Overcoming Project Risk: Lessons from the PERIL Database

Tom Kendrick
Program Manager, Hewlett-Packard Company

Introduction

How many projects fail? **Seventy-five percent** (or more) is a frequently quoted figure. There are reasons to be skeptical of this assertion (Standish Group, 1994). It relates only to large IT projects, it uses a definition of failure that few project leaders would accept, and all the data were gathered from high-level managers. Nonetheless, there is no doubt that projects fail too often. One reason for failure is that the risks and expectations grow much more quickly than our tools and methods for managing them. As project managers, we must focus their attention on identifying and managing the risks to have any hope of success.

Projects fail for three primary reasons:
- The project deliverable, as defined, is infeasible.
- The project deliverable is possible, but the timing and resource objectives are insufficient for delivery.
- The project is poorly planned, chaotic, and badly managed.

Effective risk management deals with all three situations. It provides data to modify (or quickly abandon) projects in the first two categories. And, because risk management relies on planning data, it eliminates the third possibility.

Risk management depends on effective risk identification and prioritization. In this paper, you will find a summary of common sources of project risk to assist you in failure-proofing your projects. The data discussed here is gathered from technical project leaders into the Project Experience Risk Information Library (PERIL) database. This analysis expands on earlier work (Kendrick, 2003) that included about half as much data.

The PERIL Database

Over the past ten years, in the context of a series of workshops on risk management, I have asked hundreds of project leaders to describe typical past project problems, defining both what went wrong and the amount of impact it had on their projects. This data is collected in the **Project Experience Risk Information Library** (PERIL) database, and it serves as the basis for the following analysis of high-tech project risk.

In projects, risks encountered are either "known" risks, those anticipated in planning, or "unknown" risks, encountered with no advance knowledge or preparation. The goal of this analysis is to provide a framework for risk identification that significantly improves the ratio of known risks, and decreases the number of surprises.

The data

A few disclaimers at the start:
- PERIL is not comprehensive. It represents a small fraction of the tens of thousands of projects undertaken by the project managers from whom it was collected.
- PERIL is not unbiased. The information was not collected randomly, and all of it is self-reported.
- PERIL represents only significant risks. The point of the effort was to collect data on major problems.

Despite this, the risk information collected represents a wide range of typical risks, and a number of instructive patterns emerge. The geographic and project type information in the PERIL database are summarized in Table 1. Whatever the type or location, most of these projects share a strong dependence on new technology and most involved software development. Most projects had durations between six months and one year, and typical staffing on these projects was between 10 and 25 people.

	Americas	Asia	Europe/Middle East/Africa	Total
IT/Solution	149	34	17	200
Product Development	115	52	20	187
Total	264	86	37	387

Table 1: Sources of PERIL data

In the PERIL database, all risk impact is reported in terms of time. For most of the projects, this was the primary reported impact. For projects where the impact was primarily unplanned overtime, scope reduction, or some other project change, we conservatively estimated an equivalent duration impact to deliver the original scope on a standard schedule. (Project risks that had no time impact or that resulted in project termination are excluded.) The average impact for all risks was slightly under seven weeks, representing about a 20 percent slip for a typical nine-month project. The averages by project type and by region were consistently very close to the average for all of the data, ranging from just under five weeks to over seven weeks.

Risk types

Categorizing risks is a useful way to identify specific problems. Categories suggested by the project triple constraint—scope, schedule, and resources—were used in organizing the PERIL database. The resource, schedule, and scope risks in PERIL are further subdivided into categories and subcategories based on the sources of the risks.

For most of the risks, the categorization was fairly obvious. For others, the risk spanned a number of factors, and the categorization was a judgment call. In each case, however, the risk was grouped under the project parameter where it had the largest effect, and then by its primary perceived root cause. Scope risks were both most numerous and most damaging, but all risk categories contributed significant project delay. These data are in Table 2, and a Pareto chart summarizing total impact is in Figure 1.

