SOFTWARE ENGINEERING

Presented by:

B.Pranalini

UNIT 5 PART 1: RISK MANAGEMENT

Index:

WHAT IS RISK?

- Risks are potential problems that may affect successful completion of a software project.
- Risks involve uncertainty and potential losses.
- Risk analysis and management are intended to help a software team understand and manage uncertainty during the development process.

REACTIVE VERSUS PROACTIVE RISK STRATEGIES

- Reactive strategies
  - very common, also known as fire fighting mode
  - project team sets resources aside to deal with problems
  - team does nothing until a risk becomes a problem
- Proactive strategies
  - risk management begins long before technical work starts, risks are identified and prioritized by importance
  - team builds a plan to avoid risks if they can or to minimize risks if they turn into problems

SOFTWARE RISKS

- Project risks
  - threaten the project plan
- Technical risks
  - threaten product quality and the timeliness of the schedule
- Business risks
  - threaten the viability of the software to be built (market risks, strategic risks, sales risk, management risks, budget risks)

- Known risks
  - predictable from careful evaluation of current project plan and those extrapolated from past project experience
- Unknown risks
  - some problems will simply occur without warning

RISK IDENTIFICATION

- Product-specific risks
  - the project plan and software statement of scope are examined to identify any special characteristics of the product that may threaten the project plan
- Generic risks
  - are potential threats to every software project
    - product size
    - customer characteristics
    - development environment
    - technology to be built

R isk identification and focuses on some subset of known and predictable risks in the following generic subcategories:

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

ASSESSING OVERALL PROJECT RISK

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?
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

- *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.

Impact assessment.
Source: [Boe89].

Category \ Components		Performance	Support	Cost	Schedule
Catastrophic	1	Failure to meet the requirement would result in mission failure		Failure results in increased costs and schedule delays with expected values in excess of $500K	
	2	Significant degradation to nonachievement of technical performance	Nonresponsive or unsupportable software	Significant financial shortages, budget overrun likely	Unachievable IOC
Critical	1	Failure to meet the requirement would degrade system performance to a point where mission success is questionable		Failure results in operational delays and/or increased costs with expected value of $100K to $500K	
	2	Some reduction in technical performance	Minor delays in software modifications	Some shortage of financial resources, possible overruns	Possible slippage in IOC
Marginal	1	Failure to meet the requirement would result in degradation of secondary mission		Costs, impacts, and/or recoverable schedule slips with expected value of $1K to $100K	
	2	Minimal to small reduction in technical performance	Responsive software support	Sufficient financial resources	Realistic, achievable schedule
Negligible	1	Failure to meet the requirement would create inconvenience or nonoperational impact		Error results in minor cost and/or schedule impact with expected value of less than $1K	
	2	No reduction in technical performance	Easily supportable software	Possible budget underrun	Early achievable IOC

Note: [1] The potential consequence of undetected software errors or faults.
[2] The potential consequence if the desired outcome is not achieved.

RISK PROJECTION

- *Risk projection, also called risk estimation, attempts to rate each risk in two ways—*

(1) the likelihood or probability that the risk is real

(2) the consequences of the problems associated with the risk, should it occur.

- Work along with other managers and technical staff to perform four risk projection steps:
  1. Establish a scale that reflects the perceived likelihood of a risk.
  2. Delineate the consequences of the risk.
  3. Estimate the impact of the risk on the project and the product.
  4. Assess the overall accuracy of the risk projection so that there will be no misunderstandings.
- The risk drivers affecting each risk component are
  - classified according to their impact category
  - potential consequences of each undetected software fault or unachieved project outcome are described

DEVELOPING A RISK TABLE

- A risk table provides with a simple technique for risk projection. A sample risk table is

Sample risk table prior to sorting

Risks	Category	Probability	Impact	RMMM
Size estimate may be significantly low	PS	60%	2	
Larger number of users than planned	PS	30%	3	
Less reuse than planned	PS	70%	2	
End-users resist system	BU	40%	3	
Delivery deadline will be tightened	BU	50%	2	
Funding will be lost	CU	40%	1	
Customer will change requirements	PS	80%	2	
Technology will not meet expectations	TE	30%	1	
Lack of training on tools	DE	80%	3	
Staff inexperienced	ST	30%	2	
Staff turnover will be high	ST	60%	2	
Σ				
Σ				
Σ				

