Risk Management

CSCI 3300/5300

Why should you care?

- Large Companies
  - Large Projects
  - You are not in charge of project management
  - Can save a project by pointing out the risks
- Small Companies
  - Frequently no formal project management
  - Might not do any risk management
  - Can make a difference by introducing risk management: save the project in the long term
  - Red flag if risks are ignored
- Startup Culture
  - Software Engineer does more than just software
  - Crucial to understand risks and properly manage them

Your Definitions of a Risk

- Something that can result in bad consequences but can be prevented based on the context and condition
- A chance of a consequence given a certain condition and context
- Given a certain condition, there is a possible consequence and context
- A chance that a certain consequence could occur given some conditions and other context information
- Condition that is met that leads to a consequence in a given context
- Each condition results in a consequence that may involve relevant context
- Condition that results in a very positive or negative consequence given a certain context
- Future outcome which people did not really want

Is This A Risk?

- We just started integrating the software
  - and we found out that COTS* products A and B just can't talk to each other
- We've got too much tied into A and B to change
- Our best solution is to build wrappers around A and B to get them to talk via CORBA**
- This will take 3 months and $300K
- It will also delay integration and delivery by at least 3 months

*COTS: Commercial off-the-shelf

**CORBA: Common Object Request Broker Architecture

Is This A Risk?

- No, it is a problem
  - Being dealt with reactively
- Risks involve uncertainties
  - And can be dealt with pro-actively
  - Earlier, this problem was a risk

Earlier, This Problem Was A Risk

- A and B are our strongest COTS choices
  - But there is some chance that they can't talk to each other
  - Probability of loss P(L)
- If we commit to using A and B
  - And we find out in integration that they can't talk to each other
  - We'll add more cost and delay delivery by at least 3 months
  - Size of loss S(L)
- We have a risk exposure of

$$RE = P(L) * S(L)$$

Are there problems in your team project that WERE risks earlier in the semester?

How Can Risk Management Help You Deal With Risks?

- Buying information
- Risk avoidance
- Risk transfer
- Risk reduction
- Risk acceptance

Risk Management Strategies: Buying Information

- Let's spend $30K and 2 weeks prototyping the integration of A and B
- This will buy information on the magnitude of P(L) and S(L)
- If RE = P(L) * S(L) is small, we'll accept and monitor the risk
- If RE is large, we'll use one/some of the other strategies

Other Risk Management Strategies

Risk Avoidance

COTS product C is almost as good as B, and it can talk to A

Delivering on time is worth more to the customer than the small performance loss

Risk Transfers

If the customer insists on using A and B, have them establish a risk reserve (increase budget).

To be used to the extent that A and B can't talk to each other

Risk Reduction

If we build the wrappers and the CORBA corrections right now, we add cost but minimize the schedule delay

Risk Acceptance

If we can solve the A and B interoperability problem, we'll have a big competitive edge on the future procurements

Let's do this on our own money, and patent the solution

What Risk Management Strategies would have been helpful in your team project?

What Else Can Risk Management Help You Do?

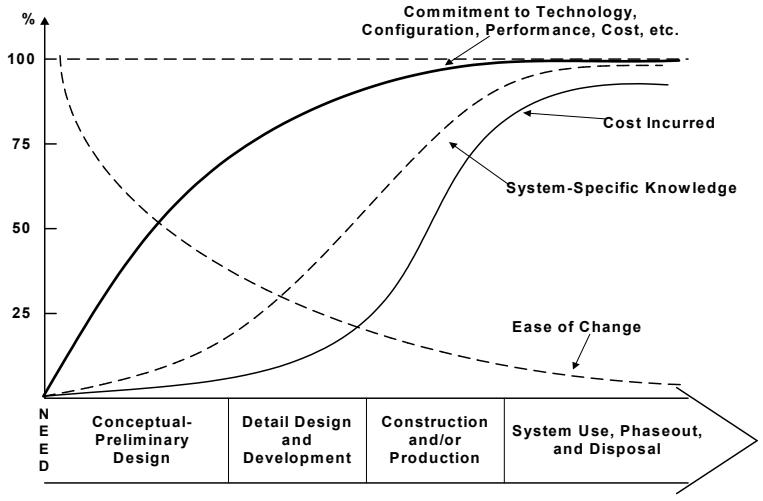
- Determine "How much is enough?" for your products and processes
  - Functionality, documentation, prototyping, COTS evaluation, architecting, testing, formal methods, agility, discipline, …
  - What's the risk exposure of doing too much?
  - What's the risk exposure of doing too little?
- Tailor and adapt your life cycle processes
  - Determine what to do next (specify, prototype, COTS evaluation, business case analysis)
  - Determine how much of it is enough
- Get help from higher management
  - Organize management reviews around top-10 risks

