Risk management :Introduction, nature of risk, managing risk, risk identification, risk analysis

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

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

Nature of Project Risks

- Planning assumptions
- Estimation errors
- Eventualities

Planning Assumptions

- Why assumptions
  - Uncertainties in early stage of the project
- Common assumption:
  - “Everything will go smoothly”
    - Environment is reliable and fixed
    - Design will be perfect first time
    - Coding will be ‘nearly perfect’
- Guidelines
  - List all the assumptions
  - Identify the effects of these assumptions on the project if they are no longer valid

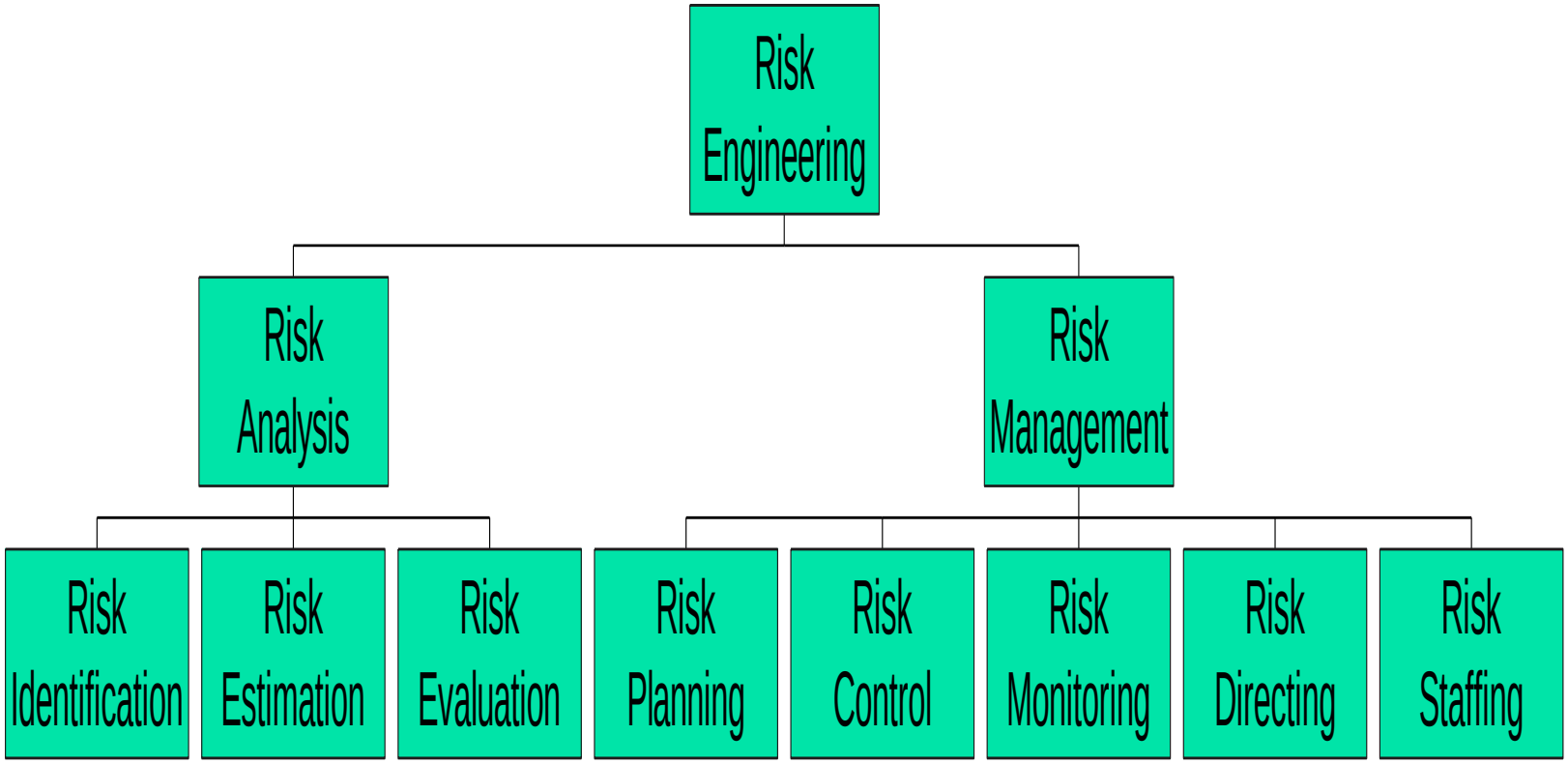
Estimation Errors

- Difficult to have accurate size or time estimations
  - Lack of experience of similar tasks
  - Lack of historical data
  - Nature of the task
- Estimation can be improved by analyzing historic data for similar tasks and similar projects
  - Keep historic data of your estimation and the actual performance
  - Compare your estimation and the actual value
  - Classify the tasks that are easy or difficult to give accurate estimation

Eventualities

- Unexpected and unimaginable events
- Common unexpected events
  - Hardware cannot be delivered on time
  - Requirements specification needs to be rewritten
  - Staffing problem

Boehm's Risk Engineering



Managing Risk

- Step 1: Risk Identification
  - Generate a list of possible risks through brainstorming, problem identification and risk profiling.
    - Macro risks first, then specific events
- Step 2: Risk Assessment
  - Scenario analysis for event probability and impact
  - Risk assessment matrix
  - Failure Mode and Effects Analysis (FMEA)
  - Probability analysis
    - Decision trees, NPV, and PERT
  - Semi quantitative scenario analysis

Risk Identification

Background

- Risk identification is a systematic attempt to specify threats to the project plan
