eCameron, Inc.

Project Management Newsletter

Issue 11
September 2006

In This Issue:

This Issue

Classical, Critical Chain, Agile Project Management and How They Relate A comparative look at ten aspects of projects, where they fail and how these methodologies approach the issues.

Identifying and Managing Project Risk, by Tom Kendrick. A general text on project risk management which covers many aspects of accounting for risk in project planning and execution.

Next Issue

Change Management. Change Management is more that a process, forms and policy, there are ways of implementing and specifics to certain methodologies that change how you should look at scope change.

Critical Chain Project Management, by Lawrence Leach. A thorough text on Critical Chain describing the details the theory, how to implement it and monitor a project's progress.

Newsletters are published monthly and issued mid-month. To receive back issues, be added or be removed from the distribution, send an email to:

newsletter@ecaminc.com

Methodologies

In the spirit of the upcoming Pacific NW Software Quality Conference's session on "*It's Not What You Do, But How You Do It*", we felt it was appropriate to do a comparative analysis of project methodologies. In this issue Agile, Critical Chain and classic methodologies will be juxtaposed to explain how they work, advantages of each and where they may not be appropriate.

The final article in our series on risk is a synopsis of Tom Kendrick's book ***Identifying and Managing Project Risk,*** an entry level text on risk management.

CCPM, Agile and Classical Project Management

In the project management community, there is significant debate about project management methodologies. The conversation centers on three major classifications—classical (aka waterfall), Agile and Critical Chain. These debates often have a religious fervor where the proponents of each methodology cite attributes of other methodologies that they feel contribute to project failure. This article will provide a high level overview describing some of the attributes of these methodologies and how they relate to each other in order to provide a neutral view.

Classical project management techniques have been in use for over a century. For that reason there is a plethora of data that has been analyzed by numerous people and organizations. Examples of successful projects using this method are obvious in construction (i.e. canals, buildings), manufacturing and new product development (aircraft, automotive) and science and military (NASA, weapons development). There have been equally significant failures.

While it is easy to find discussions and proponents of the different methodologies it is a little more difficult to find an analysis of the methodologies and how they might apply to a situation that a practitioner may encounter.

Each of the methodologies works, but only if implemented properly. Misconceptions, improper implementations and "corporate abuse" will cause any of these methodologies to fail. But, discussion of these issues will have to wait for another article.

In this article I would like to point out some of the differences in the processes. To achieve this, each methodology is compared in ten general categories (presented alphabetically). All attempts have been made to present a neutral view.

Project Methodology Workshops

Your company now has the choice of three workshops to assist in improving your project performance.

1. **Recovering Projects:** Every company has a project that is over budget or behind schedule, sometimes more than one. This workshop focuses on tips and techniques you can use to get those projects back on track.
2. **The Right Project Methodology:** Agile, Critical Chain, classic waterfall, one or more of these methodologies may be applicable to your business. This workshop presents a systematic process for implementing the right methodology for your company.
3. **Setting up a Project Management Office:** Young or growing companies often find themselves doing an increasing amount of project work and need to consider which process to follow to minimize their efforts and maximize their return. This workshop will help you determine the level of process required in your organization.

For more information email:

workshops@ecaminc.com

or visit our website:

http://www.ecaminc.com/Workshops/Workshops.html

Change Management

Scope Creep is commonly cited as the reason projects fail. The scope wander and projects end up over budget and/or late. On the other end of the spectrum the user's requirements may change and are not reflected in the product resulting in a deliverable with significantly less or no value. This is not scope creep; it is improper scope management. Scope changes on projects and change management processes need to be in place and rigorously followed. Customers need to understand and adhere to the process.

Classical: Simply put, too many Project Managers do not follow a change management process. They either allow changes indiscriminately or prohibit all changes. This will certainly drive a project toward failure. The result will be a late delivery or a product with no value.

Agile: Agile is often criticized for allowing anything to change any time. This claim is far from the truth. Agile has a predefined number of hours to complete a set of features (not tasks) that the delivery team commits to complete. If something has to change, it can, but it cannot alter the hours or financial obligation of the given iteration. If the customer wants to add something that takes thirty hours, then they must remove features that take the same amount of time, maintaining the iterations integrity. This is not done by the Project Manager; it is done by the delivery team and the customer representatives. They know what work is required and they are responsible for delivering the iteration. If they don't meet the delivery, both the team and the customer representative are held accountable.

CCPM: Critical chain methodology preaches religious change management process. There are no special features in the Critical Chain methodology to promote or enforce this. Change requests are addressed and the resulting impact reflected in the plan.