	Count	Cumulative Impact (Weeks)	Average Impact (Weeks)
Scope	143	1115	7.8
Schedule	130	733	5.6
Resource	114	750	6.6
Total	387	2598	6.7

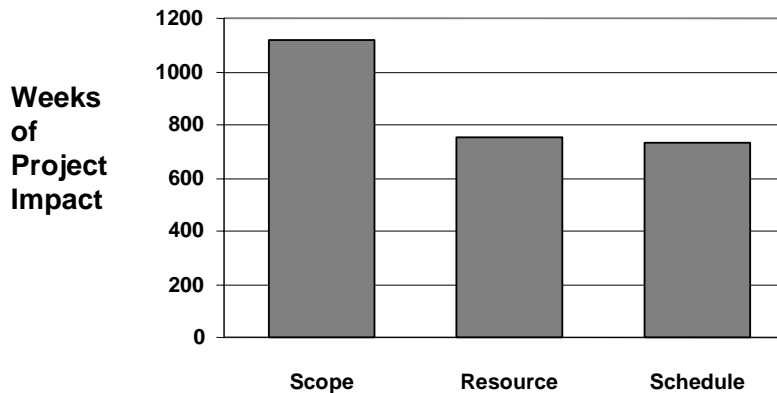**Table 2: Categories of PERIL data**

Figure 1: Pareto of risk categories

Each of these three risk categories is further characterized by root cause. A Pareto of the total impact by root-cause category is in Figure 2. Not surprisingly, scope risk associated with change dominated the data. More detailed descriptions of the subcategory data under these broad categories are included in the next three sections.
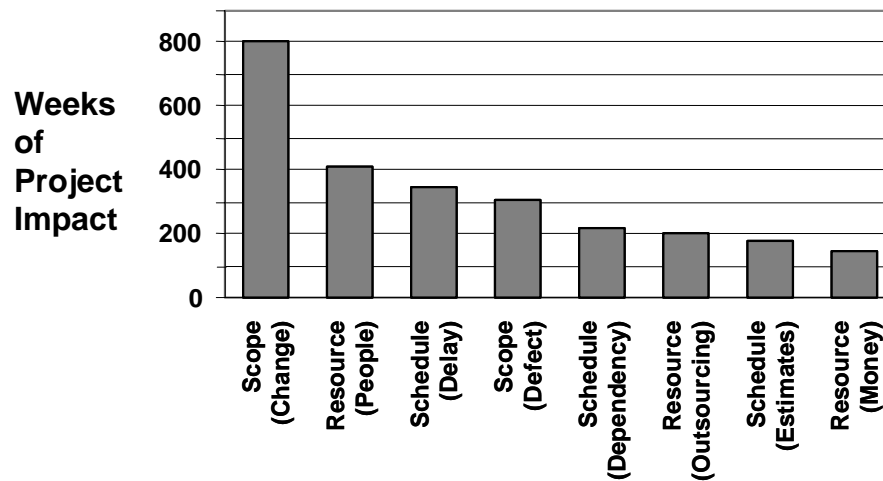

Figure 2: Pareto of risk causes

Scope Risk

Sources of scope risk

Scope risks represented almost half of the impact data in the PERIL database. The two broad categories of scope risk in PERIL relate to *changes* and to *defects*. While some of the risk situations, particularly in the category of defects, were legitimately "unknown" risks, quite a few common problems might have been identified in advance through better definition and planning. Scope risks are summarized in Table 3.

Scope Risks	Count	Cumulative Impact (Weeks)	Average Impact (Weeks)
Change	94	806	8.6
Defect	49	309	6.3

Table 3: Causes of scope risk

These two root causes were further characterized by type:
Change
- Creep: Significant scope change for non-mandatory reasons
- Gap: Legitimate scope requirements discovered late in project
- Dependency: Scope changes necessary because of external dependencies
Defect
- Hardware: Tangible deliverable problems that must be fixed
- Software: System or intangible deliverable problems that must be fixed
- Integration: Program-level defects that require scope shifts in projects

