

Program Risk Management

Russ Martinelli - Intel Corporation

Jim Waddell - Tektronix, Inc.

Introduction

Developing high technology products is risky business by nature, especially if a company wants to achieve or maintain competitive leadership. Technology risk alone is a constant, and risk-taking can provide a means to gaining competitive advantage. However, risk-taking does not mean taking chances. It involves understanding the risk/reward ratio, then managing the risks, or uncertainties, that are involved in each product development effort. Failure to do so can lead to substantial loss for the enterprise, including the possibility that the product will fail to achieve the business results intended (such as increased revenue, increased market share, or technological superiority).

On product development programs at Intel Corporation and Tektronix, Inc., the program manager is responsible for managing risk on the program, as she/he is responsible for the success of the product and business objectives driving the development effort. In order to be successful, the program manager must manage risk across multiple, interdependent projects. This requires a different perspective and techniques than managing risk on a single project; if one of the projects fails the entire program will likely fail.

In this article we will explain why risk management is a critical program management process at Intel and Tektronix, demonstrate the difference between program risk and project risk, and provide some key learnings for managing risk from personal experiences in managing product development programs in the high-technology industry.

Why Risk Management?

As stated previously, various levels of risk-taking are necessary in the high-technology industry if a company is intent on being a market leader. Products that don't push the risk envelop probably aren't worth the development investment - unless the company is a market follower. Market leaders understand they must run directly toward, instead of away from risk in order to put distance between themselves and their competitors. For this reason, risk-taking is a core value at Intel, and an expected behavior on product development efforts at Tektronix. Good risk management practice by the program manager and his/her team makes this aggressive risk-taking possible by bounding the level of uncertainty through risk mitigation plans and actions.

Risk management enables informed risk-based decisions. Program managers, project managers, general managers and executive managers make multiple decisions each day as a primary function of their jobs. Decisions may range from mundane and tactical to business critical and

strategic in nature. Having access to current factual information is critical to making a good decision. In addition, knowledge of potential risks can improve the decision process by allowing the decision maker to weigh potential alternatives or trade-offs in order to maximize the reward/risk ratio.

Finally, risk management brings a level of predictability to a very dynamic environment that is characteristic of the high-technology industry. By understanding and bounding the uncertainties on a program, the program manager is able to manage in a proactive versus reactive manner. Without good risk management practices, the program manager will be forced into crisis management activities as problem after problem presents itself. The program manager is then forced to constantly react to the problem of the day (or hour). Risk management allows the program manager to identify potential problems *before* they occur, and put corrective action in place to avoid or lessen the impact of the risk. Ultimately this behavior allows the program team to accelerate through the product development pipeline at a much greater rate.

Risk management is similar to the function of the braking system on an automobile. At first, we think the primary reason for the braking system is to slow the automobile down so the occupants get from starting point to destination as safely as possible. However, braking systems also allow us to travel from start to destination as *quickly* as possible without colliding into obstacles impeding our roadway. Similarly, good risk management practices allow program teams to move from product concept to product launch as quickly as possible by removing potential barriers well ahead of the point where they become impediments to time-to-profit goals.

Relationship between Program and Project Risk

In the second article of this series titled "Program and Project Management: Understanding the Differences"¹, we discussed what it means to manage multiple, interdependent projects in the context of delivering the "whole product". Figure 1 demonstrates the whole product concept and how our companies differentiate between program and project management. In short, the program manager is responsible for the development of the whole product (in this example, a personal computer) as well as the business objectives that the product is meant to achieve, while the project managers are responsible for the successful delivery of their respective element of the product.

