

Chapter 28

Risk Management

- Introduction
- Risk identification
- Risk projection (estimation)
- Risk mitigation, monitoring, and management

Introduction

Definition of Risk

- A risk is a potential problem – it might happen and it might not
- Conceptual definition of risk
 - Risk concerns future happenings
 - Risk involves change in mind, opinion, actions, places, etc.
 - Risk involves choice and the uncertainty that choice entails
- Two characteristics of risk
 - Uncertainty – the risk may or may not happen, that is, there are no 100% risks (those, instead, are called constraints)
 - Loss – the risk becomes a reality and unwanted consequences or losses occur

Risk Categorization – Approach #1

- Project risks
 - They threaten the project plan
 - If they become real, it is likely that the project schedule will slip and that costs will increase
- Technical risks
 - They threaten the quality and timeliness of the software to be produced
 - If they become real, implementation may become difficult or impossible
- Business risks
 - They threaten the viability of the software to be built
 - If they become real, they jeopardize the project or the product

(More on next slide)

Risk Categorization – Approach #1

(continued)

- Sub-categories of Business risks
 - **Market risk** – building an excellent product or system that no one really wants
 - **Strategic risk** – building a product that no longer fits into the overall business strategy for the company
 - **Sales risk** – building a product that the sales force doesn't understand how to sell
 - **Management risk** – losing the support of senior management due to a change in focus or a change in people
 - **Budget risk** – losing budgetary or personnel commitment

Risk Categorization – Approach #2

- Known risks
 - Those risks that can be uncovered after careful evaluation of the project plan, the business and technical environment in which the project is being developed, and other reliable information sources (e.g., unrealistic delivery date)
- Predictable risks
 - Those risks that are extrapolated from past project experience (e.g., past turnover)
- Unpredictable risks
 - Those risks that can and do occur, but are extremely difficult to identify in advance

Reactive vs. Proactive Risk Strategies

- Reactive risk strategies
 - "Don't worry, I'll think of something"
 - The majority of software teams and managers rely on this approach
 - Nothing is done about risks until something goes wrong
 - The team then flies into action in an attempt to correct the problem rapidly (fire fighting)
 - Crisis management is the choice of management techniques
- Proactive risk strategies
 - Steps for risk management are followed (see next slide)
 - Primary objective is to avoid risk and to have a contingency plan in place to handle unavoidable risks in a controlled and effective manner

Steps for Risk Management

- 1) Identify possible risks; recognize what can go wrong
- 2) Analyze each risk to estimate the probability that it will occur and the impact (i.e., damage) that it will do if it does occur
- 3) Rank the risks by probability and impact
 - Impact may be negligible, marginal, critical, and catastrophic
- 4) Develop a contingency plan to manage those risks having high probability and high impact

Risk Identification

Known and Predictable Risk Categories

- **Product size** – risks associated with overall size of the software to be built
- **Business impact** – risks associated with constraints imposed by management or the marketplace
- **Customer characteristics** – risks associated with sophistication of the customer and the developer's ability to communicate with the customer in a timely manner
- **Process definition** – risks associated with the degree to which the software process has been defined and is followed
- **Development environment** – risks associated with availability and quality of the tools to be used to build the project
- **Technology to be built** – risks associated with complexity of the system to be built and the "newness" of the technology in the system
- **Staff size and experience** – risks associated with overall technical and project experience of the software engineers who will do the work

Risk Components and Drivers

- The project manager identifies the risk drivers that affect the following risk components
 - **Performance risk** - the degree of uncertainty that the product will meet its requirements and be fit for its intended use
 - **Cost risk** - the degree of uncertainty that the project budget will be maintained
 - **Support risk** - the degree of uncertainty that the resultant software will be easy to correct, adapt, and enhance
 - **Schedule risk** - the degree of uncertainty that the project schedule will be maintained and that the product will be delivered on time
- The impact of each risk driver on the risk component is divided into one of four impact levels
 - Negligible, marginal, critical, and catastrophic
- Risk drivers can be assessed as impossible, improbable, probable, and frequent

Risk Projection (Estimation)

Assessing Risk Impact

- Three factors affect the consequences that are likely if a risk does occur
 - **Its nature** – This indicates the problems that are likely if the risk occurs
 - **Its scope** – This combines the severity of the risk (how serious was it) with its overall distribution (how much was affected)
 - **Its timing** – This considers when and for how long the impact will be felt
- The overall risk exposure formula is $RE = P \times C$
 - P = the probability of occurrence for a risk
 - C = the cost to the project should the risk actually occur
- Example
 - P = 80% probability that 18 of 60 software components will have to be developed
 - C = Total cost of developing 18 components is \$25,000
 - $RE = .80 \times \$25,000 = \$20,000$

Risk Mitigation, Monitoring, and Management

Background

- An effective strategy for dealing with risk must consider three issues
 - Risk mitigation (i.e., avoidance)
 - Risk monitoring
 - Risk management and contingency planning
- Risk mitigation (avoidance) is the primary strategy and is achieved through a plan
 - Example: Risk of high staff turnover (see next slide)

Strategy for Reducing Staff Turnover

- Meet with current staff to determine causes for turnover (e.g., poor working conditions, low pay, competitive job market)
- Mitigate those causes that are under our control before the project starts
- Once the project commences, assume turnover will occur and develop techniques to ensure continuity when people leave
- Organize project teams so that information about each development activity is widely dispersed
- Define documentation standards and establish mechanisms to ensure that documents are developed in a timely manner
- Conduct peer reviews of all work (so that more than one person is "up to speed")
- Assign a backup staff member for every critical technologist

Background (continued)

- During risk monitoring, the project manager monitors factors that may provide an indication of whether a risk is becoming more or less likely
- Risk management and contingency planning assume that mitigation efforts have failed and that the risk has become a reality
- RMMM steps incur additional project cost
 - Large projects may have identified 30 – 40 risks
- Risk is not limited to the software project itself
 - Risks can occur after the software has been delivered to the user

(More on next slide)

Background (continued)

- Software safety and hazard analysis
 - These are software quality assurance activities that focus on the identification and assessment of potential hazards that may affect software negatively and cause an entire system to fail
 - If hazards can be identified early in the software process, software design features can be specified that will either eliminate or control potential hazards

The RMMM Plan

- The RMMM plan may be a part of the software development plan or may be a separate document
- Once RMMM has been documented and the project has begun, the risk mitigation, and monitoring steps begin
 - Risk mitigation is a problem avoidance activity
 - Risk monitoring is a project tracking activity
- Risk monitoring has three objectives
 - To assess whether predicted risks do, in fact, occur
 - To ensure that risk aversion steps defined for the risk are being properly applied
 - To collect information that can be used for future risk analysis
- The findings from risk monitoring may allow the project manager to ascertain what risks caused which problems throughout the project

Seven Principles of Risk Management

- **Maintain a global perspective**
 - View software risks within the context of a system and the business problem that is intended to solve
- **Take a forward-looking view**
 - Think about risks that may arise in the future; establish contingency plans
- **Encourage open communication**
 - Encourage all stakeholders and users to point out risks at any time
- **Integrate risk management**
 - Integrate the consideration of risk into the software process
- **Emphasize a continuous process of risk management**
 - Modify identified risks as more becomes known and add new risks as better insight is achieved
- **Develop a shared product vision**
 - A shared vision by all stakeholders facilitates better risk identification and assessment
- **Encourage teamwork when managing risk**
 - Pool the skills and experience of all stakeholders when conducting risk management activities