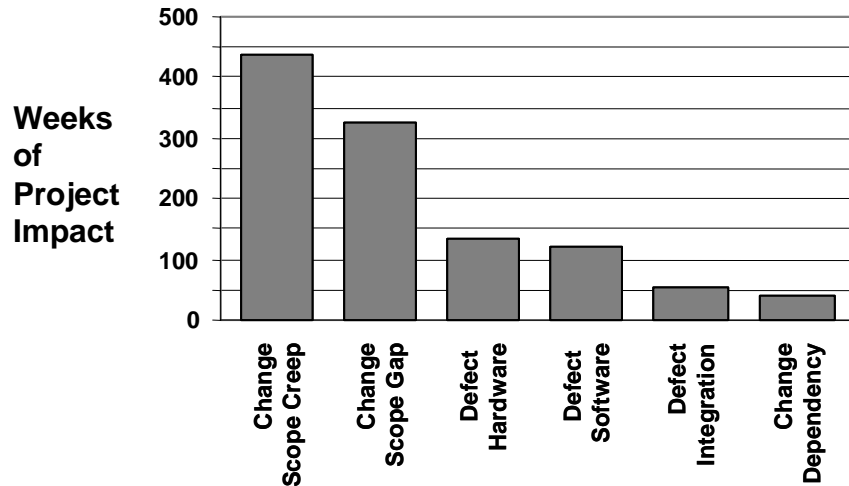A Pareto chart of this data is in Figure 3.

500

400

Weeks of Project Impact

300

200

100

0

Change Scope Creep | Change Scope Gap | Defect Hardware | Defect Software | Defect Integration | Change Dependency

Figure 3: Pareto of scope risk

Change risk

Scope creep is the most serious category, and represented the majority of change risks. Nearly all of these incidents represented unanticipated additional investment of time and money that could have been visible with clearer scope definition. The average slip for projects reporting scope creep was over two months. Some of the changes were modifications of existing specifications, but others were additions well beyond the stated project objective. In severe cases, scope changes delayed the project so much that the ultimate result had little or no value.

Other project changes in the PERIL database resulted from gaps in the project scope that were discovered during the project. Most of these risks are due to overlooked requirements—required work recognized only late in the project. In a small number of cases, the project objective was so unlike earlier work that the gaps were probably unavoidable, but in most of the cases, the gaps came from inadequate analysis.

A third category of change relates to unexpected scope dependencies. (Dependency risks that primarily affect the project timeline, rather than the scope, are included with schedule risks in the database.) While there were some changes that no amount of realistic analysis would have uncovered, most of the situations were due to factors that should not have come as complete surprises. Some were due to changes in the infrastructure the project was depending upon, such as a new version of system or application software or hardware upgrades. Some projects in the database were hurt due to an unplanned delay in access to new software versions or product releases.

Defect risks

Technical projects rely on many complicated things to work as expected. Unfortunately, new things do not always operate as hoped. Even normally reliable things may break down, or fail to perform as desired in a novel application. Hardware failure was the most common defect risk in the PERIL database, followed closely by software problems. In several cases, the root cause was new, untried technology that proved unsuitable. In other cases, a component created by the project (such as a custom integrated circuit, a board, or a software module) did not work initially and had to be redone. In still other cases, critical purchased components delivered to the project failed and required replacement.

Some hardware and software functional failures related to quality. In some projects, functioning components failed to meet a stated standard of performance. Hardware issues included throughput, power consumption, and excessive electromagnetic interference. Software issues included speed and ease of use problems.

The third type of defect risk, after hardware and software problems, occurred above the component level. In large technical programs, work is decomposed into smaller, related sub-projects executing in parallel. Successful

integration of the outputs of each of the sub-projects into the ultimate system requires not only that each component meet its specification, but also that the combination of all these parts works as a functioning system. Integration risk is particularly problematic, as it generally occurs very near the deadline and is rarely simple to diagnose and correct.

Managing scope risks

Discovery of scope risk begins with scope planning and clearer definition of all deliverables. Many late project changes, both discretionary and mandatory, can be eliminated through use of an "Is Not" list, explicitly documenting the boundaries of project scope early so modifications may be either made during initial planning or removed from expectations early. Scope definition—developing a thorough work breakdown structure—will reveal gaps and potential dependencies that represent risk. Worst-case and system analysis are additional tools for probing for scope weaknesses. Exploration of options using better-established technology also minimizes risk. For truly revolutionary projects, evolutionary or cyclic development methods can be effective in managing project scope exposure.