Customer Relationship

In general, customers have devolved into the project's nemesis instead of what they should be—a partner. The reasons are numerous but here are some reasons I have experienced:

- Not talking to the customer openly, since it is felt that the customer may determine out that there are parts of the project that are have not figured out.
- Not letting the customer interact with the team because that will let them sneak in scope changes and the project will fail.
- Assuming the project team knows more about the needs than the customer does, so they do not need to be consulted.
- Taking the attitude that the customer always demands the team do things that are not in scope or are really their responsibility.

Removing communication with the customer most often (if not always) results in disaster.

Classical: There is nothing in the classical method of project management that causes customer relationship problems; inappropriate procedures put in place to solve poor management processes have created the situations mentioned above. The increase in technology has exacerbated this issue. Technology savvy engineers end up too enamored by the technology and forget that the goal is a product that is easy to use and solves a problem.

Agile: Agile goes to the greatest lengths to solve this problem by trying to co-locate the customer and the development team. Where practical, this is an excellent way to solve the problem but with dispersed teams or large projects this can be impractical. During an iteration, the customer validates the features being developed and may make changes in a feature to provide more value. This reduces the confusion in translation of requirements and, by achieving mutual agreement between the customer and the team, minimizes the impact of the changes.

CCPM: Critical Chain approaches the problem from a very different angle. Co-locating the team may be a solution used, but instead they first focus on the root cause of the problem. For instance, they would look at the team's communication using an "evaporating cloud" as shown in Figure 1. It would note that there is a conflict between having the team talking directly with the customer and not talking directly with the customer. This conflict points to an incorrect business assumption. The conflict may result in an architect being "in the middle" to translate the requirements into a document and having the team interpret the document. This leaves room for errors in multiple interpretations and may results in building a feature incorrectly. The Critical Chain team would look for the incorrect assumption and work to resolve it. In this case the incorrect assumption is that the team will allow the scope to change when talking to the customer. The solution may be educating the team on the scope and methods to remove scope creep.

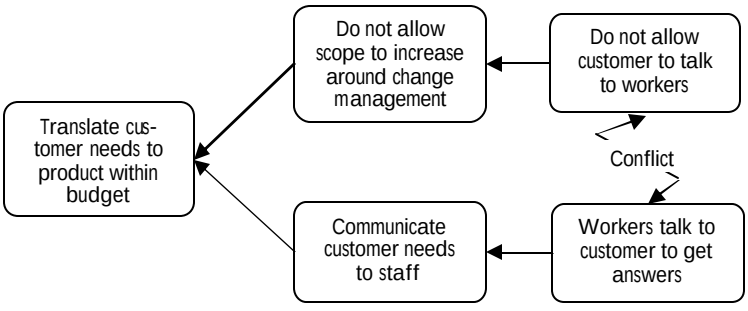


Figure 1 - Evaporating Cloud for Conflict Identification

Estimations

In general people hate estimating. Their estimates are always wrong. But the *estimate* is not wrong; it is treating the estimate like a quote that is wrong. Management will inevitably say "I thought you told me it would be done in X days. Where is it?" To reduce the pain, an entire culture has been set up. First, people determine an estimate they are almost certain to hit (say, a 90% chance), then add ten to twenty percent, since they know management will remove that amount. They do not use methods like PERT because if management finds out about the optimistic numbers they will hold people to those numbers. If people finish early, they fear it will prompt management to trim estimates even more based on the "current empirical data".

Essentially, estimates are expected to be quotes and people are reprimanded for not making their estimate. Behavior is molded to avert pain and results in an undesired effect. Both Agile and Critical Chain are empathic about addressing this issue.

Classical: Classical methods of managing projects have degenerated into the above game. This is not the fault of project methodology, it is the result of managers and stakeholders not understanding estimates and treating the project as if it were a deterministic system. The Project Manager must address this with the stakeholders.

Agile: Iterations are relatively short (thirty days). At the beginning of the iteration the development team "commits" to the functionality to be provided in the iteration. Since planning horizons are short, the scope tends to be better identified. The product of the iteration must contain customer ready functionality (potentially shippable). The team is required to stay focused on the deliverables and work only on the tasks that they have chosen. At the end of the cycle they must demonstrate and deliver that functionality.

The key is accountability. The Team commits to the functionality they are going to build, laying out the steps to achieve the proper functionality, accepting tasks to complete the functionality, documenting it and testing it. It is not the Project Manager's responsibility to lay this out.

The Project Manager's job is much harder—reeducating management on estimates. The team may not deliver what they have "committed" to since these were estimates. There are completed items and time-to-complete for the remaining functionality. The Team does not get reprimanded for "lack of delivery" unless they did not follow the rules of Agile, then they get reprimanded for not following the rules.

