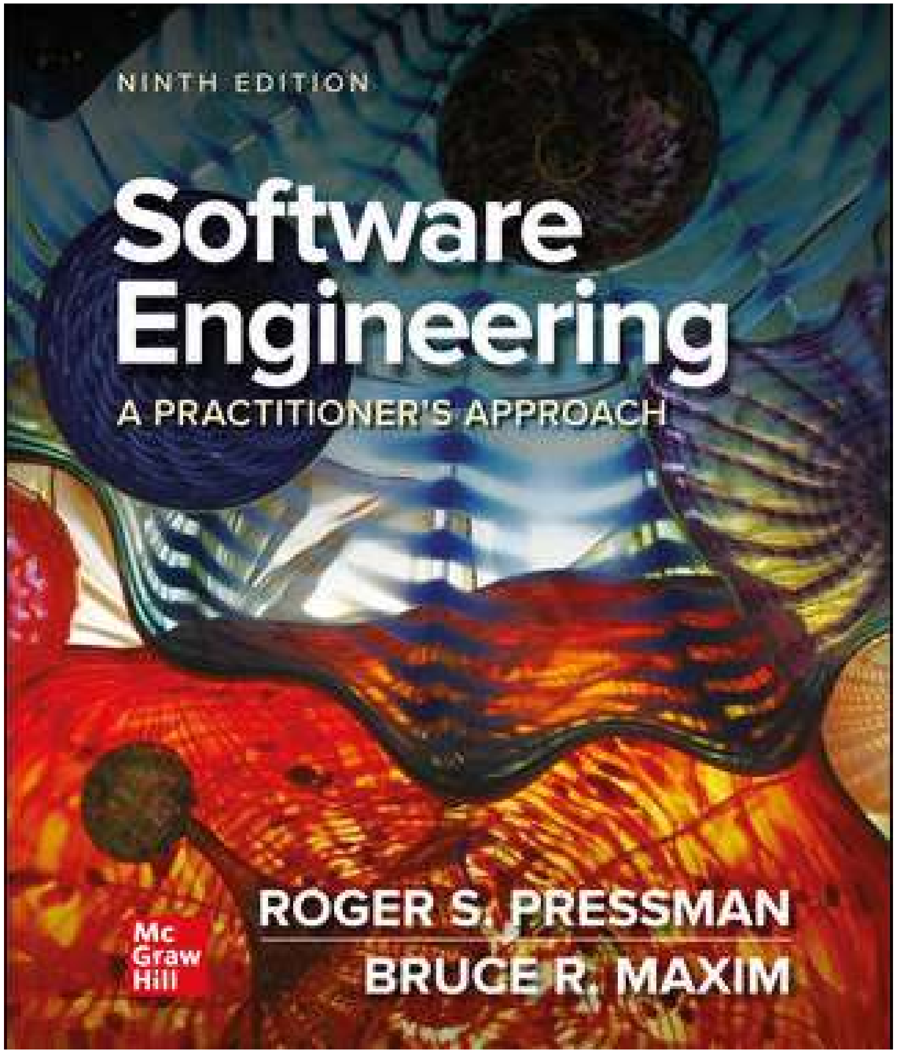
Because learning changes everything.®

Chapter 26

Risk Management

Part Four - Managing Software Projects



Reactive Risk Management

- Project team reacts to risks when they occur.
- Mitigation—plan for additional resources in anticipation of fire fighting.
- Fix on failure—resource are found and applied when the risk strikes.
- Crisis management—failure does not respond to applied resources and project is in jeopardy.

Proactive Risk Management

- Potential risks are identified, their probability and impact are assessed, and they are ranked by importance.
- Software team establishes a plan for managing risk.
- Primary objective is to avoid risk, but because not all risks can be avoided.
- Team works to develop a contingency plans that will enable it to respond in a controlled and effective manner.
- Proactive risk management is a software engineering tools that can be used to reduce technical debt.

Software Risks

- ***Project risks*** threaten the project plan.
- ***Technical risks*** threaten the quality and timeliness of the software to be produced.
- ***Business risks*** threaten the viability of the software to be built and often jeopardize the project or the product.
- ***Known risks*** are those that can be uncovered after careful evaluation project plan.
- ***Predictable risks*** are extrapolated from past project experience.
- ***Unpredictable risks*** can and do occur, but they are extremely difficult to identify in advance.

Risk Management Principles 1

Maintain a global perspective - view software risks within the context of system and the business problem.

Take a forward-looking view - think about the risks that may arise in the future; establish contingency plans.

Encourage open communication - if someone states a potential risk, don’t discount it.

Integrate - a consideration of risk must be integrated into the software process.

Risk Management Principles 2

Emphasize a continuous process - team must be vigilant throughout the software process, modifying identified risks as more information is known and adding new ones as better insight is achieved.

Develop a shared product vision - if all stakeholders share the same vision of the software, it likely that better risk identification and assessment will occur.

Encourage teamwork - the talents, skills and knowledge of all stakeholder should be pooled.

Risk Identification 1

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.

Customer characteristics - risks associated with the 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 by the development organization.