Documented specifications are only part of the management of scope risk, as many of the most significant slips in PERIL were due to poorly managed (or unmanaged) change. Once the scope is set, it needs to be locked into a baseline that is changed only through a disciplined process for scope change control.

Resource Risk

Sources of resource risk

Resource risks represent less than one-third of the records in the PERIL database, but their impact is, on average, nearly seven weeks. There are three overall categories of resource risk: *money*, *outsourcing*, and *people*. Money is the smallest category, but it represents higher average impact than any other root cause. In addition, money was a contributing factor for many of the people and outsourcing risks. Resource risks are summarized in Table 4.

Resource Risks	Count	Cumulative Impact (Weeks)	Average Impact (Weeks)
Money	9	144	16.0
Outsourcing	30	200	6.7
People	75	406	5.4

Table 4: Causes of resource risk

These root causes of resource risk are further characterized by type:
Money
- Limitation   Slip due to funding limits

Outsourcing
- Delay   Deliverable late from vendor
- Late Start   Contracting and approval delays
- Turnover   Learning curve and personnel thrash

People
- Conflict   Slips due to disagreements, discord
- Late Start   Staff available too late; often due to late finish of earlier projects
- Loss   Permanent staff loss due to factors such as resignation, reassignment, or layoffs
- Motivation   Loss of team cohesion and interest; typical of long projects
- Queuing   Slip due to resource bottleneck
- Temp loss   Temporary staff loss due to factors such as illness, hot site, or other work

The Pareto chart in Figure 4 shows the single most damaging project factor was delay of outsourced work.
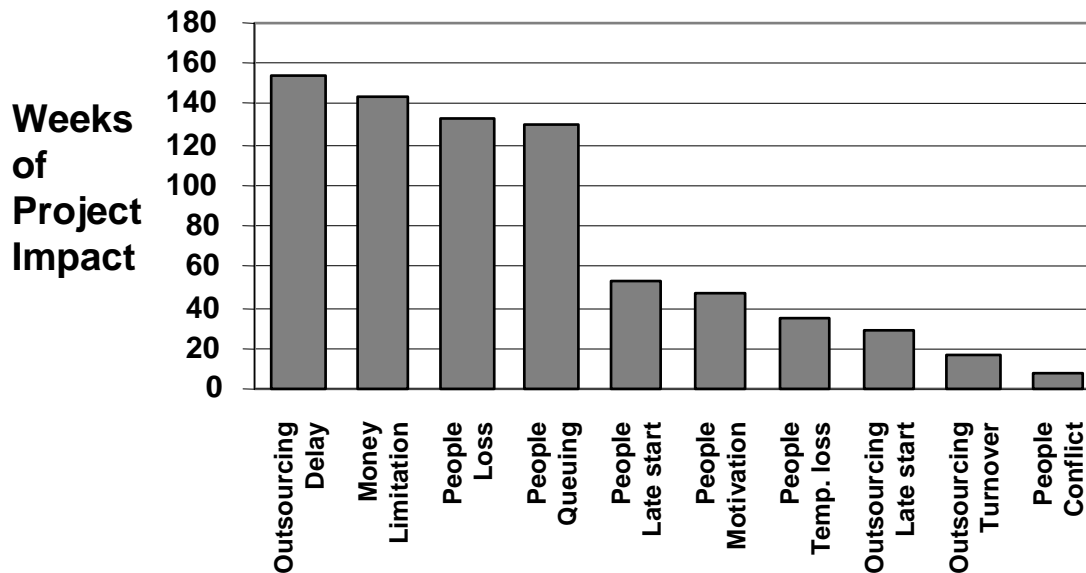
Figure 4: Pareto of resource risk

Money risks

While a lack of money was not very common in the PERIL database, the damage done on projects with a significant shortfall was very high. Several projects were ultimately a year late because of expense caps that resulted in much slower execution than would have been possible (often at higher overall cost).

Outsourcing risks

Outsourcing accounts for more than a quarter of the resource risks. The most significant source of risk is delay, when a supplier fails to complete assigned work on schedule. These risks slipped average projects more than two months on average, and they frequently came as complete surprises very late. Because outsourced work may be done offsite, the project team may not be able to determine the actual the cause of the problem. Project delay was exacerbated in several examples when late deliverables were so deficient the team had to redo them entirely.

