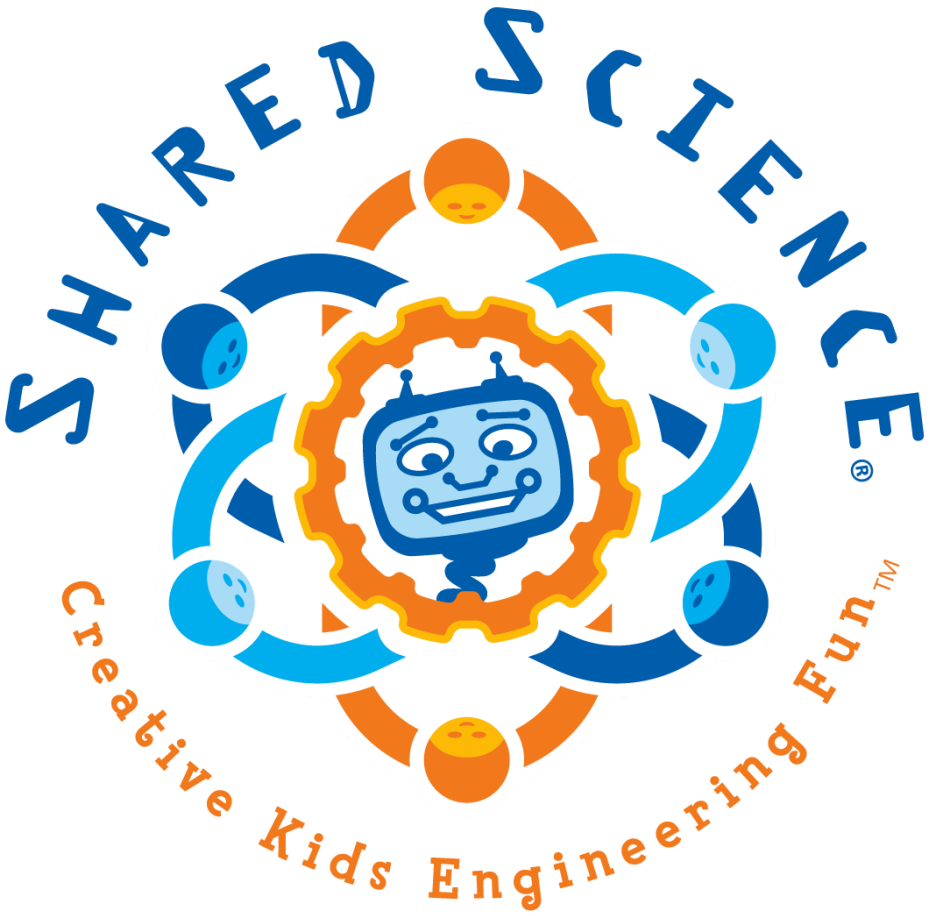
Requirements Engineering: Change Management & Risk Management

CECS 590

Looking for volunteers



https://www.facebook.com/SharedScience

Recap time!

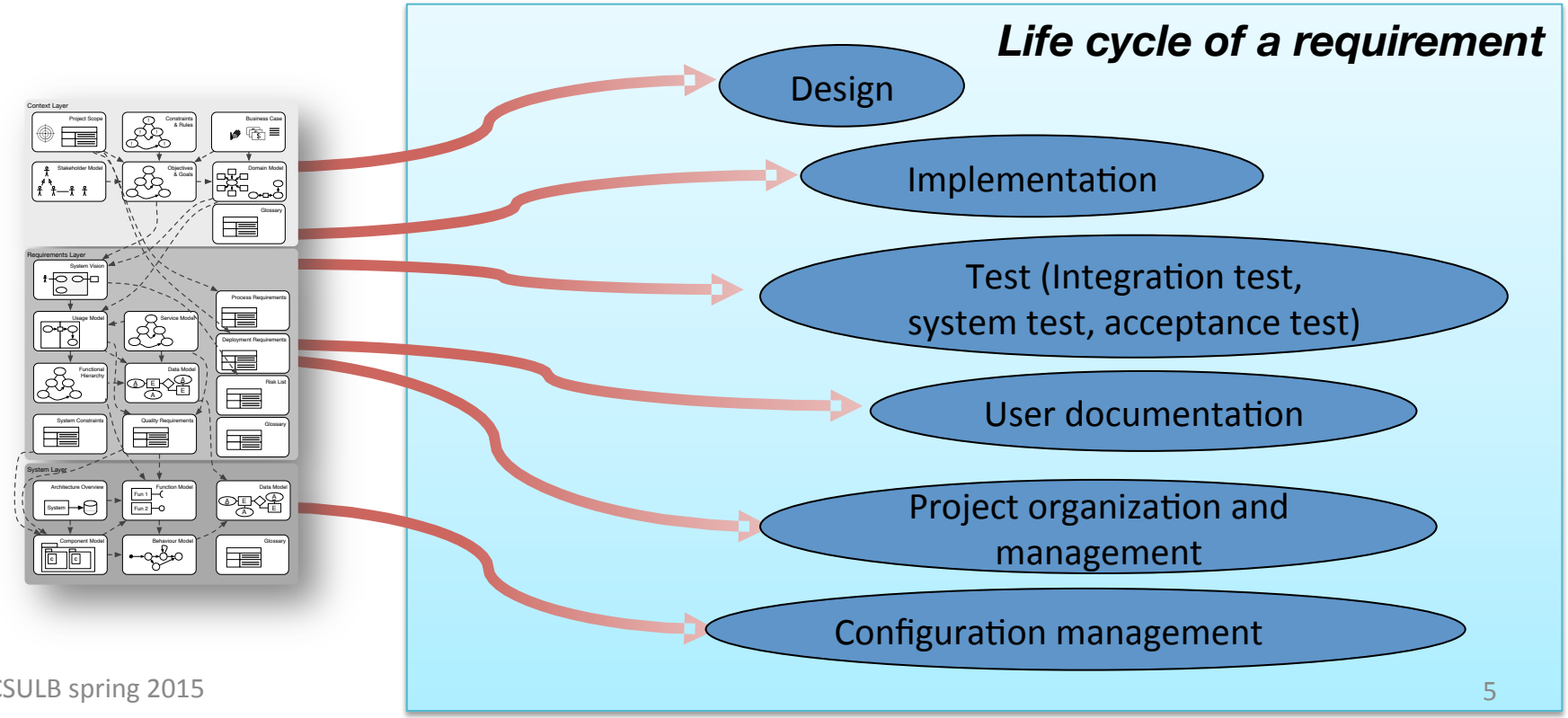
- Quality assurance
  - What are quality defects?
  - How is QA defined?
  - What is constructive QA?
  - What is analytical QA?

