

PM World Today Tips & Techniques Risk Analysis and Prioritization

By Daniel Galorath

Risk management requires top-level management support, acknowledgment that risks are realities, and a commitment to identify and manage them. One discriminator of a successful organization or project is the use of risk management to anticipate potential negative conditions, problems, and realities. Ineffective projects are forced to react to problems; effective projects anticipate them.

Risk analysis

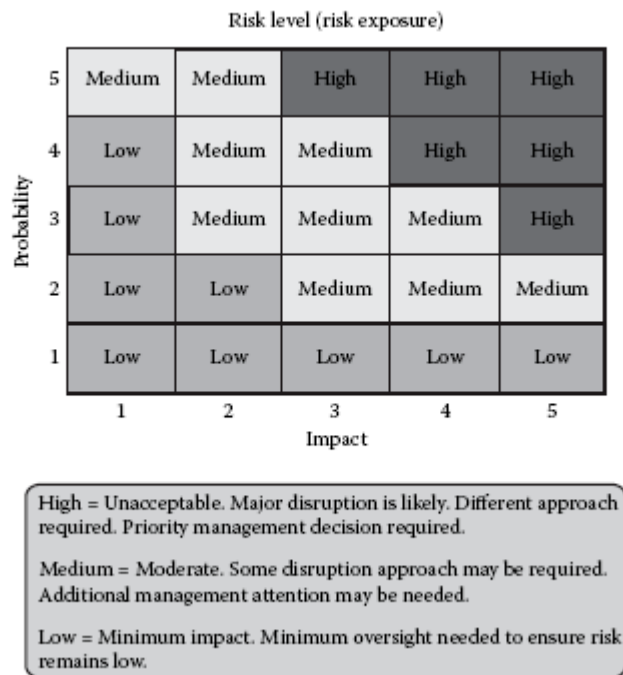


Figure 1: Risk grid.

Risk analysis includes evaluating risk areas to determine risk events, assigning likelihood and consequences to each risk event, and creating a risk index. Project components such as systems engineering and personnel should undergo risk analysis. Risk analysis comprises a number of activities:

- Evaluation of identified risk events and determination of possible outcomes
- Identification of critical variances from known best practices
- Determination of likelihoods that the events will occur
- Descriptions of possible consequences

Evaluation — Each identified risk can be rated against the program criteria and assigned a low, moderate, or high rating (See Figure 1).

Critical variance — For each risk event related to process, the analysis team must determine the variance of the process from known standards or best practices and rate it. As shown in Figure 1, the risk assessment process has five levels (1 through 5). No variance equals no risk.

Likelihood or probability — The likelihood that each risk will occur must also be determined. The subjective criteria are remote, unlikely, likely, highly likely, and near certainty. Zero likelihood of an event means no risk.

Consequence — For each risk identified, the following question must be answered: if the event occurs, what is the magnitude of the consequence? Consequence levels are numbered 1 through 5. A consequence is a multifaceted issue. Most often, four factors are considered when determining consequence: technical performance, schedule, cost, and impacts on other organizations. At least one of the four consequence areas needs to apply for risk to be present. If no adverse consequences are found in any areas, no risk exists.

Use of Metrics

Metrics provide managers with near real-time measures of project status and preclude the need for subjective decision making. They furnish quantitative planning goals and early risk warning triggers that require timely corrective program management action.

Two important elements of an effective metrics program are management support and a knowledgeable software support staff. Management support is required to ensure funding for the metrics effort and to encourage the use of metrics in decision making. A knowledgeable staff understands specific program and software system issues beyond the numbers in the raw metrics data and can translate crude metrics data into valuable information on program and system status.

Key management processes are implemented based on variances in predefined thresholds or trip wires. Metrics program preparation activities include:

- Identification of key metrics based on specific program decision requirements
- Tailoring selected metrics to track identified program risks
- Ensuring that metrics tie closely to the risk management plan
- Definition of metric thresholds
- Determination of a metrics review process and exit criteria

Use of Quantitative Triggers

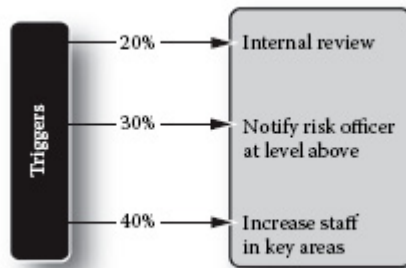


Figure 2: Sample risk triggers.