CCPM: Critical Chain follows the same philosophy of not reprimanding team members for missing their estimates. They get reprimanded for not following the rules. Some of rules in Critical Chain, though, are different since planning horizons are longer.

Estimates for task durations are supposed to be "50:50 estimates". This means that there is a 50% chance that task will be completed in the estimate length of time. If all tasks are estimated the same, some will take longer and some take shorter time, on the average they will remain on schedule. But, as with the Theory of Constraints, the constraint (the resource leveled Critical Path, aka critical chain) cannot be allowed to be left waiting for work. Therefore, prior to any branch's merge with the critical chain a buffer of time (a feeding buffer) must be inserted. This prevents statistically late branches from affecting the Critical Chain.

The primary rule is that team members must be dedicated to their work. They must not multi-task or pick up other work that is not part of their assigned task. The Project Manager's job is to be the enabler for this behavior. The Project Manager must assist in prohibiting other people from interrupting the tasks being performed.

Process

Processes for project management are generally left to the Project Manager or a company's Project Management Office (PMO). There are few defined processes that are standard. Part of this is due to the fact that there are so many different types of projects. Organizations like PMI have developed guidelines, but these are more focused on "what" to do, not "how" to do it. This is due to the earlier point—all projects are different. Agile and Critical Chain define specific processes that are directed at parts of projects that most often fail.

Classical: Unless a Project Manager or an organization has a set of defined processes, the most one finds are listings of "What" should be done. The value of the seasoned Project Manager or a consultant is their process. It is the "how" not the "what" that completes the project.

Agile: Agile's claim is that it is a lean process. That does not mean that it does not have processes. The processes it has are extremely rigid. As mentioned above, I will argue that it has one of the most rigid Change Management processes for project work. The Configuration Management process is not only rigid it is extremely efficient and therefore lean. In essence, it will allow scope change to include item "A", but only after the customer has removed some number of items that are of equal size (effort and cost). Other processes and rules are also rigidly enforced, these include:

- Time-boxed iterations and meetings
- No feature additions within an iteration
- Strong customer interaction
- No multitasking on items outside the iteration's scope

This results in the ability to easily identify areas where the scope has changed.

CCPM: Critical Change follows the general outline provided by PMBOK, but applies defined processes to critical areas. Where it differs from PMBOK are in its approach to handling the human/managerial element of projects and Critical Chain's categorization of risk into "common-cause" and "special-cause". This is an extremely important factor in developing the methodology and in taking corrective action to deviations in the project. Common-cause variation is normal variation in the process that must be accepted and cannot be controlled. The area where it is most evident is in estimates. All estimates are established at a 50% probability of being achieved. This means that half of the tasks will take more and half will take less time—there is nothing that can be done to change that. It is common-cause variation in the system.

Prominent processes that Critical Chain enforces are:

- Estimates are based on a 50:50 chance of achieving.
- All project schedules are resource leveled.
- Project buffers are established at 50% of the estimated project length.
- Feeding buffers (buffer at the end of all task chains feeding the Critical Chain) are established at 50% of the feeding chain length.

This results in the advantage of highlighting areas where additional resources will compress the schedule and a 25 to 30% shorter project.

Project Constraints

Projects are governed by three constraints—scope, schedule and resources (money and people). It is well known that a single constraint cannot change without affecting at least one of the others. Scope cannot increase without increasing schedule or resources; schedule cannot be shortened without reducing scope or increasing resources, and so forth. Of course, what customer does not want more for the same price and schedule? But it simply does not work.

Classical: Nothing in a classical approach to project management inhibits enforcing project constraints. But it is often not properly spelled out to the customer. To make matters worse a lax change management process will promote unaccounted scope changes. Being nice to the customer and giving them something-for-nothing does nothing to help them when the project is late and they do not get the functionality they need.

Agile: Agile has very specific rules governing constraints. It is a required step in the project initiation—fix one constraint, allow moderate change in the second constraint and let the third constraint float. This prioritizes the constraints. After this is set the time-boxed iteration assists in enforcing the rules. If there is a change requested during a cycle it cannot effect the end date of the cycle. Changes in the project (between iterations) must use this for prioritization. Requested features are added to a list and prioritized by the customer with a full understanding that other features will be forfeited. There is generally a fixed amount of time and money to implement the features, when it is used up the project is over.

