SEER by GALORATH

# Searching for Project Risk Management

By

Evin J. Stump P.E.

## Introduction

Project risk management has been an arcane subject that now is now a topic of increasing interest. I believe there are three reasons for this:

- Customers are more demanding
- Competition is increasing
- Project are more complex

Project managers are more than ever concerned that goals be met in accordance with a competitive plan. They don't want surprises. They want to understand their risks and take whatever measures they can to abate them.

The name of the activity that pursues these objectives is project risk management (PRM). But a project manager looking to implement PRM will find a maze of conflicting writings and opinions. Much of the literature on PRM is written by statisticians, or by the statistically adept. Often it advocates use of methods that are incomprehensible to the average project manager, or even to the average project engineer. Moreover, these methods tend to focus on the possibility of variation in the cost or schedule of individual tasks in the project for vague and unspecified reasons, then worry about (or ignore) the troublesome and complex statistical problem of correlation between tasks.[1] Other parts of the literature say that statistical methods are of no value because the data are suspect – garbage in, garbage out. What is a project manager to believe?

[1] I call this "the sky is falling" approach to risk analysis. Another name I use is the "angst" approach. When attempting to analyze risks in this way, the risk analyst must focus separately on each task and ask, considering all of the things that could go wrong in this task, what is the effect likely to be? This is a question the human brain is ill equipped to handle. Sensing this, many project managers shy away from this kind of risk analysis exercise.

I would like to say that he or she should believe what I write here, and I hope that will be the case. If I am believed, however, I trust it will be because what I have to say makes good sense, not because I, under color of authority, have said it.

Due to the limited scope of this article, what I will have to say will fall far short of a complete tutorial on PRM. But I do hope to provide satisfactory answers to some of the most frequently asked questions. The questions I will address are:

- What is the relationship between goals, plans, and PRM?
- What exactly is PRM?
- How can risks be efficiently classified and organized?
- How should risks be managed?

## Goals, plans, and PRM

A project is a goal oriented activity that employs resources over a period of time. Hopefully, this is done in the context of a well crafted plan. It is necessary to point out (because it is so often forgotten) that if the goals are not crisply defined, or if the plan is sloppy, it is not much use to talk about PRM. If you don't know where you want to go, or what road you want to take to get there, whatever happens can hardly be called risk. It is simply what happens.

If you are standing in the middle of a golf course in a lightening storm and have a death wish, there is a *probability* but no *risk* that you will be struck by lightening. But if you very much want to finish your MBA program and marry your childhood sweetheart and live the American Dream, you are well advised to go to shelter, because you are at risk.

As a beginning for PRM, then, I say there must be a well crafted plan. What do I mean by well crafted? That depends on the circumstances. If you are challenged with the task of assembling a new bicycle on a child's birthday, then your plan probably is the instructions for assembly that came with the bicycle. If you are mechanically adept, have good tools, and have assembled other bicycles recently, it may not much matter if the instructions for assembly are poorly done, or even if they are written in Swahili. You will get it done smartly. But if you are a poet

with no mechanical aptitude, have only a pair of pliers and a Swiss army knife, it helps if the instructions are perfectly written and beautifully illustrated.

So, what I mean by a well crafted plan is that the plan is appropriate to the skill level of the project team. Project teams that stay together for project after project, doing similar work in each project, have a shared experience that makes it unnecessary to provide more than a planning framework.[2] They will instinctively fill in those details.

The recent trend has been to assemble teams from across the enterprise, do the project, and then disband them, sending the members to other teams. On the plus side, the people are typically the best people that can be found, but on the minus side, the people have not worked together before and may not share a common gestalt. For them it is necessary to have a thorough plan that includes task lists and definitions, budgets, schedule network, and importantly, efficient business processes and infrastructure that makes things run smoothly and minimizes the distractions that prevent getting on with project business.

Moreover, the team must be fully oriented to the nature of the plan and the processes and infrastructure that are available. If the project proposal was done by one group, and a new group is doing the implementation, it is very easy for organizational memory to be lost.

As preconditions for thinking about PRM, then, I tentatively suggest the following:

- Crisply defined project goals
- A well crafted project plan.