Requirements Engineering – Outline

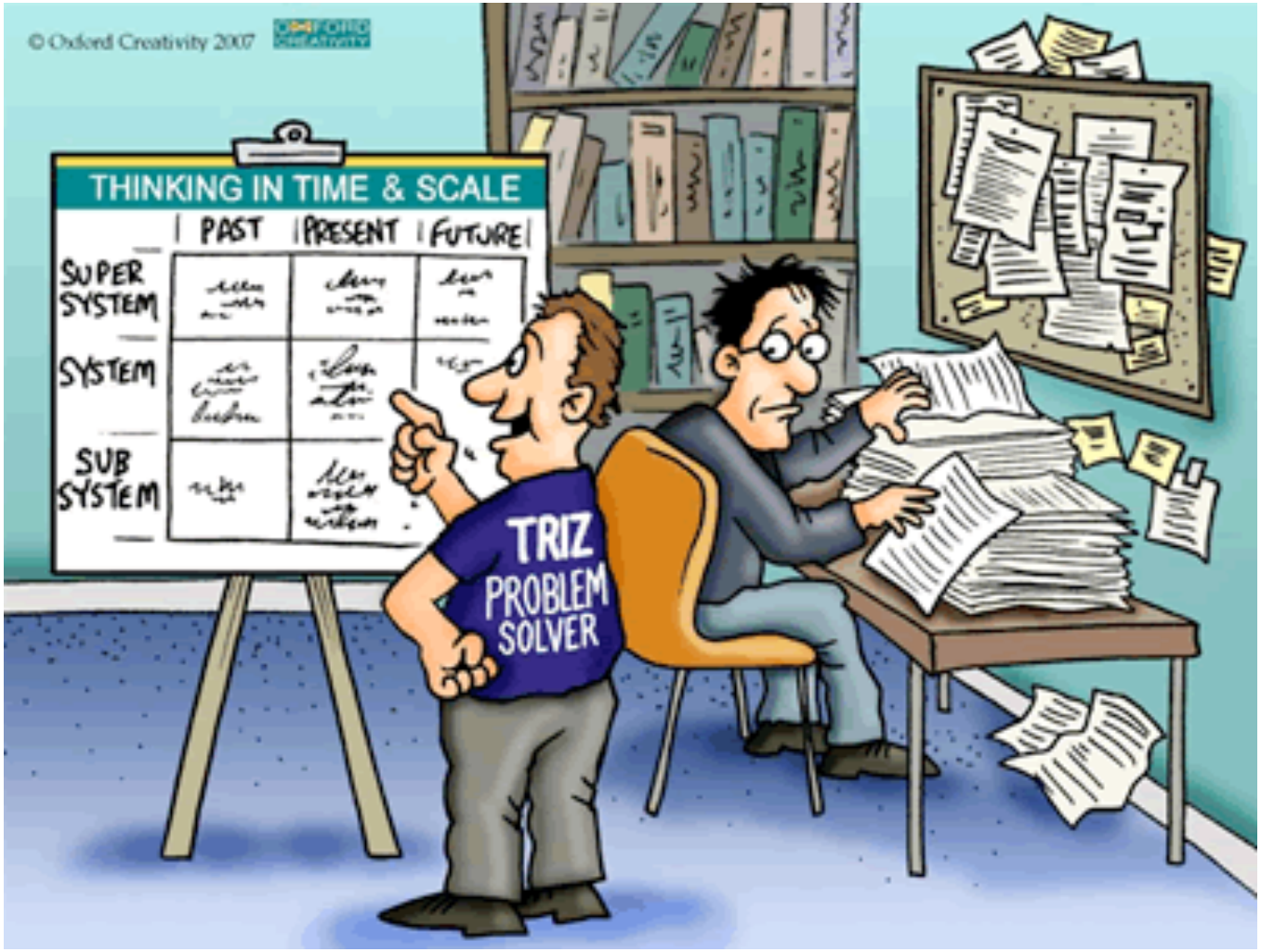
- WHY do we need Requirements Engineering and what is it?
- Principles: Definitions, process, roles, problem/solution view, artifact orientation
- System Models: Decomposition and abstraction, system views
- Frameworks: What reference structures can I use for requirements?
- Business Case Analysis: Why are we building this system?
- Stakeholders: Who are the people to talk to about requirements?
- Goals and Constraints: What are the major objectives for the system?
- System Vision: What exactly do we want to achieve?
- Domain Models: What are the surrounding systems ours interacts with?
- Usage Models: How will the system interact with the user?
- Software quality models: How to determine the quality characteristics?
- Quality requirements: How to specify which qualities need to be met?
- Process requirements: How to specify constraints for development?
- Towards a system specification: How to hand over to design?
- Quality assurance: How to ensure that RE is done in a good way?
- **Change management: How to evolve requirements?**