CCPM: Critical Chain has very specific rules governing constraints. It is a declared step in project initiation—just like Agile. The assumption with Critical Chain is that anything worth doing it worth finishing as quickly as possible. Hence schedule generally takes priority. Any changes to the project must follow a strict change management process. In this light, this methodology is similar to classical project management methodology.

Project Manager

There are probably as many project management styles are there are projects. But the role of a Project Manager has become a label that anyone can tack on his or her name. I disagree with this practice. Project management requires facilitating the formation of a team, getting them motivated to complete their work, working with stakeholders to show progress and looking for required changes in the deliverable. The Project Manager must feed and water the team and make sure their managers meet their needs. The Project Manager must be a leader.

Classical: There are no rules that I'm aware of that say a project manager must do given tasks on a project. The PMBOK (often cited as the antithesis of Agile) says that the project management task should make sure various items get done, but it does not say that the Project Manager must do them. That is an implementation decision of the organization or the Project Manager. If improperly implemented, that task will fail.

Agile: The Project Manager does not control scope, give daily assignments or track daily progress, the team does. The Project Manager specifically removes roadblocks, reports on project (not task or feature) status and keeps the delivery team and their customer representatives from being interrupted or reassigned to other tasks. They enforce the rules of Agile methodology. The team reports features complete, changes between iterations and hours to complete features and the project.

CCPM: Much like Agile, Critical Chain looks to the Project Manager to be a leader, not a manager; fostering and protecting the team, keeping them on task. The key functions of the Project Manager are to coordinate with other projects, remove multitasking of people and enforce the rules of Critical Chain making sure schedule and resource buffers are properly minded to.

Suppliers Relations

Suppliers of materials and resources (outsourcing) introduce a significant amount of risk into a project. Unfortunately in too many cases (regardless of methodology) the corporation has placed a purchasing department between the project team and the supplier. This is to get the best price based on a set of specifications that are "invariant". This results in the same problems that the project team has with their customer to exist between them and their suppliers.

Classical: The key with a classical project is supplier management. Unfortunately there are few methods I know of other than implementing lean principles for improving the supplier relationship—making the supplier a partner and having open and honest communication.

Agile: The Agile processes levy the same involvement of the supplier as they do the customer. If they are integral to the process of creating the product, they are co-located with the team. The intent is that the supplier is using the same methodology as the project team.

CCPM: Too often suppliers are penalized for making late deliveries and never provided an incentive to make on time deliveries. Critical Chain relies on good communication with the supplier to have their deliverables in place at the time they are needed. This is simply a JIT implementation. One method to help achieve this is to provide an incentive to the supplier for on-time delivery (not early/not late). The intent is that the supplier is using the same methodology the as the project team.

Team Focus

Disruption of the team members is well known as reducing the overall effectiveness of the members. Meetings, interruptions from cohorts and managers asking for special work to be completed draw team members off task. A general complaint heard by staff is to give them some peace and

quite so they can get their work done. The reaction, if any, by management is to provide private cubicles or move them to another area of the building. This often results in moving the staff far from the customer where they can complete their work in quiet solitude. The obvious problem with this approach is that teams need interaction and communication within the group and with their customer.

Classical: The aforementioned scenario is often seen in classic project management; give the team a quiet place to work where the customer cannot induce scope creep. Design questions are funneled through an architect and answers come slowly and in the form of thick documentation. This need not be the case. Educating the team on the proper communication with the customer, prohibiting management from meddling with the team and promoting team member interaction (co-locating the team close to the customer) builds stronger teams and reduces variation in the project.

Agile: One of the primary goals of Agile is to build a team close to the customer. The Project Manager has the responsibility of stopping outside interference. Outside interference is defined as outside the project as a whole, not outside the delivery team. Direct communication of the delivery team with the customer is a critical component.

CCPM: One of the tenants of Critical Chain is to stop the multitasking of the team. Assign them to a task and let them finish the task and move to the next task. The idea that a multitasking individual is most productive is incorrect. Getting restarted from an interrupted task is a huge waste of time. Removing multitasking requires thorough planning but results in a shorter start-to-finish time for all projects—the cost is that some projects start later, but they finish in less time, maintaining their due date.

Team Members

The first line in team selection for Agile projects is to get above average team members that are self-managed and organized. Many end up thinking "Well of course, if you require better than average team members you will get better than average results." But why isn't the thought "That makes sense, let's do that"? The reason lies in the fact that most Project Managers do not pick their team, they are given a team or a pool of resources to pick from that may not have the right skill set. Management says that these are your resources and they are better than nothing. Cost accounting—instead of spending $80K on a resource that can do the task we will spend $60K on a resource that will make the project four months late and cost the company $200K. But, the individual will show productivity since he or she was working on a project.