In addition to being well crafted, it helps if the plan can also be made what I call robust. At a minimum level, a robust plan is like a car with good shock absorbers, as opposed to one with worn shocks. It absorbs the impacts from potholes and most bumps, and passengers hardly feel them. At this level,

[2] A prime example is the famous Lockheed-Martin skunk works that has worked together to build some fantastic airplanes over the years. Many other companies have such teams, though most are less well known.

robustness means cross-trained project team members, flexible resources, and good team morale.

At a maximum level, a good analogy is the reserve parachute. If your main chute doesn't open, you deploy the reserve. At this level, robustness means management reserves, and standby plans. An example of a standby plan: You build an apartment house, and find that there are insufficient tenants for it to be profitable. But because of location and carefully thought out architectural features, you are able to quickly and cheaply convert it to a strip mall, and as a strip mall, it is a roaring success. Or, you build a space launch facility and hope for business from governments wishing to launch satellites. But there aren't enough government buyers of launch services, so you go after commercial satellite launches as well.

So now I slightly amend the preconditions to read as follows:

- Crisply defined project goals
- A well crafted project plan as robust as you are able to make it.

We are now positioned to first describe, and then define, project risk.

## Project risk and risk drivers

Suppose we have goals and a plan. We know where we want to go and how we want to get there. In principle, we can push the button to start our plan in motion, and it will call up resources in accordance with the built-in time table. The resources will do the work, and when we are finished, we will have the desired product, system, infrastructure, or whatever it is we wanted.

In principle. In the real world, outcomes can be more complicated.

It is important to keep in mind that a project is an agent of change. While Nature itself is in the midst of continuing, gradual change, projects accelerate change, and send it in new directions. They try to move from the existing state of Nature to a new state that Nature would most likely never reach on its own. If we fully know the existing state of Nature, we potentially can plan and flawlessly execute a transition to a new state of nature without a hitch.

The problem is, we don't fully know the state of Nature, and none of us will live long enough to find it out. Even if we could, human beings are components of projects, and they are "wild cards" whose actions cannot be fully predicted. So if anyone is ever to do a project, they must proceed in spite of some lack of knowledge as to how things are, and consequently, how they will be. Only when we are executing the project will we discover how things truly are. Sometimes these realities will be of little consequence, but often they can have serious impacts on the project plan. These impacts are not necessarily bad, although omnipresent badness is a common (and unwarranted) assumption in most risk analysis. The six o'clock news notwithstanding, not everything that happens in the world is a disaster.

We can now define the notion of a "risk driver," and that will lead us to a simple definition of risk.

**Risk Drivers and Risk**

**A risk driver is an independently acting force that may move a project away from its plan.**

**Risk is the presence of risk drivers.**

Except for the word "independently," which I will explain shortly, the above definitions are simple and clear. The risk driver view moves us away from the entanglements of the angst approach, with its unfocused look at individual tasks and all possible things that can go wrong with each one of them, and interactions between tasks, and leads us instead to an issues based view of the project as a whole.

With the risk driver view, we can focus sharply on the plan itself. We can look at the changes it proposes to make to the state of Nature, and whether we are bringing to bear the appropriate level and mix of resources to accomplish them in the time allotted. We can look at the possibility that the resources we plan to use may not be available at the time we need them. We can look at possible changes in the state of Nature other than the ones we propose to make to analyze their potential impact on the plan. We can ask which specific tasks are impacted by each driver. We can ask how likely it is that each driver will act, and what impacts it may have. In short, we study each risk driver separately instead of studying each task separately. We decide which tasks are impacted by each driver, and we add, in simple arithmetic fashion, all of the impacts on each task. This is an enormous simplification.

In the risk driver view of risks, our project is like an army tank making its way through hostile forces to reach an objective. There are tank traps, land mines, and people shooting at it (risk drivers). Instead of focusing separately on what might happen to the engine, or to the cannon, or to the treads, we look at the threats (risk drivers). If we encounter a land mine, it can impact the treads but not the engine. If we encounter a machine gun, it can damage the communications system, but cannot penetrate the armor, and so on. To enhance the likelihood that the tank will reach its objective, we analyze the defenses available to the tank, and the power of the forces arrayed against it. We attempt to adjust the path of the tank to avoid high risk areas, or to roll quickly through them. We provide infantry and artillery support. We use cover and concealment. We use surprise.