Impact values:
1—catastrophic
2—critical
3—marginal
4—negligible

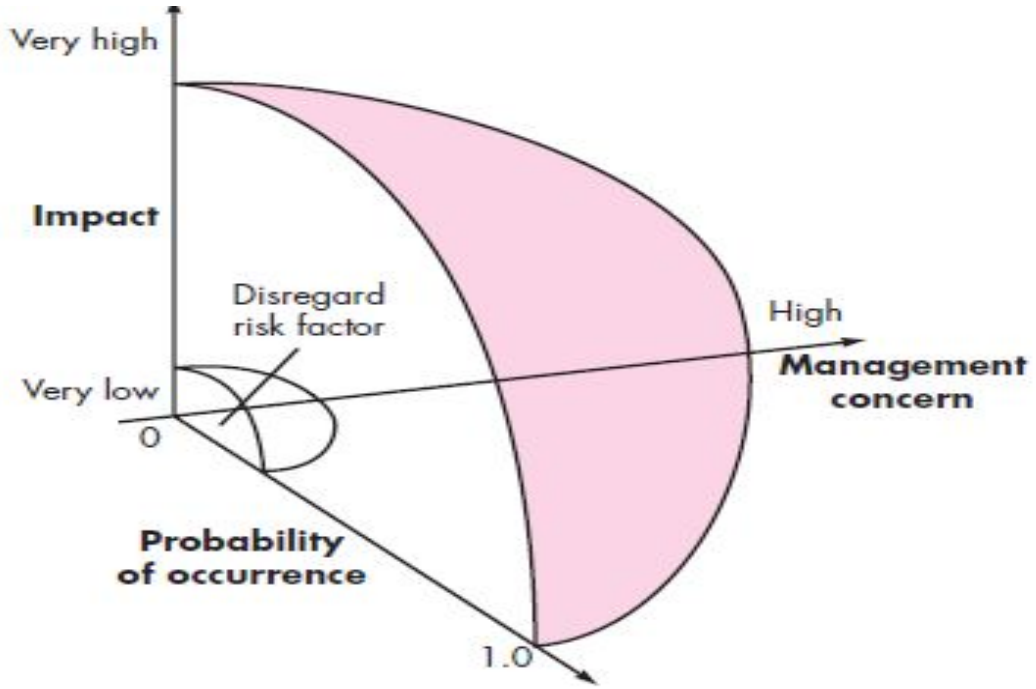
Risk Estimation

1. Establish a scale indicating perceived likelihood of risk occurring
2. Determine consequences.
3. Estimate impact of consequences on project (for each risk).
4. Note overall accuracy of risk projection (to avoid misunderstandings).

Risk Table Construction

- List all risks in the first column of the table
- Classify each risk and enter the category label in column two
- Determine a probability for each risk and enter it into column three
- Enter the severity of each risk (negligible, marginal, critical, catastrophic) in column four.
- Sort the table by probability and impact value
- Determine the criteria for deciding where the sorted table will be divided into the first priority concerns and the second priority concerns
- First priority concerns must be managed (a fifth column can be added to contain a pointer into the RMMM document)

Risk and management concern



- Risk impact and probability have a distinct influence on management concern.
- A risk factor that has a high impact but a very low probability of occurrence should not absorb a significant amount of management time.
- However, high-impact risks with moderate to high probability and low-impact risks with high probability should be carried forward into the risk analysis.

ASSESSING RISK IMPACT

- Define referent levels for each project risk that can cause project termination
  - performance degradation
  - cost overrun
  - support difficulty
  - schedule slippage
- Attempt to develop a relationship between each risk triple (risk, probability, impact) and each of the reference levels.
- Predict the set of referent points that define a region of termination, bounded by a curve or areas of uncertainty.
- Try to predict how combinations of risks will affect a referent level

To determine the overall consequences of a risk:

(1) determine the average probability of occurrence value for each risk component;

(2) using Figure Impact assessment, determine the impact for each component based on the criteria shown, and

(3) complete the risk table and analyze the results as described in the preceding sections.

The overall *risk exposure RE is determined using the following relationship*

RE=*P* **C*

where *P is the probability of occurrence for a risk, and C is the cost to the project should the risk occur.*

For example, assume that the software team defines a project risk in the following manner:

Risk identification. Only 70 percent of the software components scheduled for reuse will, in fact, be integrated into the application. The remaining functionality will have to be custom developed.

Risk probability. 80 percent (likely).