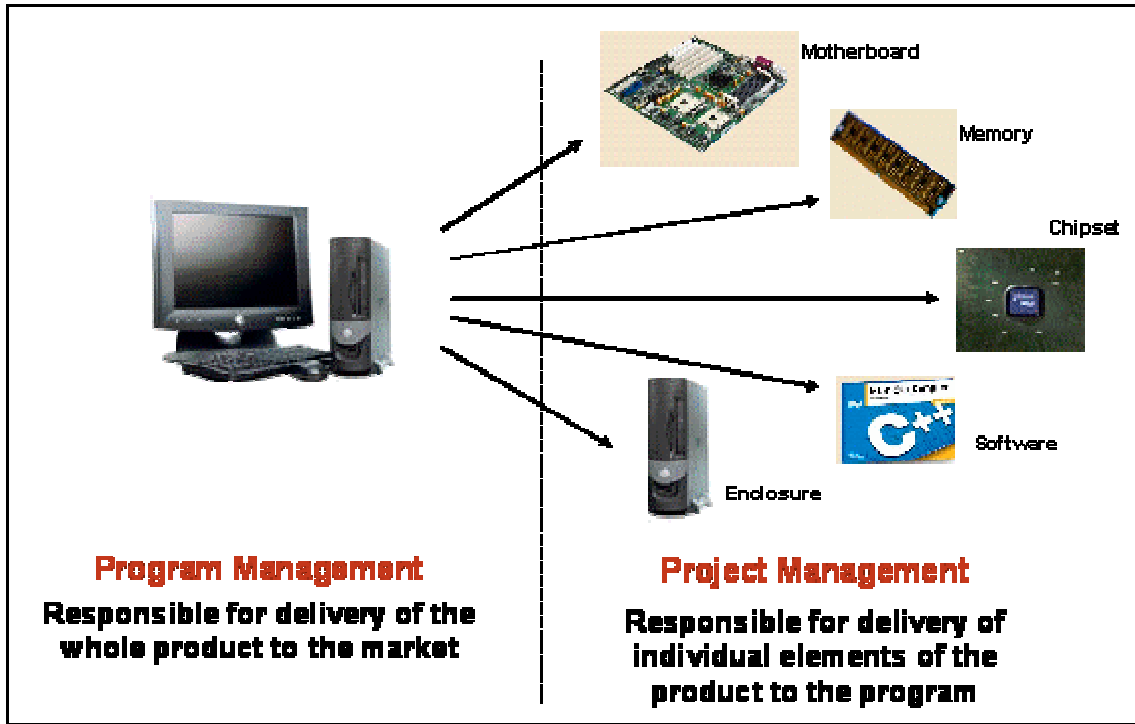


Figure 1: Program Management, Project Management, and the Whole Product

With respect to managing risk, the project managers are responsible for any risk that can affect the successful delivery of their element of the product. The program manager is responsible for any risk that can affect the success of the product, up to and including the business risk. In addition, she/he is responsible for the many risks involved in the interdependencies between the projects (for example, the delivery of a software release to support a motherboard milestone). Figure 2 illustrates a portion of a product development team organized in a matrix structure which is utilized by both Intel and Tektronix.