Late starts are also fairly common with outsourced work. Contracts need to be negotiated, approved, and signed—all processes that are time consuming, particularly with new suppliers. When the need is particularly unusual, the search for qualified suppliers often results in long delays.

The third subcategory of outsourcing risk, turnover, also contributes to project delay. Particularly with offshore contractors, replacement of staff with new (and often less experienced) people is common, requiring additional training, project planning and specification reviews, and relationships rebuilding.

People risks

Risks related to people represent the most numerous resource risks, nearly two-thirds of the incidents. Staff loss, staff joining the project late, and queuing problems were the most frequent problems, all related to staff availability.

Losing people midproject, whether permanently or temporarily, represented almost half of the reported incidents of people risk, with permanent loss leading to an average slip of more than six weeks, and temporary loss causing a typical slip of more than two weeks. The reasons for permanent staff loss were many, with staffing cutbacks as a big risk for current projects. Although its overall impact was lower, temporary loss of project staff was a very common people-related risk, most frequently due to a customer problem (a "hot site") related to an earlier project.

There were also a number of projects that slipped more than five weeks, on average, because required staff was

unavailable at the beginning. Staff joining the project late had a number of root causes, but the most common was late completion of earlier work. Whenever a prior project is late, some, perhaps even all, of the staff for the new project is remains tied up. As a consequence, following projects get a ragged start, with key people beginning to work only after they break free. Even when these team members do become available, there can be additional delay, as they are often exhausted from the bulge of work and long hours typical of a troubled project. This "rolling sledgehammer" creates a self-perpetuating cycle that is very difficult to break.

Queuing for scarce resources also delayed typical projects by more than five weeks. Specialized expertise is often expensive, so businesses minimize the cost by investing as little as possible. Most technical projects rely on special expertise shared with other projects, such as system architects initially, testing personnel for closure, and other specialists throughout the project. If an expert happens to be free when a project expects work to start, it executes as scheduled. If the expert has activities for five other projects queued up when your project activity needs attention, your work enters a queue, causing all dependent project work to slip. Optimizing project resources based only on cost drives out any spare capacity and causes project delay.

Other people-related risks in the PERIL database involved conflict, where people failed to get along and work cooperatively, and poor motivation. Falling morale is one risk (among many) on lengthy projects.

Managing resource risks

Thorough resource planning and continued strong project sponsorship are tactics that maintain adequate funding. Resource planning is essential for managing outsourcing risks, providing clear and well-documented resource requirements throughout the procurement process. Effective negotiation and thorough understanding of all the terms of the contract also minimizes outsourcing risk. Work to ensure that both the project team and the outsourcing partner understand the terms and conditions of the contract, especially the scope of work.

Managing staffing risks can be difficult, but thorough planning and credible scheduling of the work well in advance will reveal many of the most serious potential exposures. In planning, determine the staffing of all project activities *by name*, and get explicit commitments from all activity owners and contributors. People-related risks related to conflict, misunderstanding, and motivation can be mitigated through co-location of the team, at least periodically.

Histogram analysis of resource requirements can provide insight into staffing shortfalls, but unless analysis of project resources is credibly integrated with comprehensive resource data for other work within the business, the results may not be useful. Aligning staffing capacity with project requirements requires ongoing attention; the primary root cause for understaffed projects is initiation of new projects without considering project planning information, triggering the "too many projects" problem. Good record-keeping and trend analysis are useful in setting realistic project expectations. Retrospective analysis of projects over time is also a powerful way to detect and measure the consequences of inadequate staffing, especially when resource problems are chronic.

Queuing analysis is well understood in fields such as manufacturing, engineering, system design, and computer networking. A project, like any queuing system, requires some reserve resource capacity to maximize throughput.

Schedule Risk

Sources of schedule risk

Schedule risks represented just over a third of the records in the PERIL database. They fall into three categories: *delays*, *dependencies*, and *estimates*. Schedule dependency risks related to unanticipated linkages or missing inputs that affected the project timeline (dependencies that primarily affect the project deliverables are grouped with scope risks). Another category of schedule risk comes from duration estimates that are insufficient for completion of scheduled project activities. Delays occurred whenever something expected by the project—a part, a decision, a piece of information—was late. The summary for schedule risks is in Table 5.