Connection to Req Mngmt

QA because results of RE are needed for



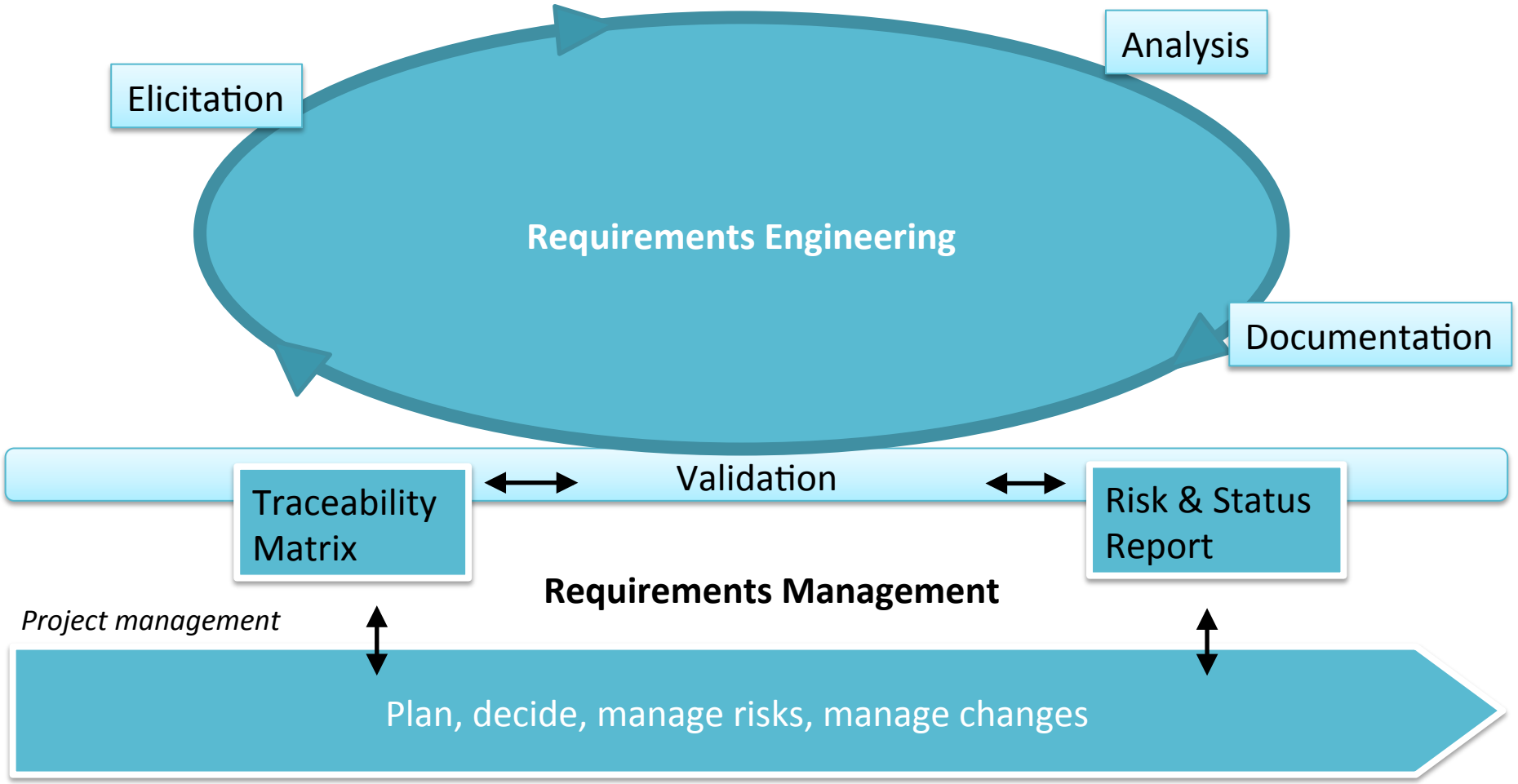
Change & Risk Management



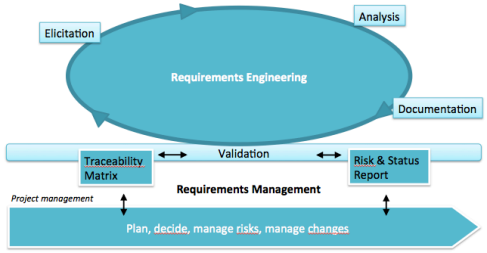
Overview: Change & Risk Management

- Recap: Requirements Management
- Change Management
- Risk Management

Recap: RE in the process



Requirements Management: Tasks



- Rationale Management and Traceability
  - Rationale for requirements
  - Relation between content items
- Management of the requirements
  - Structuring, documentation and archiving
  - Attribution
- Interdependency with other management tasks
  - Validation and Verification
  - Change management and Impact Analysis
  - Version management
  - Configuration management
  - Claim management
  - Support for distributed RE
  - Tool support for RM