So with PRM. We look at the strength of the project plan, and the obstacles that it may encounter. Considering the risk drivers that are "out there," we try to adjust the plan to make it more robust. These adjustments may include actions we might not have taken in the absence of a PRM activity. For example, they

may include stronger contract terms and conditions, extra engineering studies, more testing, use of experts, and purchase of insurance.

Now I return to the matter of the word "independently" that I used in the definition of a risk driver. Briefly, an independent force is one that does not act in concert with another force. This is a simple idea, but I find that it is not always readily grasped. It is perhaps best understood through an example of two "drivers" that are not independent, which I will now provide. Consider the following hypothetical risk drivers:

- Driver #1: Ajax Co. may not be able to enter the next phase of the competition.
- Driver #2: If Ajax Co. is not able to enter the next phase of the competition; we may be able to license their technology.

These do not meet the criteria for independence, because the first triggers the second. The appropriate view is that the first is a root cause driver (assuming a more fundamental cause cannot be identified), while the second is merely one of its possible outcomes. An outcome is simply one of the things that can happen to the plan due to the action of the driver, and is not itself a driver. So in identifying drivers, we need to look for root causes, and not confuse drivers with their outcomes.

With regard to the Ajax Co. example above, here is how it might be recast into a single risk driver with multiple outcomes:

- Driver: Ajax Co. may not be able to enter into the next phase of the competition.
    - Outcome #1: They will be able to, and we will have to develop our own technology
    - Outcome #2: They will not be able to, and we can license their technology.
    - Outcome #3: They will not be able to, but they will not license their technology.

There are three reasons for keeping risk drivers independent. The first is that we can arithmetically add the impacts of independent drivers, greatly simplifying

analysis. The second is that keeping them independent allows us to assign clear lines of responsibility for managing them. The third is that we eliminate concerns about a complex statistical problem known as correlation, which we have already mentioned.

Our next concern will be with the classification and organization of risk drivers so we can better deal with them.

## Classification and organization of risk drivers

When people are asked about the kinds of risk a project might experience, a frequent answer is technical, schedule, and cost. Some will add that there is really another category called programmatic, which has to do with administrative matters, terms and conditions, and so on. Some who have given thought to the subject will recite a list, citing such terms as legal, logistical, manufacturability, regulatory, inflation, currency exchange, overhead rates, productivity, inexperience, and so forth. To deal effectively with risk drivers, it behooves us to keep our classifications short and simple, while still accounting for reality. I will do this by the novel trick of classifying impacts, not causes.

The number of cause types is legion, but the number of impact types of practical interest is small. While you may quibble that there are other concerns, most projects are interested in just three impact types:

- Cash flow
- Duration
- Performance.

Let's look briefly at each of these to be sure we have a common understanding. Almost all projects are concerned with costs, and many have revenue concerns as well, so we need an expression that covers both. The expression that does that job is cash flow. Net cash flow, the difference between revenues and cost, is approximately synonymous with profit. A cash flow driver is one whose impact is principally on cash flow, either cost or revenue, or both. It has little or no impact on duration or performance. Typical examples are cash flow estimating errors, and all of the following that are not adequately accounted for in the plan: inflation, foreign exchange rates, overhead rates, taxes, and the like.

Duration refers to the span of time occupied by the project, or by any significant part of it that may be of interest. I believe the expression "duration driver" is more precise than the expression "schedule driver," so I have chosen the former. A duration driver is one whose impact is principally on duration (span of time) of the project work. It has little or no impact on cash flow or performance. Typical examples are duration estimating errors and all of the following that are not accounted for in the plan and that do not impact cash flow: logistic delays, bad weather, administrative delays, approval delays, and various kinds of project interference.

It is useful to define cash flow and duration drivers because they exist and are relatively easy to deal with, but we need another class of drivers that are more complex, because many drivers will impact both duration and cash flow and not just one task but many. I call these *performance* drivers. Many technology, programmatic, and logistic risks are of this type.

In analyzing project risks, we want not only to look at the impacts of individual drivers, but also at the aggregated effects. Readers having some knowledge of statistics may know that that the average (mean) impact of a set of drivers is equal to the sum of the average impacts of the individual drivers. But usually we want to know more about the aggregated effect than just the average. We also want some idea of a realistic range of impacts, and the likelihood that aggregated impacts will be above or below certain thresholds of interest.