Schedule Risks	Count	Cumulative Impact (Weeks)	Average Impact (Weeks)
Dependency	29	216	7.4
Estimates	27	175	6.5
Delay	74	342	4.6

Table 5: Causes of schedule risk

These root causes of schedule risk are further divided into subcategories:

Dependency
- Legal — Legal, regulatory, or standards shift
- Owner — Multiple owners or loss of owner leading to delay
- Project — Interface delay in programs
- Space — Infrastructure lapse in project
- Support — Needed support not available (printing, IT, shipping, etc.)

Estimates
- Deadline — Top-down imposed unrealistic deadlines
- Judgment — Poor estimating process of lack of analysis
- Learning — New work assumed to be easier than it turns out to be

Delay
- Decision — Slip due to untimely decision for escalation, approval, phase exit
- Hardware — Needed equipment arrives late (or breaks)
- Information — Slip due to unavailability of specification or other needed data
- Parts — Delay waiting for needed components of deliverable
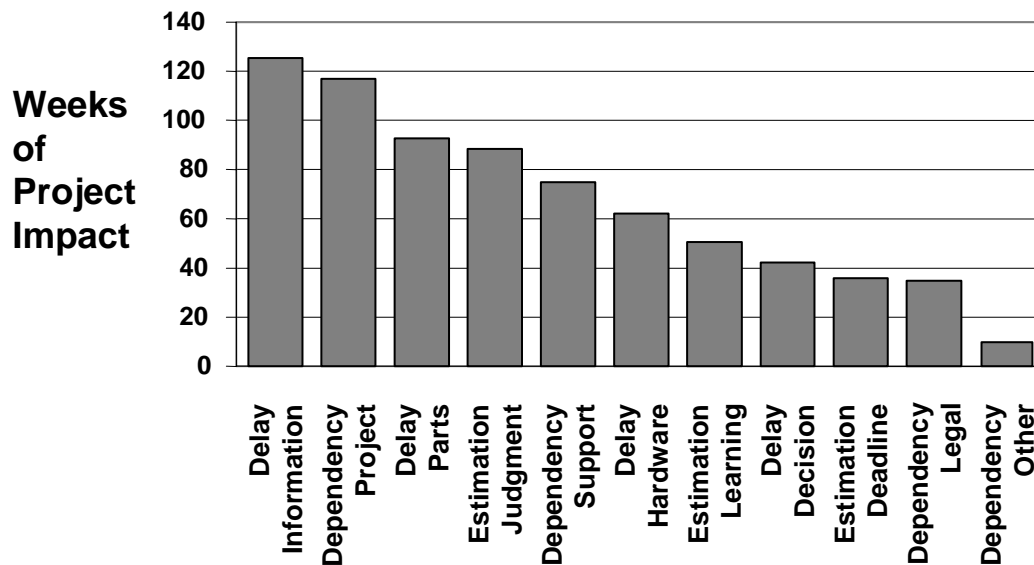
The Pareto chart of these risk causes is in Figure 5.

Figure 5: Pareto of schedule risk

Dependency risks

Dependencies, on average, are the most severe category of schedule risk in the PERIL database, averaging over seven weeks of impact per incident. There are a number of dependency types, but the most numerous ones in the

database are dependencies on other projects and on project support, each averaging about two months of project slip. The highest average, not surprisingly, was nearly three months of slip for legal problems.

Dependency risks from other projects are common in programs, whenever a number of smaller projects have cross-dependencies. In addition to providing each other with information and deliverables that meet well-defined specifications (which is a scope-risk exposure), each project within a larger program must also synchronize their timing to avoid being slowed down by (or slowing) other projects. Managing all these connections is difficult, and the consequences increase with time; many of the risks in the PERIL database were noticed only late in the project. Even for the interfaces that were defined in advance, delay was fairly common due to the significant likelihood that at least one of several related projects would stall. With many possible failure modes, problems are almost certain.

Other significant dependencies that interfere with project schedules involve support problems. Examples include downtime for computer systems or networks required by the project, or inadequate access to resources such as help desks, system support, and people who understood older applications. Several projects were delayed by scheduled maintenance outages that were unknown to the project team. One project had severe impact when the legal and paperwork requirements for international shipments changed abruptly.