Overview: Change & Risk Management

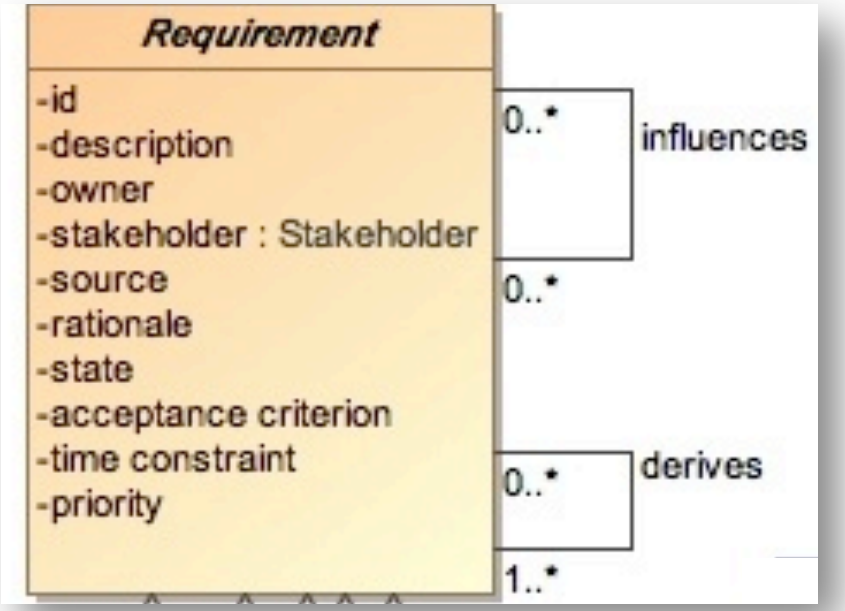
- Recap: Requirements Management
- Change Management
- Risk Management

Change Management – why?

K Rayker, stock.xchng

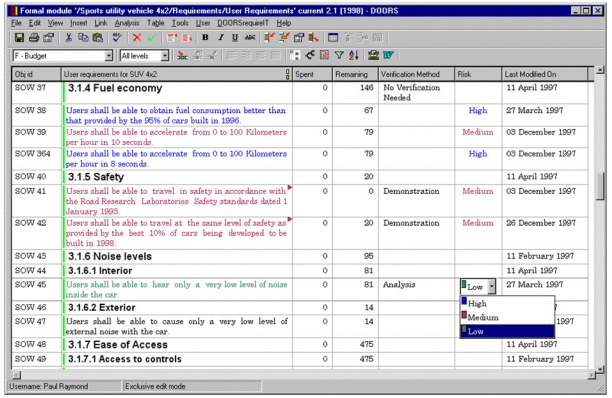
Recap: Attibutes for requirements

- ID
- Description
- Owner
- Stakeholder
- Source
- Rationale
- State
- Accceptance Criterion
- Time Constraint
- Priority



→ Make sure all of these are updated in the change process

Managing requirements



- Big projects may have thousands of requirements
- For management, the standard process is
  1. All are submitted to a database
  2. And managed via their attributes
  3. For the respective milestone, requirements are decided upon („Freeze“ / „Baseline“)
  4. And often gathered in a document for official approval

→After this official approval, further changes have to adhere to a standardized change process

Change Management – how?

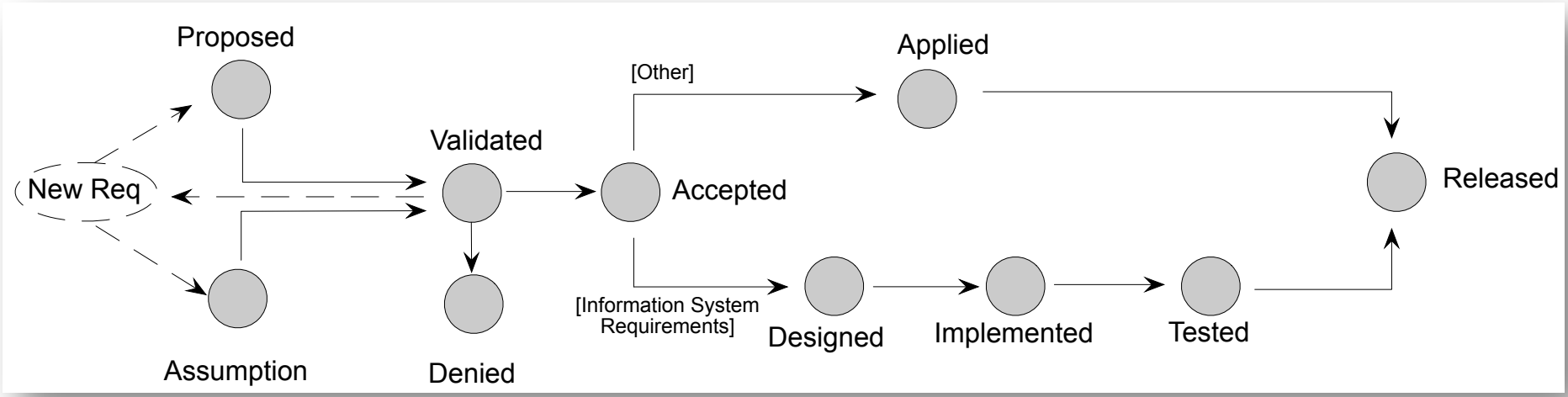
Four steps

- Systematic elicitation of Change Requests
- Decision on priority and costs
- Decision on implementation
- Implementation of changes

Important related **tasks**:

- Impact analysis: What consequences do changes have?
- Version management**:** Which versions of requirements exist?
- Configuration management**:** Which requirements form consistent baselines?
- Defect management**:** How to deal with defects and their correction?

Change Management – how?