Risk Identification 2

- ***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 Project Risk 1

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 their customers?
4. Have customers been involved fully in the definition of requirements?
5. Do end-users have realistic expectations?
6. Is project scope stable?

Assessing Project Risk 2

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

- ***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.

Risk Projection (risk estimation)

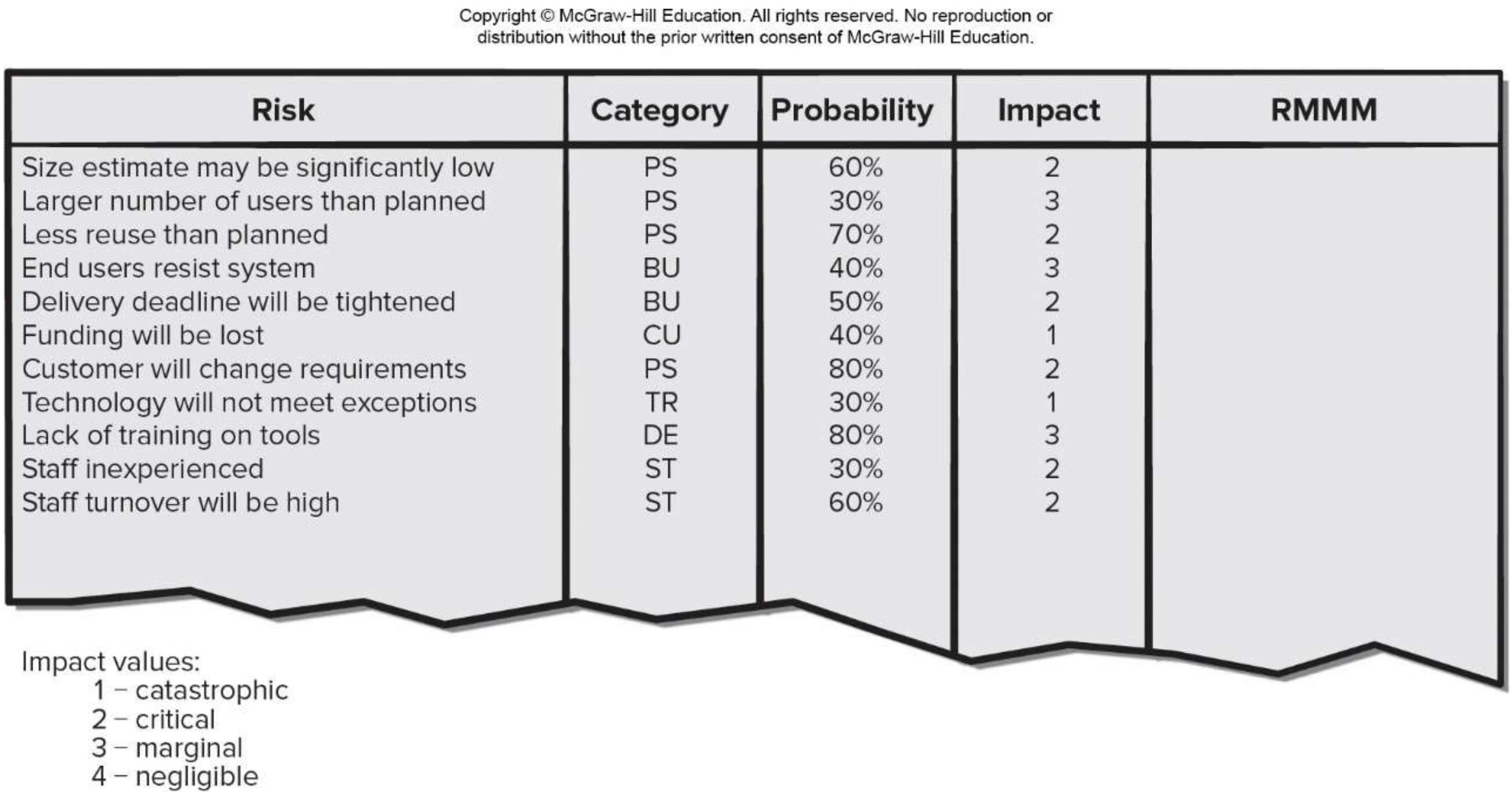
Risk projection attempts to rate each risk in two ways:

- Likelihood or probability that the risk is real.
- Consequences of the problems associated with the risk,

The are 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. Note the overall accuracy of the risk projection so that there will be no misunderstandings.

Risk Table



Risk	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 exceptions	TR	30%	1	
Lack of training on tools	DE	80%	3	
Staff inexperienced	ST	30%	2	
Staff turnover will be high	ST	60%	2	

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

Access the text alternative for slide images.

Building Risk Table

- Estimate the probability of occurrence.
- Estimate the impact on the project on a scale of 1 to 5, where,

  1 = low impact on project success

  5 = catastrophic impact on project success

- Sort the table by probability and impact.

Risk Impact (Exposure)

- The overall *risk exposure,* RE, is determined using the following relationship [Hal98]:

$$\mathbf{RE} = \boldsymbol{P} \times \boldsymbol{C}$$

where

P is the probability of occurrence for a risk, and

C is the cost to the project should the risk occur.