- By identifying known and predictable risks, the project manager takes a first step toward avoiding them when possible and controlling them when necessary
- Generic risks
  - Risks that are a potential threat to every software project
- Product-specific risks
  - Risks that can be identified only by those a with a clear understanding of the technology, the people, and the environment that is specific to the software that is to be built.
  - This requires examination of the project plan and the statement of scope.
  - "What special characteristics of this product may threaten our project plan?"

Risk Item Checklist

- Used as one way to identify risks
- Focuses on known and predictable risks in specific subcategories (see next slide)
- Can be organized in several ways
  - A list of characteristics relevant to each risk subcategory
  - Questionnaire that leads to an estimate on the impact of each risk
  - A list containing a set of risk component and drivers and their probability of occurrence

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

Questionnaire on Project Risk

(Questions are ordered by their relative importance to project success)

1) Have top software and customer managers formally committed to support the project?
2) Are end-users enthusiastically committed to the project and the system/product to be built?
3) Are requirements fully understood by the software engineering team and its customers?
4) Have customers been involved fully in the definition of requirements?
5) Do end-users have realistic expectations?
6) Is the project scope stable?

(More on next slide)

Questionnaire on Project Risk (continued)

7) Does the software engineering team have the right mix of skills?
8) Are project requirements stable?
9) Does the project team have experience with the technology to be implemented?
10) Is the number of people on the project team adequate to do the job?
11) Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built?

