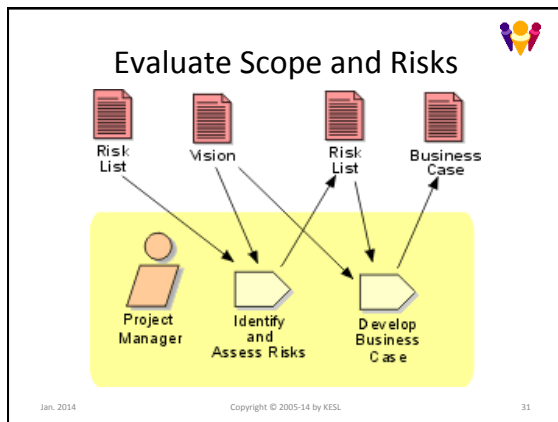
www.cs-solutions.com/riskplus.htm

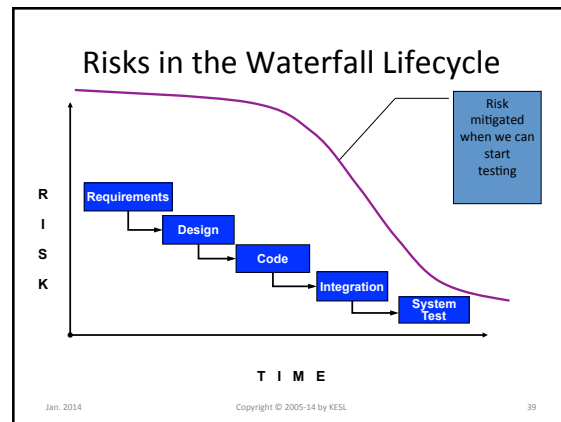
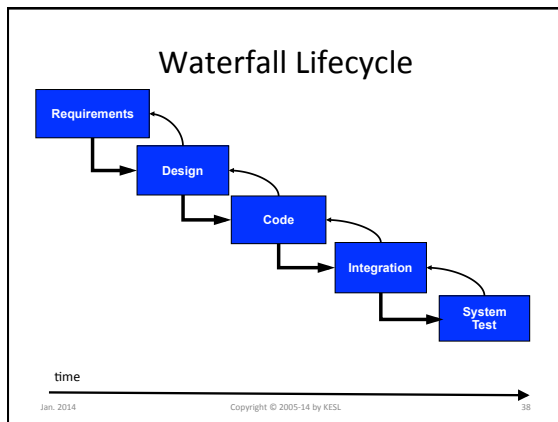
29



Jan. 2014

30





Pros and Cons of the Waterfall:

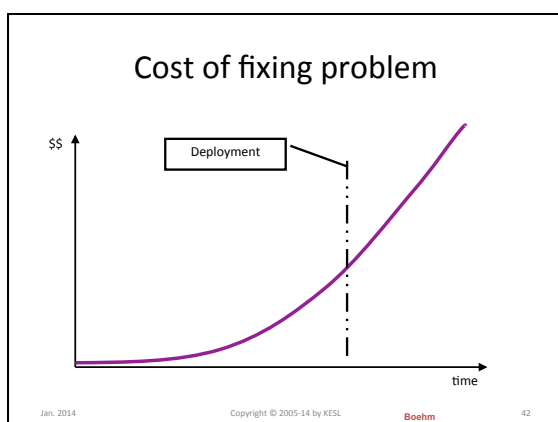
- Pros
 - Used in other disciplines
 - Well-established
 - Rigorous and rational
 - Minimize waste
- Cons
 - Needs very mature organizations and process
 - Relies on analysis of intermediate artifacts
 - Assumes stable environment and stable requirements
 - Leaves little room for learning
 - and its “feedback loop” too limited (or too long)

Jan. 2014 Copyright © 2005-14 by KESL 40

More Cons

- An initial design will likely be flawed with respect to its key requirements
- 50% of time and effort spent in Final integration and test
- Late discovery of design defects results in costly over-runs and/or project cancellation
- The time and money spent implementing a faulty design are not recoverable

Jan. 2014 Copyright © 2005-14 by KESL 41



Limits of the Waterfall Lifecycle

Can't do well:

- Projects with short time-frame
- Projects with lots of unknowns
 - requirements
 - technology
- Projects with lots of “new” stuff
 - People, language, domain, etc..
- Large project....

Jan. 2014 Copyright © 2005-14 by KESL 43

Iterative Development

Quote: “Plan to throw one away; you will anyhow.”

(Fred Brooks, the Mythical Man Month, 1975)

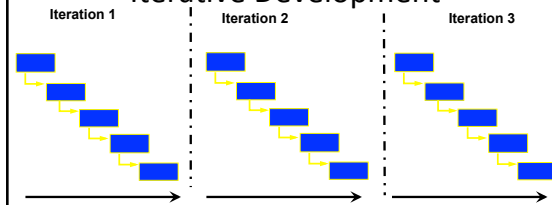
- Design a little, build a little, test a little
- Supported by efficient software development environments
- “Several waterfall one after another” ???