Classical: Nothing in any book, standard or practice that I have seen has ever read said to work with sub-standard people for a task. But it happens. I have been caught in this dilemma—trying to keep good resources assigning them tasks for which they are not qualified. These are hard workers with bills to pay; why should that person take the brunt of the fact this project cannot use their skills at this point in time? Being empathetic, one wants to "find" something for the better resources to do on the project, even if they are not completely qualified for the task. Resources should not be penalized for bench time due to poor policies.

This is not the Project Manager's problem. This is a cost accounting problem that must be addressed and if the person has good skills they get put on something productive, or billed to all projects, or maintained in the cost of doing business.

Agile: Every Agile process wants "the best" or "above average" people to work tasks. Beyond knowledge, the key attributes for team members are:

- Self-organized
- Self-managed
- Open minded
- Team oriented

Team members are trained in the processes of Agile and need to adhere to those rules.

CCPM: The requirements for the Critical Chain follow the same as Agile but "retraining" on how to do work is more significant since it entails changing mindsets. People are learning an entirely new culture. For instance:

- Time estimates are not those that the will guarantee you will finish the task, but give you a 50:50 chance of completion.
- Multitasking in order to please other people and be a "team player" is not allowed.
- Completing tasks early is more important than "saving pad" for the next project.
- Finishing over the estimated time is not bad since there was only a 50:50 chance of achieving it.

Variation

Uncertainty is the inability to predict what is going to happen. Variation is the change that is experienced in a system; you know there will be variation; you are uncertain what the value will be in a given instance. Every system has variation and it is fruitless to try to control it. In fact, attempting to control natural variation usually makes it worse. Influences from outside the system can be controlled, though, to make the system perform better.

Classical: Governing models, like that of PMBOK, make no distinction between internal and external cause of variation on systems nor do they discuss system theory. Hence classical project management is often ill equipped to handle variation. For this reason, when variations are experienced in a project it usually results in an inappropriate reaction by management. Variations from external influences can go for long periods without being noticed. Small shifts in task durations or cost can be less noticeable. At the same time, normal variations (fluctuation around estimates) may solicit overreaction.

Agile: Variation in Agile projects is handled by continually using short time frames for project check points. A thirty- to forty-day cycle between each planning cycle quickly highlights excessive variation and allows plans to be made to adjust the project. Excess variation is very visible and hence needs less explanation than in classic project management methodology.

CCPM: Critical Chain methodology is built around system

theory and the concept of internally and externally influenced variation. This is the center point of the estimation process. Normal variation for a system will cancel itself out since the deviation around the mean should be equal in the positive and negative directions. External (special-cause) variation will result in drift in one direction. The methodology relies on understanding the "system" of project management and thorough texts, like Leach's, cover this in detail.

In conclusion, it is the defined processes that make Critical Chain and Agile appear so superior to many classically managed projects. These processes can be applied to many projects to improve them, but the real goal of these methodologies is to tear down the perversions to other methodologies such as adding processes which change estimation process, cost accounting, treatment of variation and customer relations that have caused the problems in project management.

Where they shine:

Classical: Classical project management is more easily used in low-risk projects where the general type of project has been done many times in the past. History provides a reduction in risk and less variation in the task durations, in other words, there is predictability.

Agile: Agile works best on smaller projects in the realm of new product development (hardware or software). These projects are not predictive due to the fact that they are covering new areas, and often use new untested technologies. In new product development, requirements may not be well known and iterative cycles accommodate change "naturally". Other areas can benefit from the Agile's approach, but Agile loses a number of advantages when projects become larger and require coordination among numerous groups. A common complaint of the process (a project management failure) is lack of traceability and maintenance documentation.

CCPM: Nearly any project can benefit from Critical Chain methodologies. Although some projects may be better served with Agile, Critical Chain addresses flexibility in a manner that will help new product development.

Book Synopsis

Identifying and Managing Project Risk, by Tom Kendrick, is about identifying and managing risk on projects. Although the book is written in a very generic manner, it has a decidedly high-tech flavor. This is partially due to the fact that the author worked at Hewlett-Packard for twelve years. Kendrick aligns the chapters of the book to the Project Management Institute's (PMI) ***Guide to the Project Management Body of Knowledge***, (PMBOK) 2000 edition and the text is designed to be used as a supplemental source of information when studying for the Project Management Professional (PMP) certification exam.

Kendrick's coverage of risk, and more prominently uncertainty, is complete in a general fashion focusing a majority of his discussion on risk in projects due to poor planning and inadequate change management processes.

