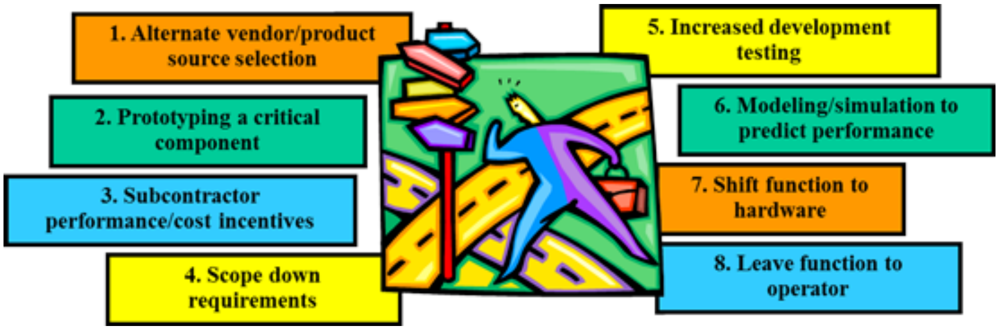
Risk Control

Planning

The first step in Risk Control is Risk Planning. Feasible options are identified for managing the risk; examples of risk management options include:



1. Alternate vendor/product source selection
2. Prototyping a critical component
3. Subcontractor performance/cost incentives
4. Scope down requirements
5. Increased development testing
6. Modeling/simulation to predict performance
7. Shift function to hardware
8. Leave function to operator

Note that some strategies are aimed at avoiding risk including (2) prototyping, (4) scope down requirements, and (6) modelling and simulation, while others are aimed at transferring risk including (3) subcontractor performance and cost incentives, (7) shift function to hardware, and (8) leave function to operator. You need to compare effectiveness and the cost of various options. One method is to apply the Risk Reduction Leverage (RRL), a quality for price ratio, which is defined as:

RRL = (Risk Exposure Before − Risk Exposure After) / Risk Reduction Cost

For example, assume the loss due to a software interface error is estimated at $100,000 and the probability of error being introduced is estimated at 0.3 or 30%. Then the Risk Exposure Before = $100,000 x (.3) = $30,000. Your research shows that there are three approaches for eliminating the error:

- An interface (I/F) checking tool that costs $5,000 and reduces the probability to 0.2 or 20%,
- An I/F testing approach that costs $15,000 and reduces the probability to 0.1 or 10%, or
- Reusing tried and proven interface software that costs $3,000 and reduces the probability to 0.1 or 10%

Using this data, we can solve the equations for the RRL as:

RRL = (Risk Exposure Before − Risk Exposure After) / Risk Reduction Cost

Option 1: RRL (checker) = [$30K − ($100K x (.2))] / $5K = 2

Option 2: RRL (testing) = [$30K − ($100K x (.1))] / $15K = 1.33

Option 3: RRL (reuse) = [$30K − ($100K x (.1))] / $3K = 6.66 (**Best Solution**)

Now with this quality for price data, specific action plans including the implementation criteria for success and/or action thresholds can be developed for the options. The risk management plans may be **contingency plans**, in case the failure occurs, or **mitigation plans**, to proactively reduce the likelihood that the risk will occur, or both.

Project management then reviews the risk management plans, refines them, and the appropriate approach is selected. The plans must answer the following questions for each risk item:

- Why? Answering risk item importance and relation to project objectives
- What? When? Answering risk resolution deliverables, milestones, and activities
- Who? Where? Answering responsibilities and organization
- How? Answering the approach
- How much? Answering budget, schedule, and key personnel

Resolution

Effective risk resolution often hinges on finding a creative solution to the specific risk. Thus the solution depends on the risk. There are eight (8) generic methods to resolve risks.

Avoid the risk	Do not do the risky activity: negotiate elimination of the requirement or hire a consultant (specialist) to do the risky work
Transfer the risk	Buy special purpose hardware rather than develop software or find a COTS solution
Buy information about the risk	Spend money to investigate the risk: prototype, bring in a consultant to evaluate your design, or establish an IR&D project well ahead of time
Eliminate the root cause of the risk	Analyze the risk to look for underlying causes using techniques such as Kaizen, Lean, or Six Sigma; and minimize or eliminate root causes
Assume the risk	If the consequences are small and the cost to avoid the risk is large, simply accept the consequences if the risk occurs
Publicize the risk	Notify upper management, marketing, and the customers about the risk and consequences:

	minimize the surprise if it occurs and get buy in for resolutions
Control the risk	Accept that the risk might occur and develop plans to handle it if it does
Remember the risk	Build a collection of risk management plans and resolutions for use on future projects

Monitoring

In Risk Monitoring, the software manager tracks the status of open and potential risk areas. Technical, cost, and schedule performance of each implemented risk plan is qualitatively assessed at internal periodic project management reviews to:

- Verify outcomes were as envisioned,
- Identify opportunities for refinement of risk plan, and
- Provide feedback to those making programmatic decisions

Customer personnel are apprised of each risk area at the monthly progress reviews and major technical reviews. Risk monitoring techniques include collecting metrics, reviewing risks, and tracking the top ten risks as in the chart.