K Rayker, stock.xchng

Overview: Change & Risk Management

- Recap: Requirements Management
- Change Management
- Risk Management

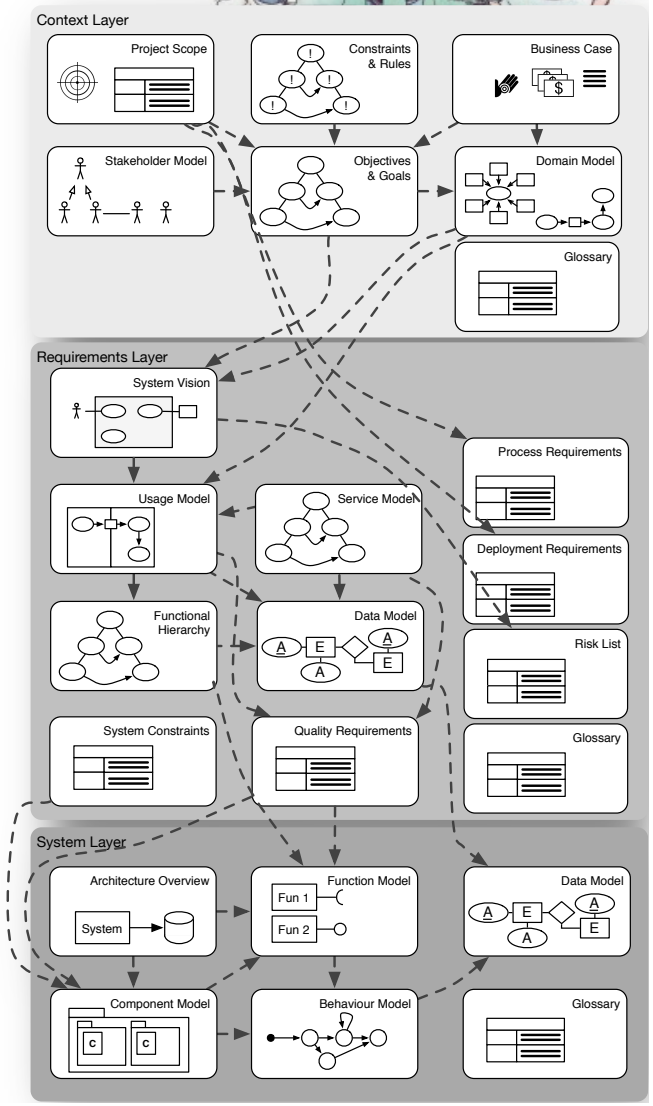
Risk Management

Risks that are related to requirements have to be identified, assessed and analyzed, so that action can be taken early on.


RISK MANAGEMENT CYCLE
Risks Identified
•Scope defined
•SWOT
•Checklists/tools
•Experience
•Interviews
Risks Prioritized
•Impact
•Probability
Risks Grouped By Categories
•Financial
•Schedule
•Quality
•Scope
Risk Analysis & Response
•Qualitative
•Quantitative
•Decision Trees
•Contingency Time and Costs
Risks Response
•Mitigate: Enhance Opportunities and Decrease Threats
•Log/Plan actions & alternatives
Risk Monitoring
•Know risk triggers
•Determine if assumptions are still valid
•Corrective actions

Exercise: Avoid disasters

1. *What is the problem in the following situations?*
2. *How could it have been avoided? (in 3, 5, and 7)*



The top 10 IT disasters of all time

1. Faulty Soviet early warning system nearly causes WWIII (1983)
The threat of computers purposefully starting World War III is still the stuff of science fiction, but accidental software glitches have brought us worryingly close in the past. Although there are numerous alleged events of this ilk, the secrecy around military systems makes it hard to sort the urban myths from the real incidents.

However, one example that is well recorded happened back in 1983, and was the direct result of a software bug in the Soviet early warning system. The Russians' system told them that the US had launched five ballistic missiles. However, the duty officer for the system, one Lt Col Stanislav Petrov, claims he had a "...funny feeling in my gut", and reasoned if the US was really attacking they would launch more than five missiles.

The trigger for the near apocalyptic disaster was traced to a fault in software that was supposed to filter out false missile detections caused by satellites picking up sunlight reflections off cloud-tops.

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

2. The AT&T network collapse (1990)
In 1990, 75 million phone calls across the US went unanswered after a single switch at one of AT&T's 114 switching centres suffered a minor mechanical problem, which shut down the centre. When the centre came back up soon afterwards, it sent a message to other centres, which in turn caused them to trip and shut down and reset.

The culprit turned out to be an error in a single line of code — not hackers, as some claimed at the time — that had been added during a highly complex software upgrade. American Airlines alone estimated this small error cost it 200,000 reservations.

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

3. The explosion of the Ariane 5 (1996)
In 1996, Europe's newest and unmanned satellite-launching rocket, the Ariane 5, was intentionally blown up just seconds after taking off on its maiden flight from Kourou, French Guiana. The European Space Agency estimated that total development of Ariane 5 cost more than $8bn (£4bn). On board Ariane 5 was a $500m (£240m) set of four scientific satellites created to study how the Earth's magnetic field interacts with Solar Winds.

According to a piece in the *New York Times Magazine*, the self-destruction was triggered by software trying to stuff "a 64-bit number into a 16-bit space".

"This shutdown occurred 36.7 seconds after launch, when the guidance system's own computer tried to convert one piece of data — the sideways velocity of the rocket — from a 64-bit format to a 16-bit format. The number was too big, and an overflow error resulted. When the guidance system shut down, it passed control to an identical, redundant unit, which was there to provide backup in case of just such a failure. But the second unit had failed in the identical manner a few milliseconds before. And why not? It was running the same software," the article stated.

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

4. Airbus A380 suffers from incompatible software issues (2006)
The Airbus issue of 2006 highlighted a problem many companies can have with software: what happens when one program doesn't talk to the another. In this case, the problem was caused by two halves of the same program, the CATIA software that is used to design and assemble one of the world's largest aircraft, the Airbus A380.

This was a major European undertaking and, according to *Business Week*, the problem arose with communications between two organisations in the group: the French Dassault Aviation and a Hamburg factory.