Risk Management Starts on Day One

- Early and Late Risk Resolution
  - Quotes, Notes, and Data
  - Temptations to Avoid
- Early Risk Resolution with the WinWin Spiral Model
  - Identifying Stakeholders and Win Conditions
  - Model Clash and Win-Lose Risk Avoidance
  - Avoiding Cost/Schedule Risks with the SAIV Model

Risk of Delaying Risk Management: Systems

—Blanchard- Fabrycky, 1998



Day One Mistakes

- It's too early to think about risks. We need to:
  - Finalize the requirements
  - Maximize our piece of the pie
  - Converge on the risk management organization, forms, tools, and procedures.
- We don't have time to think about the risks. We need to:
  - Get some code running right away
  - Put on a socko demo for the customers
  - Be decisive. Lead. Make commitments.

Day One Mistakes

- Unwillingness to admit risks exist
  - Leaves impression that you don't know exactly what you're doing
  - Leaves impression that your bosses, customers don't know exactly what they're doing
  - "Success-orientation"
  - "Shoot the messenger" syndrome
- Tendency to postpone the hard parts
  - Maybe they'll go away
  - Maybe they'll get easier, once we do the easy parts
- Unwillingness to invest money and time up front

Software Risk Management

- Risk Assessment
  - Identification
    - Checklists
    - Decision driver analysis
  - Analysis and Prioritization
- Risk Control
  - Risk Management Planning
    - Buying information
    - Avoiding risk
    - Reducing risk
    - Transferring risk
  - Resolution
    - Prototyping
    - Simulations
    - Benchmarks
  - Monitoring

Risk Identification Techniques

- Risk-item checklists
- Decision driver analysis
  - Comparison with experience
  - Win-lose, lose-lose situations
- Decomposition
  - Pareto 80 – 20 phenomena
  - Task dependencies
  - Murphy's law
  - Uncertainty areas
- Model Clashes

Example Risk-item Checklist: Staffing

- Will you project really get all the best people?
- Are there critical skills for which nobody is identified?
- Are there pressures to staff with available warm bodies?
- Are there pressures to overstaff in the early phases?
- Are the key project people compatible?
- Do they have realistic expectations about their project job?
- Do their strengths match their assignment?
- Are they committed full-time?
- Are their task prerequisites (training, clearances, etc.) Satisfied?

Risk Identification: Examining Decision Drivers

- Political versus Technical
  - Choice of equipment
  - Choice of subcontractor
  - Schedule, Budget
  - Allocation of responsibilities
- Marketing versus Technical
  - Gold plating
  - Choice of equipment
  - Schedule, budget
- Solution-driven versus Problem-driven
  - In-house components, tools
  - Artificial intelligence
  - Schedule, Budget
- Short-term versus Long-term
  - Staffing availability versus qualification
  - Reused software productions engineering
  - Premature SRR, PDR
- Outdated Experience

The Top Ten Software Risk Items

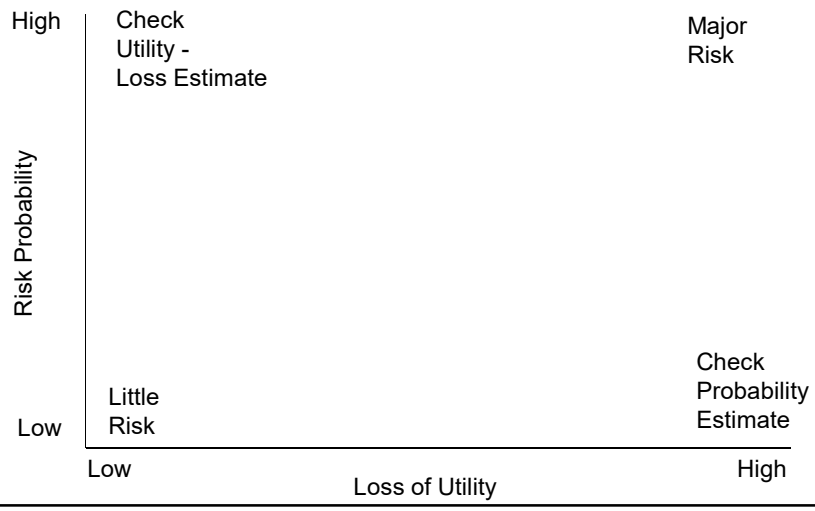
Risk Item	Risk Management Techniques
1. Personnel Shortfalls	Staffing with top talent; key personnel agreements; incentives; team-building; training; tailoring process to skill mix; peer reviews
2. Unrealistic schedules and budgets	Business case analysis; design to cost; incremental development; software reuse; requirements descoping; adding more budget and schedule
3. COTS; external components	Qualification testing; benchmarking; prototyping; reference checking; compatibility analysis; vendor analysis; evolution support analysis
4. Requirements mismatch; gold plating	Stakeholder win-win negotiation; business case analysis; mission analysis; ops-concept formulation; user surveys; prototyping; early users' manual; design/develop to cost
5. User interface mismatch	Prototyping; scenarios; user characterization (functionality, style, workload)