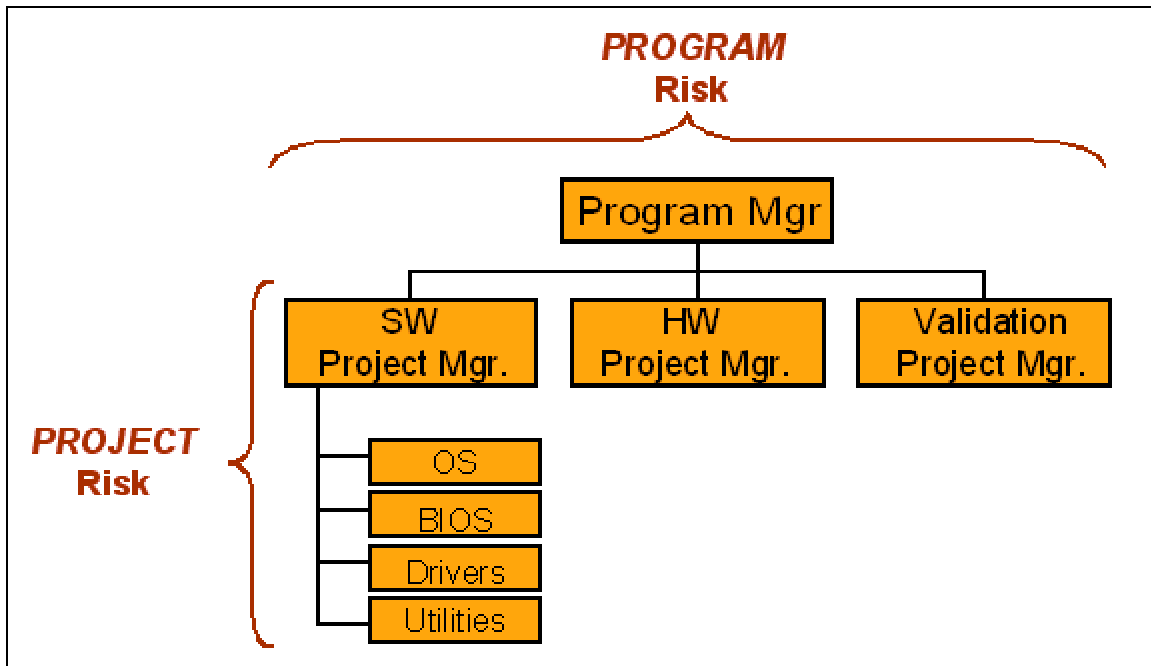


Figure 2: Program and Project Risk Management

The program consists of three interdependent functional project teams; hardware development, software development, and validation. The project manager for each project team is responsible for identifying, analyzing, tracking and responding to all risks that may prevent his/her team from delivering their element of the product on time, within budget, and with specified features and quality levels. The program manager, in conjunction with the project managers, is responsible for identifying the pertinent program level risks that span across the projects and are related to the interdependencies between the projects.

In order to effectively manage risks that may impact the success of the product and business objectives, the program success criteria must be clearly identified. The Program Strike Zone is an effective tool for identifying the program and business success criteria. We introduced the Program Strike Zone in the first article of this series; "The Program Strike Zone: Beyond the Bounding Box"². Figure 3 shows an example of an abbreviated Program Strike Zone.










Program Strike Zone			
Critical Success Factors	Strike Zone		Status
	Target	Threshold	
Value Proposition: <ul style="list-style-type: none"> ▪ Increase market share in product segment <ul style="list-style-type: none"> ▪ Order growth within 6 months of introduction ▪ Market share increase one year after introduction 	10%	5%	
	5%	0%	
Schedule: <ul style="list-style-type: none"> ▪ Product Proposal Approval ▪ Program Plan Approval ▪ Initial "Power On" ▪ Product Launch to Market 	3/15/2003	3/30/2003	
	6/15/2003	6/30/2003	
	10/01/2003	11/01/2003	
	4/15/2004	5/30/2004	
Financials <ul style="list-style-type: none"> ▪ Program Budget ▪ Product Cost ▪ Profitability Index 	100% of plan	105% of plan	
	\$8500	\$8900	
	2.0	1.8	

Figure 3: Example Program Strike Zone

The program level risk analysis should be focused on identifying and removing potential barriers to achievement of the critical success factors identified in the Program Strike Zone.

A recent product development program at Tektronix, illustrates this point. One of the critical success factors noted in Figure 4 relates to Schedule "Initial Power On". The status of this critical factor is shown as "RED" which indicates that the program schedule critical path was in jeopardy of being violated because the "Power On" milestone would not be achieved by the threshold date of November 1. The potential schedule delay was due to a known risk identified by the hardware "project" team - uncertainty that an externally supplied component could meet stringent design specification requirements. This project risk resulted in a "program" level risk due to the potential for adverse impact to the product launch date and profitability index objectives for the program. Resolution was achieved through risk planning and mitigation efforts that enabled the hardware project team to utilize an alternative supplier's component that met the required design specification. This effort insured that both the "project" risk and "program" risk were successfully mitigated, the Power On milestone was accomplished, and the team was again on track to achieve the program objectives.