Put simply, the German system used an out-of-date version of CATIA and the French system used the latest version. So when Airbus was bringing together two halves of the aircraft, the different software meant that the wiring on one did not match the wiring in the other. The cables could not meet up without being changed.

The problem was eventually fixed, but only at a cost that nobody seems to want to put an absolute figure on. But all agreed it cost a lot, and put the project back a year or more.

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

5. Mars Climate Observer metric problem (1998)
Two spacecraft, the Mars Climate Orbiter and the Mars Polar Lander, were part of a space programme that, in 1998, was supposed to study the Martian weather, climate, and water and carbon dioxide content of the atmosphere. But a problem occurred when a navigation error caused the lander to fly too low in the atmosphere and it was destroyed.

What caused the error? A sub-contractor on the Nasa programme had used imperial units (as used in the US), rather than the Nasa-specified metric units (as used in Europe).

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

6. EDS and the Child Support Agency (2004)
Business services giant EDS waded in with this spectacular disaster, which assisted in the destruction of the Child Support Agency (CSA) and cost the taxpayer over a billion pounds.

EDS's CS2 computer system somehow managed to overpay 1.9 million people and underpay around 700,000, partly because the Department for Work and Pensions (DWP) decided to reform the CSA at the same time as bringing in CS2.

Edward Leigh, chairman of the Public Accounts Committee, was outraged when the National Audit Office subsequently picked through the wreckage: "Ignoring ample warnings, the DWP, the CSA and IT contractor EDS introduced a large, complex IT system at the same time as restructuring the agency. The new system was brought in and, as night follows day, stumbled and now has enormous operational difficulties."

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

7. The two-digit year-2000 problem (1999/2000)
A lot of IT vendors and contractors did very well out of the billions spent to avoid what many feared would be the disaster related to the Millennium Bug. Rumours of astronomical contract rates and retainers abounded.

And the sound of clocks striking midnight in time zones around the world was followed by... not panic, not crashing computer systems, in fact nothing more than new year celebrations.

So why include it here? That the predictions of doom came to naught is irrelevant, as we're not talking about the disaster that was averted, but the original disastrous decision to use and keep using for longer than was either necessary or prudent double digits for the date field in computer programs. A report by the House of Commons Library pegged the cost of fixing the bug at £400bn. And that is why the Millennium Bug deserves a place in the top 10.

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

8. When the laptops exploded (2006)
It all began simply, but certainly not quietly, when a laptop manufactured by Dell burst into flames at a trade show in Japan. There had been rumours of laptops catching fire, but the difference here was that the Dell laptop managed to do it in the full glare of publicity and video captured it in full colour.

(Unfortunately, the video capturing the incident appears to have vanished from the web. If you happen to own a copy, please send it to us as it should make interesting viewing again.)

"We have captured the notebook and have begun investigating the event," Dell spokeswoman Anne Camden reported at the time, and investigate Dell did. At the end of these investigations the problem was traced to an issue with the battery/power supply on the individual laptop that had overheated and caught fire.

It was an expensive issue for Dell to sort out. As a result of its investigation Dell decided that it would be prudent to recall and replace 4.1m laptop batteries.

Company chief executive Michael Dell eventually laid the blame the for the faulty batteries with the manufacturer of the battery cells — Sony. But that wasn't the end of it. Apple reported issues for iPods and Macbooks and many PC suppliers reported the same. Matsushita alone has had to recall around 54 million devices. Sony estimated at the time that the overall cost of supporting the recall programmes of Apple and Dell would amount to between ¥20bn (£90m) and ¥30bn.

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

9. Siemens and the passport system (1999)
It was the summer of 1999, and half a million British citizens were less than happy to discover that their new passports couldn't be issued on time because the Passport Agency had brought in a new Siemens computer system without sufficiently testing it and training staff first.

Hundreds of people missed their holidays and the Home Office had to pay millions in compensation, staff overtime and umbrellas for the poor people queuing in the rain for passports. But why such an unexpectedly huge demand for passports? The law had recently changed to demand, for the first time, that all children under 16 had to get one if they were travelling abroad.

Tory MP Anne Widdecombe summed it up well while berating the then home secretary, Jack Straw, over the fiasco: "Common sense should have told him that to change the law on child passports at the same time as introducing a new computer system into the agency was storing up trouble for the future."

http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

The top 10 IT disasters of all time

10. LA Airport flights grounded (2007)
Some 17,000 planes were grounded at Los Angeles International Airport earlier this year because of a software problem. The problem that hit systems at United States Customs and Border Protection (USCBP) agency was a simple one caused in a piece of lowly, inexpensive equipment.

The device in question was a network card that, instead of shutting down as perhaps it should have done, persisted in sending the incorrect data out across the network. The data then cascaded out until it hit the entire network at the USCBP and brought it to a standstill. Nobody could be authorised to leave or enter the US through the airport for eight hours. Passengers were not impressed.