Each risk should have an indicator that can be tracked against quantitative triggers to determine when action should be taken. When a risk is characterized, a metric is selected to indicate the increase or decrease in risk probability. This metric should be based on underlying risk conditions providing a quantitative measure of the degree of a condition in a project environment, an attribute of a product, or any artifacts that describe it. For example, a metric such as number of tests planned versus number successfully completed over a given period can be used to determine the state of the testing process and the risks associated with failing to meet critical test schedules. Product-based metrics such as defects per thousand lines of code provide indications of the potential quality of a product and accurate predictions of the risks of rework. Figure 2 illustrates sample triggers.

The risk officer is responsible for tracking the predefined thresholds (established when the risk was defined and reviewed and approved by the risk owner) tied to specific management-approved quantitative thresholds. As these thresholds are crossed, predefined actions take place automatically. Threshold categories can include:

- Analysis level: conducting secondary analysis using secondary measures and indicators to determine cause of condition
- Report level: reporting risk condition to risk officers in affected organizations
- Action level: initiating staged mitigation based on likelihood of occurrence

Triggers are quantitative. On a frequent basis, a risk officer should evaluate risks in the higher categories against the assigned metrics to assess trigger status. When a trigger trips, the risk officer should take the assigned action.

Prioritize Risks

The output of this step is an updated risk index. From the risk index, a prioritized list of risks weighted by specific offsets to address overarching critical requirements is prepared. Priorities are determined by the severity of the consequences in areas such as safety, security, and reliability that could jeopardize successful deployment. Stand-alone risk tools, such as Microsoft Excel (used as a risk register), Active Risk Manager (ARM), the Risk Rating System, Risk Radar, Risk Matrix, RiskNav, Risk Radar, RiskTrak, Technical Risk Identification and Mitigation System (TRIMS), the Program Manager's WorkStation (PMWS), or risk tools integrated into broader, integrated project management tools such as PMTOOLBOX can be used to create and maintain the risk index. Keep in mind that the risk index is dynamic and it should reflect changes in likely impacts or probabilities.

Risks must be prioritized because no project has the resources to mitigate every risk. You therefore need to know which risks pose the greatest threat to a project and should receive priority. In addition, risks must be continuously prioritized because of changing circumstances or external forces that change the priorities of particular risks. The most important risks must be identified because most organizations can only actively mitigate a certain number of risks at a time. The risk officer should focus on actively tracking the highest priority and most likely risks and reporting to management those that may threaten project success.

A risk index can be used to track and prioritize risks based on criticality, the time available for action, their effects on safety, security, critical system operations, and other vital programmatic, operational, deployment, or enterprise factors. The risk index should include means of defining and assigning weights to predefined management factors to track changing priorities. Of course, some risks (designated opportunities) can generate desirable results. Such risks should be studied and implemented as appropriate.



Mr. Daniel Galorath



Daniel D. Galorath. has over 35 years of experience in the software industry where he has solved a variety of management, costing, systems, and software problems, and performed all aspects of software development and management. Mr. Galorath is founder and president of Galorath Incorporated, maker of the SEER® suite of estimation tools. Mr. Galorath is one of the principal developers of the SEER-SEM™ Software Estimation Model. Mr. Galorath completed his undergraduate work and MBA from California State Universities. He is a member of the International Society of Parametric Analysis (ISPA), Society of Cost Estimation and Analysis (SCEA), IEEE, the International Function Point Users Group (IFPUG), and the Association of Computing Machinery (ACM). He was honored with the Freiman Award, recognizing his long-term contributions to the field of parametric analysis. Mr. Galorath teaches courses in software cost, schedule, and risk analysis; software project management; software engineering; systems architecture, and other related topics. He has lectured internationally and is the author of many papers about software project management.

*Mr. Galorath can be reached at info@galorath.com.
His website is <http://www.galorath.com>.*