Key Risk Management Learnings

There are many things we've learned, and continue to learn about managing risks on high-technology product development programs. But the top three learnings to date would have to be: 1) risk management is not an exact science, 2) risk management is an iterative process and not a one time event, and 3) the concept of organizational risk tolerance must be understood and managed.

Learning one: When we say risk management is not an exact science, we're referring to the fact that the program manager is always working with subjective risk information. After all, we're trying to predict what *may* happen in the future, not analyzing what *did* happen in the past. Therefore two things come in to play. First, it's not possible to identify and manage every risk " there will always be problems that have to be contended with (problems are prior risks that weren't identified). Second, don't get lost in trying to *quantify* every risk. This last statement can be contentious because so many people have created quantification models and algorithms to help with managing risk. These models and algorithms are great tools for people in some industries, but in high-technology product development, there is such a high number of risk events that a program manager could spend all of his/her time running the probability projections, and end up neglecting the other aspects of managing the program. For us, it doesn't matter if a particular risk event has a 0.6 or 0.8 probability of occurrence, in either case it's a high risk and a response plan has to be generated and acted upon.

Learning two: Risk management should not be viewed as a one-time event. It's not uncommon for a program team to present the risks associated with the program at each of the phase transition or gate meetings, then set the information aside until the next gate review. Practicing risk identification without practicing risk management is a fatal flaw. Risk management must be practiced consistently during all phases of the product life cycle. Figure 4 illustrates the various types of risk associated with each phase of the life cycle.

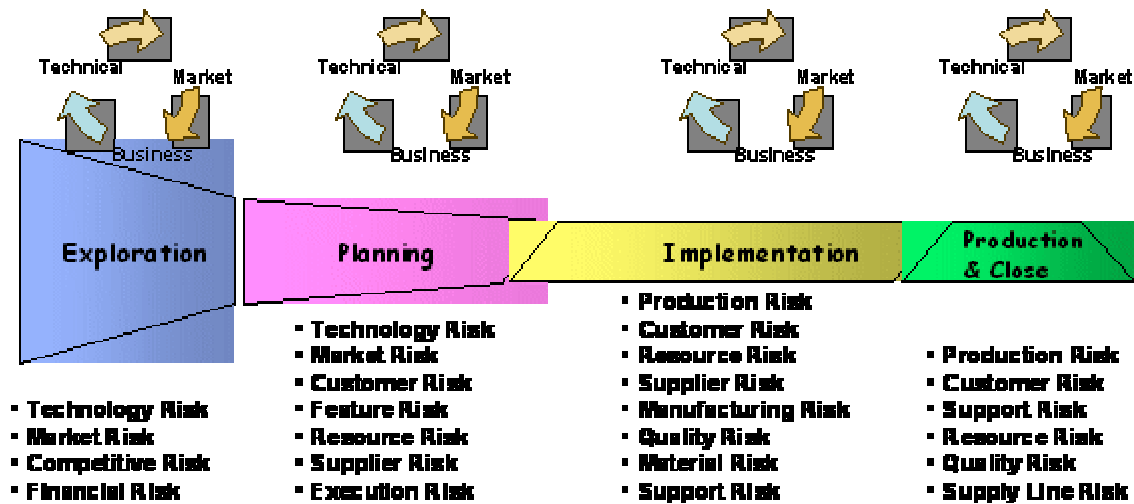


Figure 4: The Iterative Nature of Risk

Every organization is unique, but it's best that the program core team and project teams evaluate risk every two to four weeks to insure current risks are being worked and to identify new risks. The earlier a risk is identified, the more time the team has to respond to it.