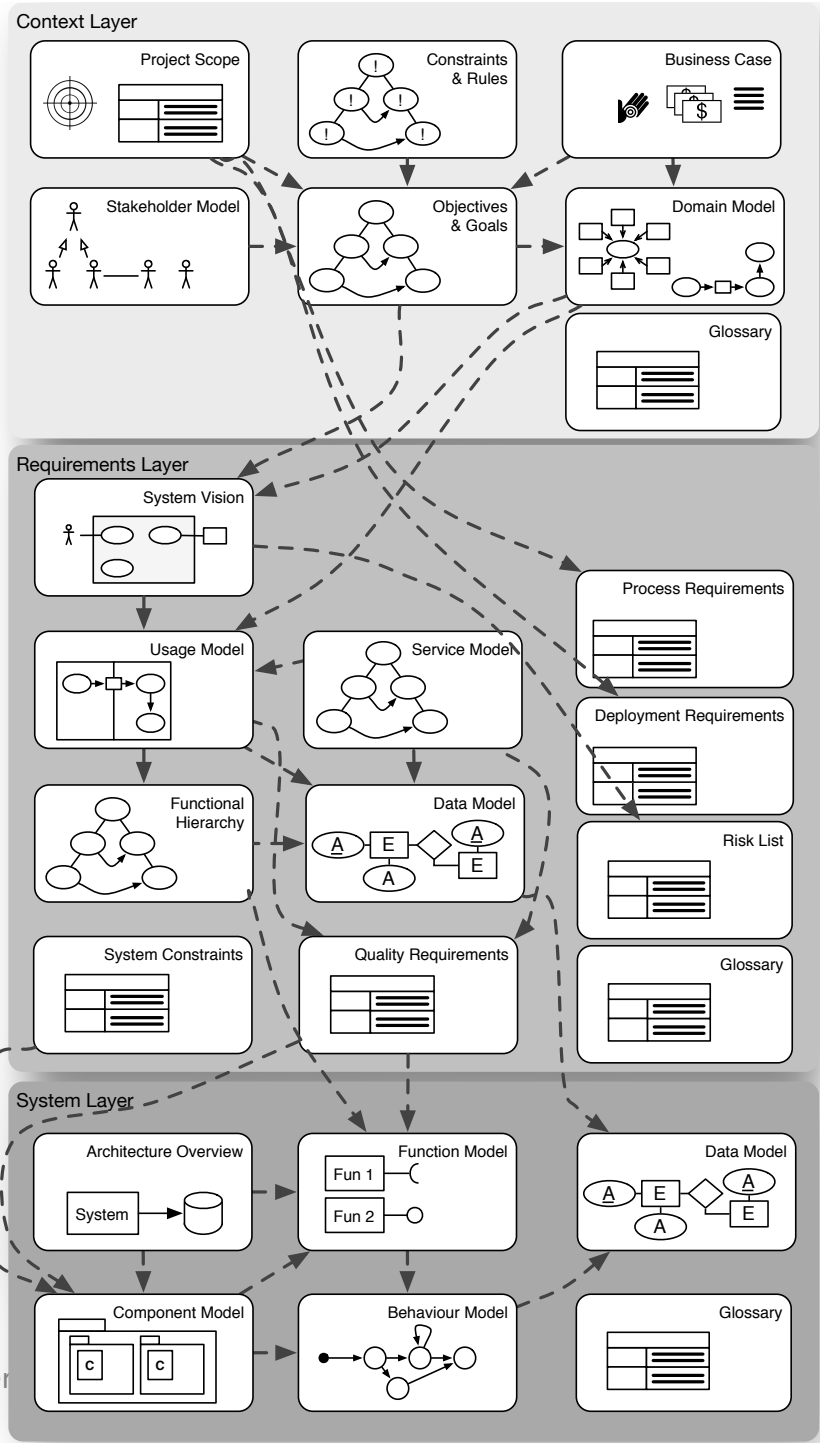
http://www.zdnet.com/the-top-10-it-disasters-of-all-time-3039290976/