Risk Exposure Example

Risk identification. Only 70 percent of the software components scheduled for reuse will be used, the rest will have to be custom developed.

Risk probability. 80% (likely).

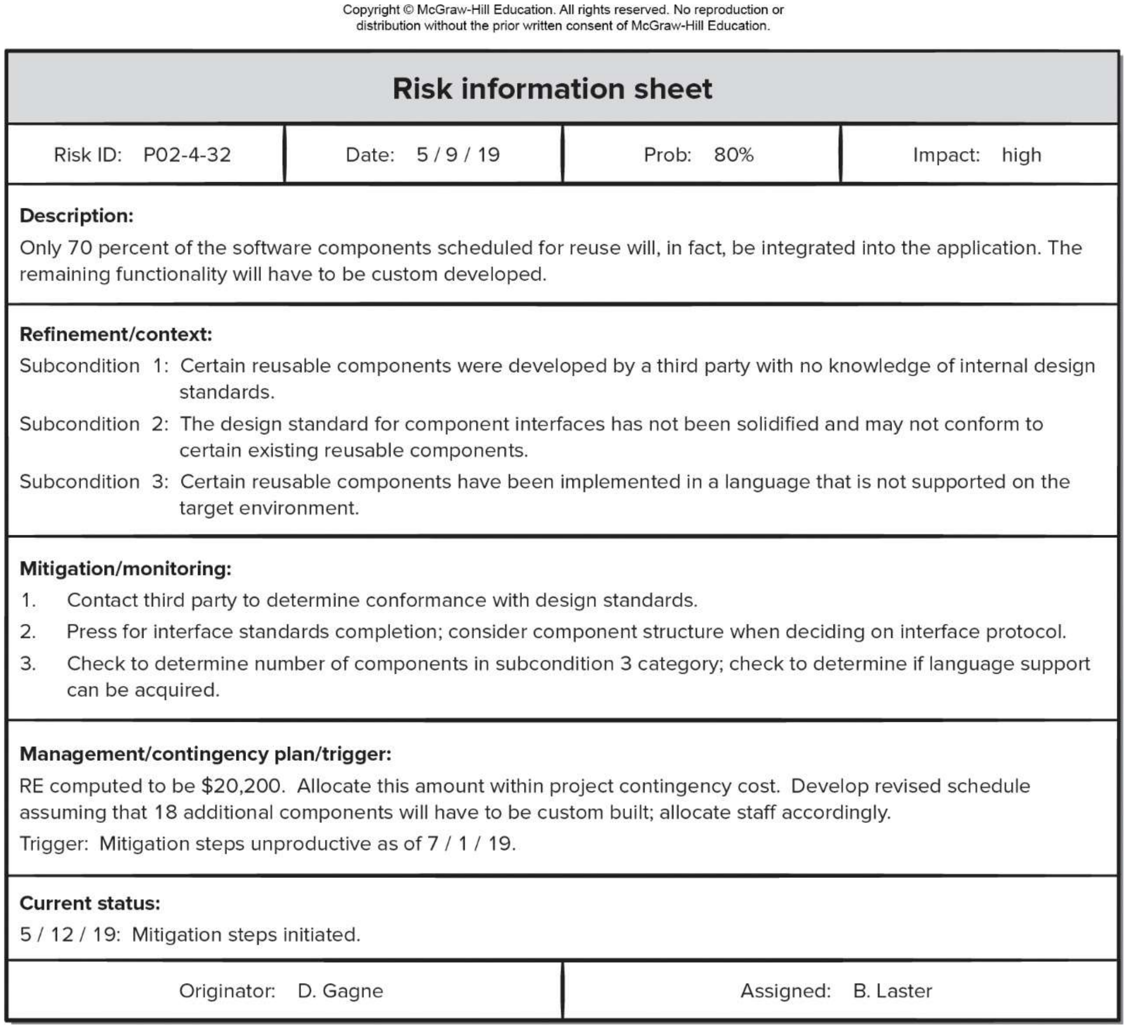
Risk impact. 60 reusable software components were planned. If only 70 percent can be used, 18 components would have to be developed from scratch. The average component is 100 LOC and the software engineering cost for each LOC is $14.00, the overall cost (impact) to develop the components is 18 × 100 × 14 = $25,200.

Risk exposure. $\mathrm{RE} = 0.80 \times 25{,}200 \square \$20{,}200.$

Risk Mitigation, Monitoring, and Management

- **Mitigation** - how can we avoid the risk?
- **Monitoring** - what factors can we track that will enable us to determine if the risk is becoming more or less likely?
- **Management** - what contingency plans do we have if the risk becomes a reality?

Risk Information Sheet



Risk information sheet

Risk ID: P02-4-32	Date: 5 / 9 / 19	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 / 19.

Current status:

5 / 12 / 19: Mitigation steps initiated.

Originator: D. Gagne	Assigned: B. Laster

Access the text alternative for slide images.

McGraw Hill Education

Because learning changes everything.®

www.mheducation.com