Learning three: Risk tolerance is the level of risk an organization is willing to accept in order to achieve its business objectives. Every individual and every organization has a different level of risk tolerance, with corporate culture and values being a primary driver behind acceptable tolerance levels. Figure 5 demonstrates risk tolerance as a continuum, from complete risk

avoidance on one end to total disregard of the consequences of risk on the other.

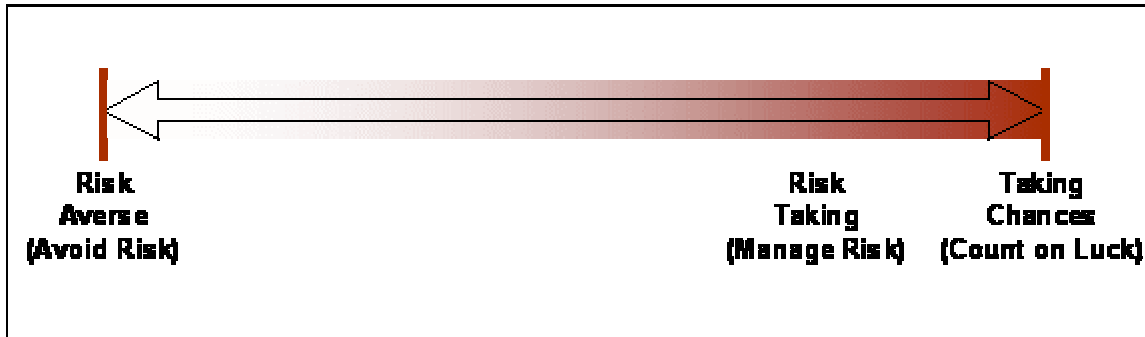


Figure 5: The Risk Tolerance Continuum

For example, companies in the aerospace and defense industries have a very low tolerance for risk due to customer requirements and product use conditions. Products in these industries must have very low failure rates; therefore companies developing the products must adopt a low risk tolerance approach. As a result, products have redundancies built in and many risk avoidance steps are included in the program plan. By contrast, in the commercial high-technology industry, risk-taking is necessary to attain and maintain a competitive advantage through the use of rapidly changing technologies. As a result, products have few built-in redundancies and program plans must incorporate schedule and budget buffers to allow for risk management actions.

So how does this affect the program manager? First, the program manager must understand the individual risk tolerance of key individuals involved with the program; such as the sponsor, functional managers, project managers, and the extended program team. For example, one team member may go into crisis mode when a potential problem is discovered, while a second team member may just say "we have it under control". The differences in reaction demonstrates that the two individuals have varying levels of risk tolerance - the program manager must flush out the risk tolerance of each individual to clearly understand if a risk may jeopardize his/her program. Second, understanding tolerance levels provides excellent guidance for how much risk a program manager can assume, and what response actions (avoidance, acceptance, mitigation, transfer) are accepted as preferred options within the organization. Unfortunately, there is no mathematical formula, no statistical calculation, and no model to help manage the various levels of risk tolerance the program manager will encounter - that's why we refer to this as the "art" of risk management. Experience, understanding the corporate culture, and developing personal interaction with the people on and surrounding the program are the best methods for learning to manage organizational risk tolerance.

Conclusion

Program risk management is an essential process in management of product development efforts in the high-technology industry for three primary reasons: 1) risk-taking is necessary to gain or maintain market leadership, 2) risk management improves the decision making process by allowing the decision maker to evaluate trade-offs to maximize the reward/risk ratio, and 3) it brings a level of predictability to an ever-changing environment. The program manager is

responsible for managing risk on the program, as she/he is responsible for the success of the product and business objectives driving the development effort. In order to be successful, the program manager must manage risk across multiple, interdependent projects, and as this article shows, this requires a different perspective and techniques than managing risk on a single project.

References

¹ [PM World Today](#), Volume VI Issue 3, May-June 2004

² [PM World Today](#), Volume VI Issue 1, March-April 2004

Copyright 2004 by Russ Martinelli and Jim Waddell