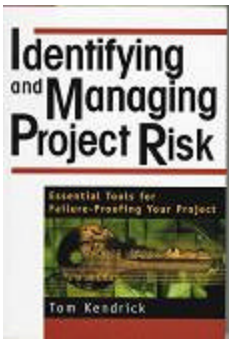
He uses a collection of project elements from various projects of his seminar attendees. He uses this data, Project Experience Risk Information Library (PERIL) database, to quantify and rank classes of risk. In the early part of his book he uses this significantly and the Appendix lists approximately 120 of the element's descriptions.

The book is structured to follow the PMBOK stages of a project—initiation, planning, controlling, executing and closure. Each chapter discusses a set of concepts and concludes with a bulleted "Key Ideas" section and an anecdote from the two attempts to construct the Panama Canal.

Analysis

The introductory chapters lay the groundwork for people who are new to project or risk management. Kendrick starts with the definition of risk as the "loss multiplied by the likelihood" and explains that this relates to uncertainty in estimates for duration and cost. He identifies the benefits o frisk analysis as:

- Project justification



Identifying and Managing Project Risk

By Tom Kendrick

March , 2003

354 Pages

ISBN: 0814407617

- Lowering cost and confusion
- Prioritization and stakeholder support
- Input for portfolio management
- Mitigation
- Setting expectations and establishing reserves
- Communication and control

Project Risk Planning

He continues the introduction by justifying project planning and details the challenges one might encounter in an organization that feels a project planning methodology is not needed. He describes ways to address the need to set up a planning process and says the implementation should be scaled to the size of the projects being performed.

The PERIL database is described and qualified while some of the biases in it are enumerated. Within the three primary constraints on a project, the database shows the risk elements frequency of occurrence as 1) schedule, 2) scope and 3) resource. Implicitly the reader can determine the database classifies each risk with a description, Project type (IT, Product development, etc.), schedule impact, cost impact, class (scope, resource, schedule) and subcategory.

Key Ideas: Project Risk Planning
Risk should be a primary driver for project selection
Project planning and definition are the foundation to controlling risk
Risks should be maintained in a Project Risk Plan

Scope Risk

Using the PERIL database, Kendrick says that even though the number of risks classified as scope related are one-third of the entries, they account for approximately half of the cumulative schedule delays. He enumerates the ranked sources as:

1. Scope creep
2. Hardware defect
3. Software defect
4. Scope gap (ill defined scope)
5. Dependency change (unexpected legal, regulatory, etc.)
6. Integration defect (change due to unexpected behavior)

Kendrick describes a variety of methods for arriving at and defining deliverables and hence defining the scope. He suggests using the "is/is not" method of bounding the scope.

Three high-level risk assessment tools are discussed—Risk Framework, Risk Complexity Index and Risk Assessment grid. Risk Framework looks at the project's technology, the market and the manufacturing effects and uses the relative change to each of these to determine the risk level of the project. Risk Complexity Index looks at the technical aspects of the project (technology, architecture and system) and generates a number from 0-99 to quantify risk. Risk Assessment creates a grid of technology, structure and size to estimate the risk.

The risk issues addressed by a work breakdown structure (WBS) are then discussed. Often considered only a project planning task Kendrick points out the uncertainty and risk introduced into a project when the WBS is not fully defined and understood. A WBS at too high a level can leave scope ill defined not allowing for proper estimates or definition of work to be performed. Often WBS elements that are defined at too high a level indicate work that is not understood and implies significant risk due to uncertainty on the project.

Key Ideas: Scope Risk
Clearly define deliverables
Ensure the value of the deliverables exceeds the scope of work
Define a work breakdown structure small enough to ensure work is understood
Assign ownership and determine reasons any items are not accepted
Note all risks, including non-quantifiable risks due to size or complexity of project.

Schedule Risk

Schedule is the second level of risks effecting project duration in the PERIL database. The top five categories (the book lists ten) are:

1. Project Dependencies
2. Parts Delays
3. Estimation errors
4. Decision Delay
5. Hardware Delay

Dependency on external parties is the largest subcategory of schedule risk in the database, (editors note: as might be expected since it is always safe to blame the other party), followed by poor estimations.

To assist in reducing these risks, Kendrick explains that the process should start with the WBS and apply estimates for effort and resources. This is an iterative process. A number of things should be kept in mind:

- Historical data should be used where applicable.
- Resource planning (done next) will affect these estimates so these processes will need to be iterative.
- Be cognizant of people hesitating to give estimates, it may imply additional uncertainty.
- If the durations are greater than two weeks, they should be broken down further.
- Make sure to incorporate holidays, vacations and other non-project time.