The Top Ten Software Risk Items (Concluded)

6. Architecture, performance, quality	Architecture tradeoff analysis and review boards; simulation; benchmarking; modeling; prototyping; instrumentation; tuning
7. Requirements changes	High change threshold; information hiding; incremental development (defer changes to later increments)
8. Legacy software	Design recovery; phaseout options analysis; wrappers/mediators; restructuring
9. Externally-performed tasks	Reference checking; pre-award audits; award-fee contracts; competitive design or prototyping; team-building
10. Straining Computer Science capabilities	Technical analysis; cost-benefit analysis; prototyping; reference checking

Prioritizing Risks: Risk Exposure

Risk Exposure - (Probability) (Loss of Utility)



Example: Risk Exposure

Risk	Probability	Loss of Utility	Risk Exposure
1. We misinterpret requirements	0.25	100	25
2. Monolithic simulation design is not suitable for parallel simulation	0.5	50	25
3. Simulation may take a long time to run	0.2	30	6
4. Major change to requirements	0.8	10	8

Example Risk 1: Risk of Loosing source code

- Risk of the server crashing?
  - cost of automated backups.
- Risk of hack attack from outside.
  - cost of firewall software, etc.
- Risk of hack attack from inside.
  - cost of off-site backup system.

Example Risk 2: Person X is assigned to three tasks on the critical path. Risk of X getting sick.

- Analyze the possible impact of a delay caused by their absence.
- Determine cost of training another person to do one or two of those tasks.
  - What is the risk exposure versus the training costs?
- Can there be a different task network or assignment of personnel?

Example Risk 3 – Employee turnover

- What happens if they leave?
  - How dependant is our schedule on people with these exact skills?
  - Will information be lost with the person?
- How can we keep them / replace them?
  - How costly would it be to raise salaries?
  - How else could we make them happy?
  - Costs to hire good replacements?

Example Risk 4 - The Market for our product may change

- How risky is it to speed production?
  - Effect of speed on quality?
  - Costs of extra personnel or overtime pay?
- What is the risk of making it a more general product?
  - Cost and time of extra features?

Example Risk 5 – Functionality based on unknown technology

- How likely is it that we don't know enough to fulfill this particular requirement?
- How important is this requirement to product acceptance?
- If someone else knows a lot about this, how much would it cost to get them here?
- Should we try two approaches at the same time?

Example Risk 6 – Data Access

- Who controls access to the database where we are supposed to get our sample data?
  - Is their boss in favor of this project?
  - Are they nice, or do we need to mow their lawn before we can see the data?
- When can we get the data?
  - If we can't get data at the beginning, can we use fake data for a while?
  - How long can we use fake data?
  - How will fake data affect quality?

Example Risk 7

To generate additional revenue, we will release a new version of a financial analysis product. Since it is currently written in COBOL, the next version should be written in COBOL.

- Do you see any potential risks?

Risk Management Plans

1. Why?
   Risk Item Importance, Relation to Project Objectives
2. What, When?
   Risk Resolution Deliverables, Milestones, Activity Nets
3. Who, Where?
   Responsibilities, Organization
4. How?
   Approach (Prototypes, Surveys, Models, …)
5. How Much?
   Resources (Budget, Schedule, Key Personnel)

Risk Monitoring

Milestone Tracking
- Monitoring of risk Management Plan Milestones

Top-10 Risk Item Tracking
- Identify Top-10 risk items
- Highlight these in monthly project reviews
- Focus on new entries, slow-progress items
  Focus review on manger-priority items

Risk Reassessment

Corrective Action

Conclusions

- Risk management starts on Day One
  - Delay and denial are serious career risks
  - Data provided to support early investment
- Win Win spiral model provides process framework for early risk resolution
  - Stakeholder identification and win condition reconciliation
  - Anchor point milestones
- Risk analysis helps determine “how much is enough”
  - Testing, planning, specifying, prototyping,…
  - Buying information to reduce risk