Risk Management

**What is risk?**

The Oxford dictionary defines risk as “the possibility that something unpleasant or unwelcome will happen”. So risk is an unpleasant and uncertain event related to the future. Risks example can be:

* **Environmental risk**: wildlife smuggling may cause the introduction of invasive and harmful species into an ecosystem, which can endanger indigenous wildlife.
* **Security risk**: using your credit card online could be a risk. Your information can be stolen if you are buying on unreliable websites, using an infected pc, not using a secure connection or using a secure connection with bugs (SSL anyone?).

In computer science, a common risk can be a schedule slip or a cost overrun. The loss is often considered in terms of direct financial loss, but also can be a loss in terms of credibility and future business.

**Why is the software world interested in risk?**

One of the main reasons of risk management is the cost factor. Mitigate events that might adversely impact a project can reduce the amount of money spent on it. “The software industry is fraught with failed and delayed projects, most of which far exceed their original budget.

The Standish Group is based in Boston, Massachusetts and is the Information Technology leader in project and value performance. We are a group of highly dedicated professionals with years of practical experience in assessing risk, cost, return and value for Information Technology (IT) Investments.

The Standish Group reported that only 28% of software projects are completed on time and on budget. Over 23% of software projects are cancelled before they ever get completed, and 49% of projects cost 145% of their original estimates.” (Standish, 1995)

Another interesting reason of being interested in risks is related to the post-mortem documentation. “A project post-mortem is a process, usually performed at the conclusion of a project, to determine and analyse elements of the project that were successful or unsuccessful. Project post-mortems are intended to inform process improvements which mitigate future risks and to promote iterative best practices.” (Wikipedia)

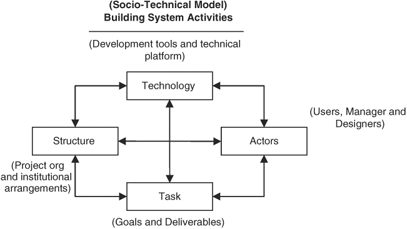
Many post-mortems of software project disasters indicate that problems would have been avoided (or strongly reduced) if there had been an explicit early concern with identifying and resolving high-risk elements.

**Reactive vs. Proactive Risk Strategies**

Reactive risk strategies have been called the “Indiana Jones School of risk management”. In the movies that carried his name, Indiana Jones, when faced with overwhelming difficulty, would invariably say, “Don’t worry, I’ll think of something!” never worrying about problems until they happened. Sadly, the average software project manager is not Indiana Jones. Commonly the software team does nothing about risks until something goes wrong. Then the team flies into action in an attempt to correct the problem rapidly. This is often called “fire fighting mode”.

A considerably more intelligent strategy for risk management is to be proactive. A proactive strategy begins long before technical work is initiated.

**What risks are of concern when managing a software project?**



There are various ways of categorise risks. Kalle Lyytinen and his colleagues have proposed a sociotechnical model.

The box labelled **Actors**refers to all the people involved in the development of the application in question. A typical risk in this area is that high turnover leads to expertise of value to the project being lost. The box labelled **Technology**refers both to the tools used to implement the application and to the technology embedded in the delivered products. Risks here could relate to the appropriateness of the technologies and to possible faults within them, especially if they are novel. **Structure** describes the management structures and systems. For example, the implementation might need user participation in some tasks but the responsibility for managing the users’ contribution might not be clearly allocated. **Tasks**relate to the work planned. For instance, the complexity of the work might lead to delays because of the additional time required to integrate the large number of components.

All the boxes are interlinked. Risks often arise from the relationships between factors – for example between technology and people. If a development technology is novel then the developers might not be experienced in its use and delay results.

**How can risks effects be foreseen? What is the likelihood it will go wrong?**

**Risk management** is a series of steps whose objectives are to identify, address, and eliminate software risk items before they become either threats to successful software operation or a major source of expensive rework. (Boehm, 1989)

Remember that every plan is based on assumptions and risk management tries to plan for and control the situations where those assumptions become incorrect. For example, a plan can be based on the assumption that we have available, for example, three experienced programmers for a project. However, suppose two developers then left for better-paid jobs and the only replacement we could find happens to be a trainee. What could have we done to prevent this situation? A solution could have been having a fund for emergencies like this one and pay our employers more.

**Slide 22**

Graph Explained: The risk management process can be broken down into two interrelated phases, risk assessment and risk control. These phases are further broken down. Risk assessment involves risk identification, risk analysis, and risk prioritization. Risk control involves risk planning, risk mitigation, and risk monitoring. It is essential that risk management be done iteratively, throughout the project, as a part of the team’s project management routine.

* **Risk Assessment**
  + *Risk identification:* listing project-specific risk items that are likely to compromise a project’s success. The two main approaches to the identification of risks are use of checklists and brainstorming. Checklists are simply lists of the risks that have been found to occur regularly in software development projects. During a brainstorming people with knowledge of different part of the system are gathered together to identify problems that might occur.  
    Question: imagine you were a travel agent who wants to organise a trip in Scotland for foreign students during summer. What risk can you identify? Possible answers can be:   
    People forget/lose their Passports, get lost somewhere, don’t know how to speak the language when left alone, don’t have umbrellas or waterproof clothing, don’t have appropriate shoes for mountaineering, get sunburns.