Estimating risks

Of all the types of schedule risk found in technical projects, estimating seems to be the most visible. People who work on technical projects are usually well aware of how inadequate their estimates are, and freely admit it. Despite this, the frequency of risks due to estimates in the PERIL database is fairly small, and their impact is about equal to the average for the database as a whole. Half of the estimating problems reported in the PERIL database relate to judgment. For a good number of projects, consistently overoptimistic estimates were the issue, with some too short by factors of three or more.

A small number of cases of estimating risk involve learning-curve issues. The impact of this was above the average for the database—over seven weeks. The quality of estimates involving new technology or new people (or both) is poor. There were also cases in the PERIL database where the estimates were wrong but the root cause was outside the project. Technical projects frequently have aggressive deadlines determined in advance with little or no project team input. An the unrealistic deadline is retained, even when the project plan shows it to be impossible.

Delay risks

Delay risk represents well over half of the schedule risks, and nearly a fifth of all the risks in the PERIL database. Impact from delays was lower on average than for other risks, slightly less than five weeks.

Delay in getting information resulted in higher than average slippage—nearly seven weeks. Some of the delay was due to time differences on global projects. Losing one or more days on a regular basis may be common, due to communication time lags.

Most of the reported delays related to tangible items: parts that were required for the project deliverable or equipment used for project work. Delay due to parts and equipment both averaged about one month. Delivery and availability problems were a common root cause for the delay, but there were also quite a few issues around international shipping, including customs and paperwork. On-time delivery of defective items also caused delay.

Slow decisions also cause slippage. Inaction by managers or other stakeholders typically resulted in three week delays. Sometimes the cause was poor access to the decision makers, or their lack of interest in the project. For other projects, delays were the result of extended debates, discussions, or dithering.

Managing schedule risks

Dependency analysis is an essential part of project planning, and it is particularly important in large programs. Better monitoring for planned or likely changes in the project environment and organizational infrastructure can forewarn of many potential dependency problems.

Better estimating can be achieved by more thorough planning, and it also requires good record-keeping. Metrics and project data archives are invaluable in developing future estimates that are more consistent with reality than past estimates have been, even for projects where things change rapidly. Having *some* data always beats having to guess.

Another powerful tool in revealing and combating optimistic estimates is worst-case analysis. Not only will the answer to the question "What might go wrong?" reveal something about the likely impact, it will also uncover potential sources of risk. The portions of project work that require staff to do things they have never done before are always risky, and thorough analysis of the work can show which parts of the project plan are most exposed. Training plans must be established for the project whenever new skills and capabilities are necessary, and need to be explicit in the project timeline and budget.

Potential delay risks may be difficult to anticipate, and many of them may legitimately be "unknown" risks. Thorough analysis of the input requirements at each stage of the project plan, however, will highlight many of them.

Conclusion

Managing risk better begins with robust identification of potential problems. Use all the sources you can find: post-project retrospectives, brainstorming, analysis tools, and other people's data.

To prioritize and assess risk severity, use qualitative (and when appropriate, quantitative) risk analysis methods. Identify the most severe risks, and work to avoid or mitigate all risks you can prevent. Establish contingency plans for, or transfer, all remaining severe risks. Risk is a consequence of project size and length; consider ways to break big projects up into less complex, smaller projects.

Avoid the three types of "impossible" projects. If planning and risk analysis reveal that your project deliverable cannot be created, use your data to **change or stop the project**, early.

If your planning data shows the project to be overconstrained, use your risk data. **Negotiate** for more time, a higher budget, or both (you are, after all, the greatest living expert on your project). Establish schedule and/or budget **reserves**, and scrupulously **manage risks**.

Finally, if you lack a plan, **CREATE ONE**.

References

Kendrick, Tom (2003). *Identifying and Managing Project Risk: Essential Tools for Failure-Proofing your Project*. AMACOM: New York.

Standish Group (1994). *The CHAOS Report*. The Standish Group International: West Yarmouth, Mass. (http://www.standishgroup.com/sample_research/chaos_1994_1.php)