Top 10 Risk Tracking: Third Week

This week	Last week	Weeks on list	Risk Item	Risk resolution progress
1	1	3	Scoping product to fit required schedule	Working session with customer
2	4	3	Resolving subcontractor OS I/F uncertainties	New changes proposed by subcontractor
3	2	3	Algorithm boundary checking	Prototype on schedule
4	6	2	Workstation availability	Delay in delivery reported
5	5	3	User I/F definition	User study complete, awaiting prototype
6	3	3	Staffing lead developer for control software	Candidate identified, offer extended
7	-	1	Reusable text-formatting software uncertainties	Evaluation underway
8	7	3	Application performance risk	Benchmark to complete this week

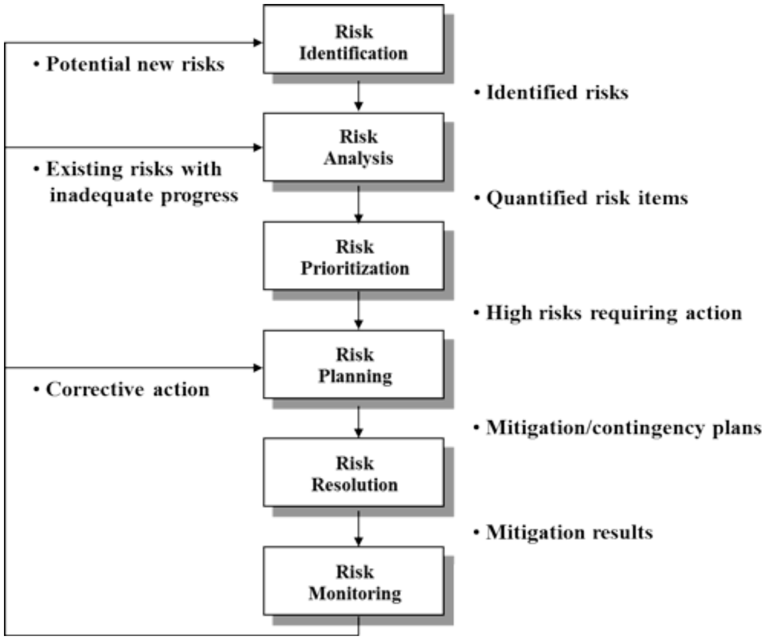
This week	Last week	Weeks on list	Risk Item	Risk resolution progress
9	9	2	CM procedures for incremental development	CM procedures identified, SEPG tailoring
10	8	2	Staffing programmer for control software	Programmer accepted offer; reports on 08/01

Risk Management Panel

Large programs/projects often create a Risk Management Panel (RMP) which should be documented in the appropriate Project Instruction. The purpose of the RMP is to provide accurate current data and multi-disciplinary inputs to Program/Project Management to make informed decisions about managing the risks. The panel typically includes the Deputy Program/Project Manager who acts as the RMP chair, the Chief System Engineer, the Quality Assurance Office Manager, the Software Development Manager, the Operations and Maintenance Manager, the Independent Test Office Manager, the Commercial off the Shelf (COTS) Procurement Manager and a Customer Representative. The RMP acts as a reviewing body at each checkpoint of the risk management process: identification, analysis, prioritization, planning, resolution, and monitoring. The outputs of the RMP are approvals and redirections for each checkpoint in the risk management process. Decisions of the RMP are reported to other relevant management forums such as the Project Review, Configuration Control Boards (CCBs), and monthly status meetings. Issues or feedback from these forums may require the RMP to deliberate further.

Risk Management Summary

In summary, the risk engineering process is defined as:



Organizations have a choice. They can take a risk based management approach or they can choose not to invest in reducing risk. There are positives and negatives to both approaches, but the software industry's current stance is the risk based management approach is better and cheaper in the long run.

	Risk-based Management	**No Risk Investment**
Orientation	"Realistically oriented", risks set up boundary conditions, foresee and contain possible failure, proactive not reactive	"Success oriented", "can do" attitude, minimize thought of possible failure, leave little extra to handle contingencies
Problems	• Risk analysis may be inaccurate, underestimation can cause major problems, overestimation can lead to excessive mitigation effort • Takes resources away from other tasks • Effects of risk engineering hard to measure: total avoidance removes proof that cost was justified	• Reactive • No planning • No options for management consideration
Benefits	• Better perception of risks & options for management: reduction in project exposure to risk • Improved credibility of plans • More proactive	• Funds are not diverted to risk mitigation activities • "Can do" attitude
Example: Think of the car industry in the 1990s	U.S. software development • Change to meet user-requested requirements, independent of component availability • More functional, higher cost, less reliable • Greater need for risk-based management approach	Japanese software development • Change functionality in response to availability of components • Less functional yet more reliable • Less need for risk management