To obtain this kind of information, the tool of choice is usually Monte Carlo simulation. A deep discussion of this subject is well beyond the scope of this paper. Suffice to say that Monte Carlo simulation requires the building of a logical model of the project, most of which is likely to b e done anyway, even if risk management is not contemplated. By repeatedly sampling of this model using a computer, the simulation builds histograms from which useful information can be inferred.

## Risk management

The classical steps of PRM are:

- Identify
- Assess
- Mitigate

These have withstood the test of time, and are valid.

It is often said that the reason we do the first two is so we can do the third. The goal is to mitigate risks when we can. While this is certainly true, I nevertheless believe that identification is the most important step. Without identification, the door is closed to assessment and mitigation.

One reason that identification is so important is that a risk driver identified is often a risk driver mitigated. I have seen teams identify and list risk drivers one day, and then within a week cross half of them off the list. All that was needed was that somebody takes a small action, often no more than a phone call.

Too often, risk identification is reduced to a bull session among project team members. If that's all that is done, many significant risks will probably be missed. There are risk search methods much more effective than brainstorming, and they should be used. A few of them are organizational and professional checklists, basis of estimate reviews, and resource loaded schedule networks.

Assessment is what you do with risk drivers you can't mitigate right away, or maybe not at all. Assessment means answering a set of questions for each driver.

- What are the possible outcomes for this driver and how likely, relatively speaking, is each?
- Which tasks can this driver impact?
- For each outcome, what is the cash flow impact on each task?
- For each outcome, what is the duration impact on each task?

I sometimes get complaints that this is too hard. It isn't really if the work is properly organized and the project team is trained to do it. For cash flow and duration drivers, the process can be made quick and simple. For performance drivers, which tend to be the "big swingers" in a project, we need to be more thorough. . Even so, the assessment of a major performance driver seldom takes

more than a few hours of work. These are often multi-million dollar drivers, and are well worth a close look.

Mitigation is what we do with drivers that have intolerable projected impacts at high levels of likelihood. There is something like fifteen or twenty distinct techniques, but for our purposes, we can boil the options down to just four:

- Swamp the risk by adding more resources to the plan, more time to the schedule, or both
- Accept the risk and its possible consequences
- Avoid the risk by reducing project goals or making the plan more robust, assuming a way can be found to do that
- Take specific mitigation actions in the form of insurance, transference, studies, tests, simulations, etc.

The decision as to which of these to do is usually an economic one. Among the questions that should be answered in making it are:

- Is it a project breaker if this risk doesn't get mitigated?
- Is it impossible to mitigate?
- Can I get rid of it somehow?
- What specific mitigation actions are reasonable, and what will they cost?
- How likely is it they will succeed?
- How does the estimated cost of the mitigation actions compare with the size of the risk?

Once a decision has been made to proactively mitigate a risk driver, the work effort should be managed like any other project task. It should have a schedule, a budget, resources, ownership, and metrics, including earned value. A common mistake is to treat mitigation as a part-time off the record activity.

Who should do PRM, and when should it be done? I am devoted to the principle that project teams should, within guidelines set down by the project manager, manage their own risks. They should do all of the steps: identify, assess, and mitigate. There should be one or more risk coordinators to be sure teams don't go at cross purposes or miss inter-team problems. As to when PRM should be done, the answer is as long as there are risks you don't want to accept and can't

get rid of. However, most of the focus should be on the first 25% or so of the project, which is where the window of opportunity is typically most widely opened.

## Summary

I have sought to de-mystify the subject of PRM. I believe it is no more mysterious and no less useful than well accepted project management techniques such as earned value management and critical path scheduling. It can easily be a mainstream activity in any major project.

What you need to do is mostly what you will be doing anyway, with a bit extra. You will need crisply defined goals and a well crafted plan that is as robust as you can make it. Your project teams will need to assess each risk driver that resists mitigation by answering the four questions I have posed. Finally, they will have to find economically feasible ways to get rid of the intolerable risk drivers.

PRM is a creative, challenging activity, worthy of the best and wisest intellects on the project team. Experience has shown that the payback is often 20, 40, even 100 to one. On your next project, consider the benefits carefully. Chances are you'll be glad you did.