Top 10 Risks in Software Project Management (by Barry Boehm)

* + - Personnel shortfalls (deficit)
    - Unrealistic schedules & budgets
    - Developing the wrong functions & properties
    - Developing the wrong user interface -> prototyping
    - Gold-plating (plating refers to adding features to the software that are only marginally useful) -> cost-benefit analysis
    - Continuing stream of requirements changes -> information hiding
    - Shortfalls in externally supplied components
    - Shortfalls in externally performed tasks
    - Real-time performance shortfalls
    - Straining computer-science capabilities
  + *Risk analysis:* assessing the loss probability and loss magnitude for each identified risk item, and assessing compound risks.
  + *Risk prioritisation:* ordering and ranking the risk items identified and analysed.

A way is needed of distinguish the damaging and likely risks. This can be done by estimating the risk the risk exposure for each risk using the formula:   
 risk exposure = (potential damage) x (probability of occurrence)  
The damage could be assessed as money value or loss in credibility of the company.   
This calculation assume that the damage is a constant but there are situations in which it is no, e.g. more software is created and more time would be needed to re-create it if it were lost.

* + - See each point of the travel agency and check probability

Boehm suggests that planners focus attention on the 10 risks with the highest risk exposure scores. Unfortunately, the potential damage values are likely to be subjective and different analysts might pick different numbers.

Another approach is to use qualitative descriptions of the possible impact and the likelihood of each risk. We can have a probability level associated with a range. For example, the probability that someone gets sunburns in Scotland is less than 20%, (this is a qualitative descriptor) we can call this range Low. And so forth.

Add table

Where the potential damage and likelihood of a risk are defined by qualitative descriptors the risk exposure cannot be calculated by multiplying the two factors together. In this case a probability impact matrix is used.

The black line is called tolerance line, it is different from project to project

Where would you put our risks?

* **Risk Control (aka RMMM)**
  + *Risk-management planning*: risk management plans should be developed for each of the previous prioritized risks so that proactive action can take place.
    - Risk acceptance: this is the do-nothing option. We will already, in the risk prioritisation, have decided to ignore some risks in order to concentrate on the more likely or damaging. We could decide that the damage inflicted by some risks would be less than the costs of action that might reduce the probability of a risk happening.
    - Risk avoidance: some activities may be so prone to accident that it is best to avoid them altogether. If you are worried about sharks then don’t go in water! For example, given all the problems with developing software solutions from scratch, managers might decide to buy an off-the-shelf solution.
    - Risk reduction: here we decide to go ahead with a course of action despite the risks, but take precautions that reduce the probability of the risk. For example, if the team is concerned that the use of a new programming language may cause a schedule delay, the budget might contain a line item entitled “potential schedule” to cover a potential schedule slip. Because the budget already covers the potential slip, the financial risk to the organization is reduced.

Risk Mitigation can sometimes be distinguished from risk reduction. Risk reduction attempts to reduce the likelihood of the risk occurring. Risk mitigation is action taken to ensure that the impact of the risk is lessened when it occur. For example taking regular back-ups of data storage would reduce the impact of data corruption but not its likelihood.

* + - Risk transfer: in this case the risk is transferred to another person or organization. For example if a company wants to create its first 3D game and they are not very good at 3D modelling they can choose to a well-established external organization to carry out this part of the project. In the meantime the company can recruit people and learn more about 3D modelling for their next games.
  + *Risk resolution*: producing a situation in which risk items are eliminated or resolved. Suppose we take this action, is it effective? If so how much? Have we chosen a good solution if this problem will show up?
  + *Risk monitoring*: risks need to be revisited at regular intervals for the team to re-evaluate each risk to determine when new circumstances caused its probability and/or impact to change. At each interval, some risks may be added to the list and others taken away. Risks need to be reprioritized to see which are moved “above the line” and need to have action plans and which move “below the line” and no longer need action plans.

**RMMM… what is it?**

Risk Mitigation, Monitoring and Management.

The goal of the risk mitigation, monitoring and management plan is to identify as many potential risks as possible. When all risks have been identified, they will then be evaluated to determine their probability of occurrence, and how the project will be affected if they do occur. Plans will then be made to avoid each risk, to track each risk to determine if it is more or less likely to occur, and to plan for those risks should they occur.

It is the organization’s responsibility to perform risk mitigation, monitoring, and management in order to produce a quality product. The quicker the risks can be identified and avoided, the smaller the chances of having to face that particular risk’s consequence. The fewer consequences suffered as a result of good RMMM plan, the better the product and the smoother the development process.

There are software used to evaluate risks and the effects of uncertainty.

PERT (Program Evaluation and Review Technique) is a method to analyse the involved tasks in completing a given project, especially the time needed to complete each task, and to identify the minimum time needed to complete the total project.

PERT was developed primarily to simplify the planning and scheduling of large and complex projects. It was developed for the U.S. Navy Special Projects Office in 1957 to support the U.S. Navy's Polaris nuclear submarine project. It was able to incorporate uncertainty by making it possible to schedule a project while not knowing precisely the details and durations of all the activities. It is more of an event-oriented technique rather than start- and completion-oriented, and is used more in projects where time is the major factor rather than cost. It is applied to very large-scale, one-time, complex, non-routine infrastructure and Research and Development projects. An example of this was for the 1968 Winter Olympics in Grenoble which applied PERT from 1965 until the opening of the 1968 Games.