He summarizes a number of estimating techniques:

- Project-level Think-Do-Check based on project size.
- Historical data
- Prior experience
- Delphi Group Estimating (a form of consensus estimates)
- Program Evaluation and Review Technique (PERT)

He spends significant time through-out the book discussing the PERT method and clearing up misconceptions on the PERT process. He explains that the PERT method of estimation generates the expected duration of a task by adding the pessimistic, optimistic and four times the most likely durations and dividing them by six. The standard deviation determined is the difference of the pessimistic and optimistic durations and dividing it by six. The standard deviation is a reflection of the uncertainty in the estimates.

After determining the durations and sequencing, the critical path can be determined. The critical path is the longest contiguous path of tasks with no lag. The easiest way to do this is by using one of the many computerized tools on the market. He cautions that an increase in the number of critical and near-critical paths will significantly increase the probability that the project will not be completed on time. This is due to the statistical probabilities of multiple paths causing failure.

As with many sections of the book, he provides lists of general items to use when planning.

Key Ideas: Schedule Risk
Estimates with wide variations should be investigated to insure they are thoroughly defined
Identify source of estimates, especially if not empirical
Identify high risk dependencies
Compare estimates to historical values looking for large variations
Pull risk forward to allow time to react
Identify all potential critical paths
Note project duration risk

Resource Risk

The final category of risk is the resource risk. The top five (ten are provided in the book) are:

- Outsourcing delays
- Lack of funds
- Attrition of resources
- People joining the team late
- Scarcity of skills

When planning one must determine the skill set required and identify and reserve the people with those skills. As with schedule sequencing he recommends computerized tools to properly look at staff bading. The loading will need to be compared to other project's needs and resource availability. Conflicts need to be resolved and documented since this also indicates inherent risk in the project.

Outsourcing, the primary cause of resource induced delay in the database, is mitigated best by thorough planning, proper RFP (Request for Proposal) generation, a clear and succinct contract and monitoring the subcontractor for progress. Although incoming inspection is required, this is too late to mitigate risk.

Funding issues were small in the PERIL database but the sources of error are due to overlooked staffing, travel, training and equipment costs. Funds for all of the anticipated financial outlays must be defined and planned for early in the project planning cycle.

Throughout the sections that discuss scope, schedule and resource risk, Kendrick repeats that analyzing these items must be iterative since they are the primary constraints and changes in one will affect the others.

Key Ideas: Resource Risk
All skills required to finish the project must have a named resource
Do not over commit staff
Identify tasks with unreliable resource estimates
Identify all understaffed tasks
Document all outsourcing risks
Build in funding for training, equipment and travel
Determine the complete project cost

Constraint Management

When controlling project constraints it must be understood that only two of the three constraints can be defined, the third will be determined by the other two. It should be determined which of the three (resource, scope or schedule are the controlling constraint and which is the most tolerant to change. Determining this and insuring the stakeholders understand the consequence of this is of utmost importance.

If scope is the least important, determine methods to achieve the most for the customer while using fewer resources, trim low priority scope, suggest alternative solutions to the problem being addressed and look for reusable components from other projects.

For resource constraints look at cross-training staff or training new people. Outsourcing is an option but introduces significant risk.

If schedule constraints are an issue it is possible to use schedule float. Also carefully analyze the schedule for tasks that can be overlapped. If they exist, consider defining these tasks with more granularity. Lastly, if the funds are available, add more resources to try to compress the schedule. All of these introduce their own risk.

Continually analyze the project for other risks. Kendrick provides a general list of risk categories to assist in brainstorming, analyzing historical data or previous projects.

Key Ideas: Constraint Management and Risk Discovery
Align project plan and objectives
Document project priorities
Identify project alternatives (mitigation)
Explore other, less risky, options for the project.
Remove unnecessary project scope
Document risk and impact of proposed changes
Use all means to identify unknown project risk

Quantifying and Analyzing Activity Risk

The key to assessing risk is determining the probability and impact of the risk and knowing when these values cannot be determined. Trying to make quantitative decisions from qualitative data is not sensible. To this end, Kendrick describes some standard and accepted methods of presenting non-quantifiable risks for the project. A majority of his time is spent discussing two methods of quantitative analysis—two dimensional bubble charts and PERT analysis. As in previous areas in the book, he presents beta-distributions of risks and lays the groundwork for the understanding of why simulation products are important when analyzing risk in complex projects. He presumes a modicum of knowledge of statistics but no knowledge of beta-distributions.