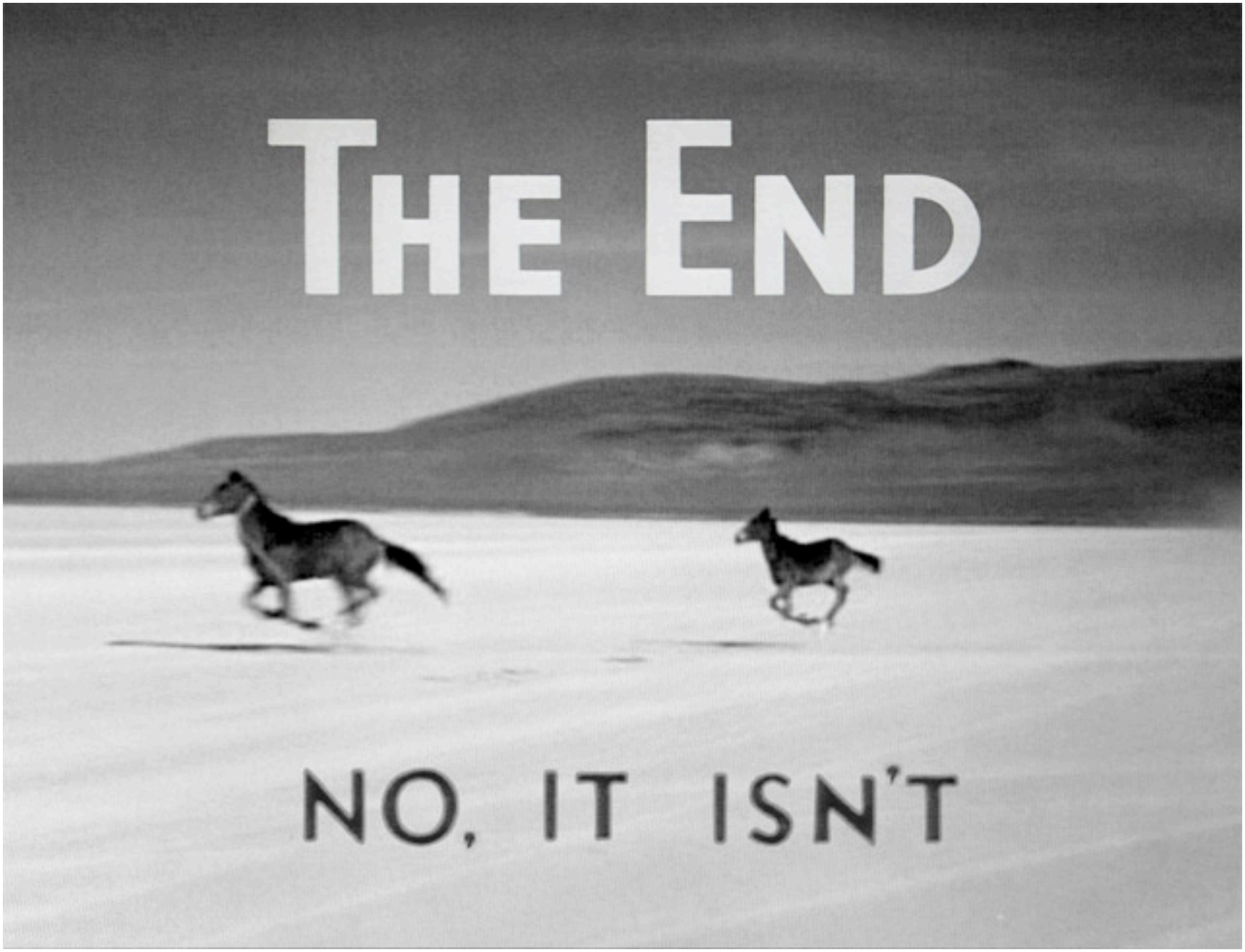
Risk Management Tools

- SWOT analysis (Strengths, weaknesses, opportunities, threats)
- Root-cause analysis

Take-away:

- Requirements Management
  - Major tasks
  - Change Management
  - Risk Management




THE END
NO, IT ISN'T

Requirements Negotiation

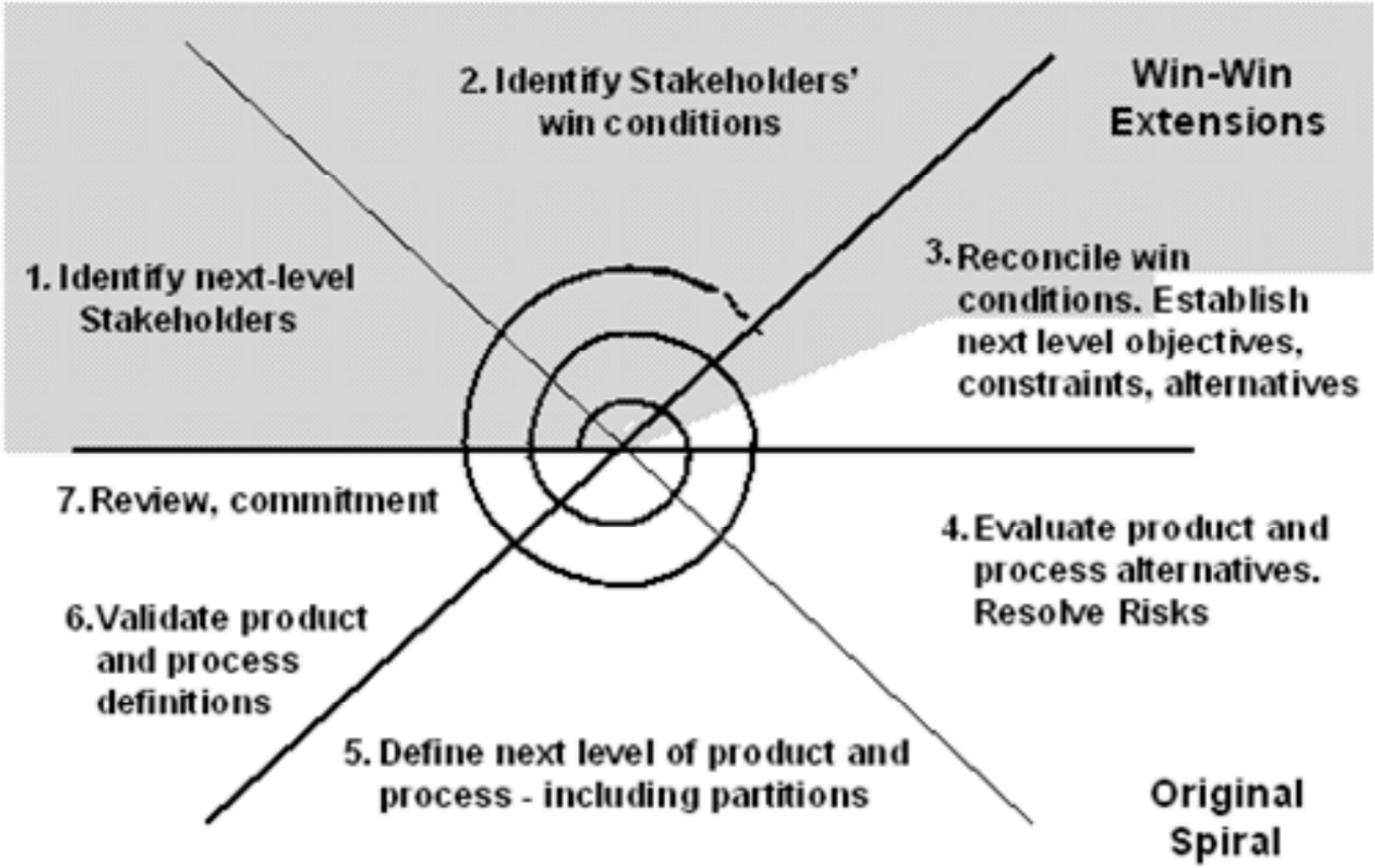


Figure 8.1: The WinWin Spiral Model of Software Engineering includes front-end activities (gray) that show where objectives, constraints, and alternatives come from. This lets users more clearly identify the rationale involved in negotiating win conditions for the product.