Jan. 2014

Copyright © 2005-14 by KESL

44

Iterative Development



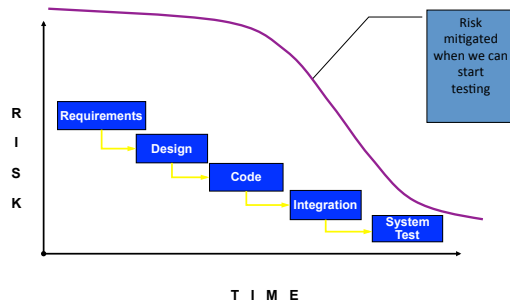
- Each iteration produces an executable release
- Each iteration includes integration and test
- Earliest iterations address greatest risks

Jan. 2014

Copyright © 2005-14 by KESL

45

Risks in the Waterfall Lifecycle

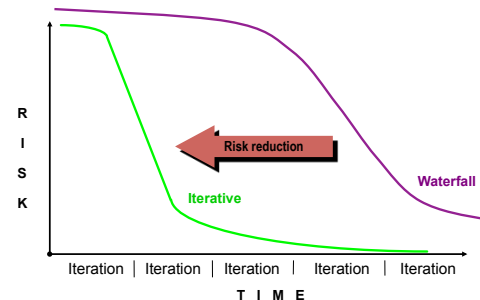


Jan. 2014

Copyright © 2005-14 by KESL

46

Risks in the Iterative Lifecycle

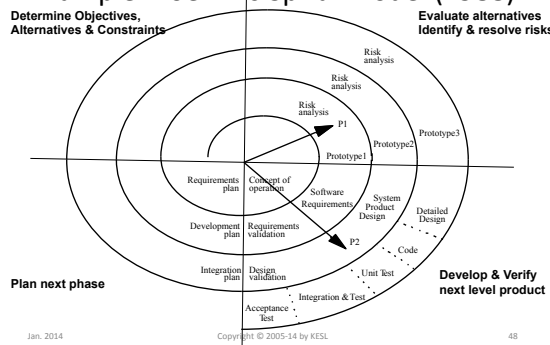


Jan. 2014

Copyright © 2005-14 by KESL

47

Example: Boehm's Spiral Model (1988)

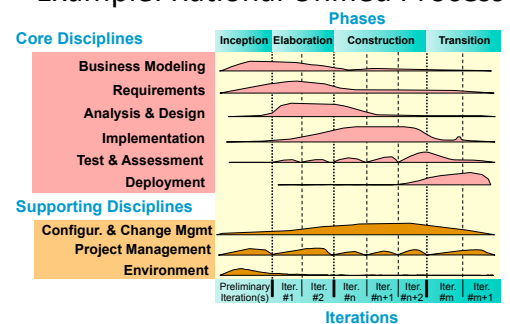


Jan. 2014

Copyright © 2005-14 by KESL

48

Example: Rational Unified Process



Jan. 2014

Copyright © 2005-14 by KESL

Source: IBM

49

More Examples

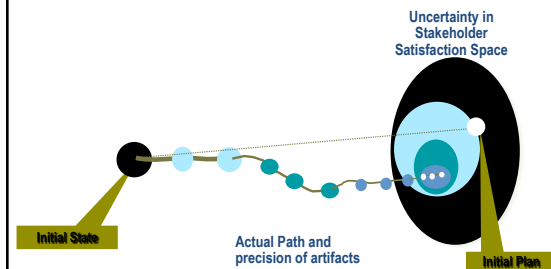
- DSDM
- All Agile Methods
 - XP
 - Scrum
 - Crystal
 - Lean Development
- Iterative development acknowledged by major standards
 - Upcoming PMBOK extension for software

Jan. 2014

Copyright © 2005-14 by KESL

50

Iterations allow tactical adjustments



Jan. 2014

Copyright © 2005-14 by KESL

Source: W. Royce, IBM

Impact on Software Project Management

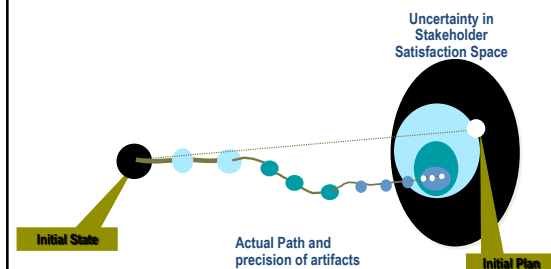
- More work for the manager
 - more planning
- Best way to do active Risk Management
- More predictability in outcome
 - Including earlier opportunity for cancellation
- Greater quality (?)
- Tighter feedback loop
 - requirements, design, process, people,

Jan. 2014

Copyright © 2005-14 by KESL

52

Iterations allow tactical adjustments



Jan. 2014

Copyright © 2005-14 by KESL

Source: W. Royce, IBM

53

Summary: Iterative Lifecycle

- Traditional **Waterfall Lifecycle** not too successful with software-intensive projects
- **Iterative Development** emerged as an alternative in the late 1980's
 - facilitates risk management
 - provides tighter, shorter feedback loop
 - allows for tactical adjustments:
 - requirements
 - engineering (design, technologies, tool support)
 - process, people, budget, etc...

Jan. 2014

Copyright © 2005-14 by KESL

54

Summary: Risks

- Develop and maintain throughout the project a list of risk
 - Define a simple rule for sorting them
 - Re-assess weekly your risk exposure
 - Expose widely the top ten direct risks
 - Make sure something is being done about the top ten risks (at least)
- Quote:** If you don't actively attack the risks, they will actively attack you. (Tom Gilb)

Jan. 2014

Copyright © 2005-14 by KESL

57