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

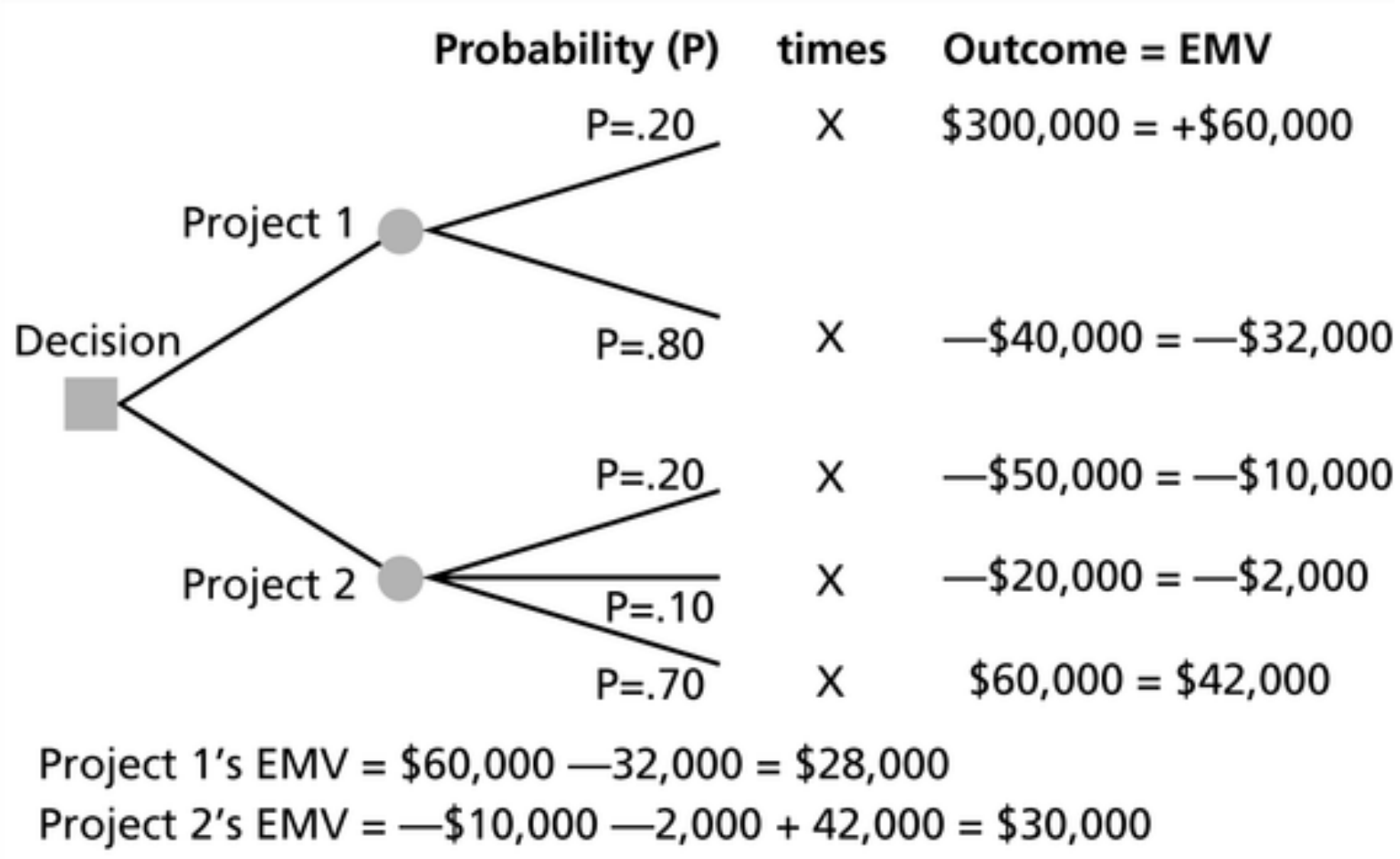
Risk Analysis

- Main techniques include
  - Decision tree analysis
  - simulation

Decision Trees and Expected Monetary Value (EMV)

- A decision tree is a diagramming method used to help you select the best course of action in situations in which future outcomes are uncertain
- EMV is a type of decision tree where you calculate the expected monetary value of a decision based on its risk event probability and monetary value

Figure 10-3. Expected Monetary Value (EMV) Example



Project 1's EMV = $60,000 —32,000 = $28,000

Project 2's EMV = —$10,000 —2,000 + 42,000 = $30,000

Simulation

- Simulation uses a representation or model of a system to analyze the expected behavior or performance of the system
- Monte Carlo analysis simulates a model's outcome many time to provide a statistical distribution of the calculated results
- To use a Monte Carlo simulation, you must have three estimates plus an estimate of the likelihood of the estimate being between the optimistic and most likely values

Application & Scope of research

Application

- FAA Air traffic organization

Scope of research

- Risk assessment