Risk impact. Sixty reusable software components were planned. If only 70 percent can be used, 18 components would have to be developed from scratch (in addition to other custom software that has been scheduled for development). Since the average component is 100 LOC and local data indicate that the software engineering cost for each LOC is $14.00, the overall cost (impact) to develop the components would be 18 * 100 * 14 = $25,200.

Risk exposure. RE = 0.80 * 25,200 ~ $20,200.

RISK REFINEMENT

- Process of restating the risks as a set of more detailed risks that will be easier to mitigate, monitor, and manage.
- CTC (condition-transition-consequence) format may be a good representation for the detailed risks
- Example: given that <condition> then there is a concern that (possibly) <consequence>.

- This general condition can be refined in the following manner:
  - **Subcondition 1.** Certain reusable components were developed by a third party with no knowledge of internal design standards.
  - **Subcondition 2.** The design standard for component interfaces has not been solidified and may not conform to certain existing reusable components.
  - **Subcondition 3.** Certain reusable components have been implemented in a language that is not supported on the target environment.

RISK MITIGATION, MONITORING, AND MANAGEMENT (RMMM)

- Risk mitigation
  - proactive planning for risk avoidance
- To mitigate this risk, you would develop a strategy for reducing turnover. Among the possible steps to be taken are:
  - 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 your 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 work product standards and establish mechanisms to be sure that all models and 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.

- Risk monitoring
  - assessing whether predicted risks occur or not
  - ensuring risk aversion steps are being properly applied
  - collect information for future risk analysis
  - determining which risks caused which problems
- *Risk Management and contingency planning*
  - actions to be taken in the event that mitigation steps have failed and the risk has become a live problem

Risk Mitigation Example:

Risk: loss of key team members

- Determine causes of job turnover.
- Eliminate causes before project starts.
- After project starts, assume turnover is going to occur and work to ensure continuity.
- Make sure teams are organized and distribute information widely.
- Define documentation standards and be sure documents are produced in a timely manner.
- Conduct peer review of all work.
- Define backup staff.

- *Software safety and hazard analysis*

  Risks are also associated with software failures that occur in the field after the development project has ended.

  - Computers control many mission critical applications today (weapons systems, flight control, industrial processes, etc.).
  - Software safety and hazard analysis are quality assurance activities that are of particular concern for these types of applications

THE RMMM PLAN

- A risk management strategy can be included in the software project plan, or the risk management steps can be organized into a separate *risk mitigation, monitoring, and management plan (RMMM)*.
- *The RMMM plan documents all work performed as part* of risk analysis and is used by the project manager as part of the overall project plan

Risk Information Sheets

- Alternative to RMMM plan in which each risk is documented individually.
- Often risk information sheets (RIS) are maintained using a database system.

- RIS components
  - risk id, date, probability, impact, description
  - refinement, mitigation/monitoring
  - management/contingency/trigger
  - status
  - originator, assigned staff member
- Risk monitoring is a project tracking activity with three primary objectives:

(1) to assess whether predicted risks do, in fact, occur;

(2) to ensure that risk aversion steps defined for the risk are being properly applied; and

(3) to collect information that can be used for future risk analysis. In many cases, the problems that occur during a project can be traced to more than one risk

Risk information sheet.
Source: [Wil97].

Risk information sheet			
Risk ID: P02-4-32	Date: 5/9/09	Prob: 80%	Impact: high

Description:
Only 70 percent of the software components scheduled for reuse will, in fact, be integrated into the application. The remaining functionality will have to be custom developed.

Refinement/context:
Subcondition 1: Certain reusable components were developed by a third party with no knowledge of internal design standards.
Subcondition 2: The design standard for component interfaces has not been solidified and may not conform to certain existing reusable components.
Subcondition 3: Certain reusable components have been implemented in a language that is not supported on the target environment.

Mitigation/monitoring:
1. Contact third party to determine conformance with design standards.
2. Press for interface standards completion; consider component structure when deciding on interface protocol.
3. Check to determine number of components in subcondition 3 category; check to determine if language support can be acquired.

Management/contingency plan/trigger:
RE computed to be $20,200. Allocate this amount within project contingency cost. Develop revised schedule assuming that 18 additional components will have to be custom built; allocate staff accordingly.
Trigger: Mitigation steps unproductive as of 7/1/09.

Current status:
5/12/09: Mitigation steps initiated.

Originator: D. Gagne	Assigned: B. Laster