Key Ideas: Quantifying and Analyzing Activity Risk
Determine probability and impact for each risk
Use qualitative methods to prioritize risk
Apply quantitative analysis to specific risks
Do not over complicate PERT analysis

Managing Activity Risk

Risks fall into three broad categories—controllable known, uncontrollable known and unknown. For the former two, the risk must be understood to develop management plans. This is done using root cause analysis. As the name implies its goal is to look for the root cause of the problem and solve it at that point. Kendrick briefly discusses the use of fishbone diagrams as a tool to assist in this process. Even unknown elements should be handled in this manner, but obviously as they are incurred.

The four ways of handling risk are:

- Avoidance: Take action to avoid the risk
- Mitigation: Define actions to take when the risk occurs
- Transfer: Someone else accepts the risk (i.e. insurance)
- Acceptance: Identify the risk as acceptable and let it happen.

Determining which option to chose is primarily financial, but schedule and manpower may be involved. As a tool, Kendrick provides a number of "checklist" opinions for looking at each of these options.

Contingency planning is briefly discussed for scope, resource and schedule.

Key Ideas: Managing Activity Risk
Do root cause analysis
Use one of the three methods of handling risk—mitigate, avoid, transfer—or acknowledge acceptance
Develop contingency plans for all risk
Publicly display risks
Look for ways to prevent risk

Quantifying and Analyzing Project Risk

Kendrick highlights the common project level risks as (citing Capers Jones as the original source):

1. Estimates that are excessively inaccurate
2. A schedule that is too aggressive
3. Poor management
4. Scope creep (poor change management)
5. Large projects not staffed appropriately

Kendrick provides a suggested survey where all responses are on a scale of one-to-three and using a weighted average (1:3:9) of the responses to determine the risk of the project biasing the negative impact. Results in the 2.51 to 6.00 range are considered medium and anything above high risk.

He then returns to the PERT modeling data generated by task and explains the use, limitations and additional theory. He uses simulations to generating an approximation of the logical outcomes.

He describes two less rigorous approaches for the looking at project risk by comparison to other projects. "Project Scale" compares the total effort months of the project to other projects, the higher the ratio the more risky. "Project Appraisal" uses a process similar to appraising a house to determine risk based on other projects.

The remainder of this section is devoted to measuring projects and determining whether a project is tracking to plan or deviating in a negative manner. He discusses a number of soft techniques (qualitative judgments) and hard techniques (financial, based on actuals) to determine the "health" of the project.

Key Ideas: Quantifying and Analyzing
Generate a formal survey project risk
Determine project uncertainty based on prior estimates
Use effort months to determine project size

Managing Project Risk

This section is an assortment of activities that Project Managers may do to align the project team and the stakeholders for the upcoming project. These items include:

- A recommended list of documentation that summarizes general best practices for a project; cautioning that quality, not quantity, is the measure.
- A project start-up workshop; including its preparation and execution. The goal is to align people to the goals and educate them on the challenges.
- Determining the appropriate metrics for the project, ensuring they are not burdensome and effect behavior in a positive manner. Too often, metrics change behavior to provide better metrics not better performance.
- Setting the amounts and conditions for the use of project reserves.
- Negotiating the final objectives of the project with stakeholders to improve the chances of project success.
- Validate that all team members and stakeholders accept the plan of record.
- Describe to all team members and stakeholders the change management process and how it will be enforced.

Key Ideas: Managing Project Risk
Start the project with a kick-off workshop
Determine the appropriate metrics
Set the project reserve
Validate and adjust the objectives
Implement and use a Change Management process

Monitoring and Controlling Risky Projects

Execution of the project entails the Project Manager applying the plan, leading the team and monitoring the project status looking for trends that can indicate variations

(good and bad) in the project execution. Results of the analysis need to be communicated and adjustments made through a change management and/or issue resolution process.

Communication is imperative. Kendrick describes a variety of tools to aid in communication and notes where challenges may be become more difficult (remote projects, multiple languages, large number of stakeholders with differing goals). He also provides the obligatory how-to-run-a-meeting discussion.

Project traceability is important and Kendrick outlines the needs and requirements of a Project Management Information System. This system should be a repository for all project documents and appropriately accessible to all people.

For longer projects Kendrick suggests a periodic project review and assessment. He provides a checklist of the items this should include and show to conduct the meeting.

Key Ideas: Monitoring and Controlling Projects
Be religious on collecting status information ensure that it is only the status needed.
Monitor status and trends continuously.
Promptly address problems.
Communicate, communicate, communicate.

Closing Projects

Proper closure of a project has significant benefits for reducing risk on future projects. Whether the project is considered a success or a failure the results should be documented and reviewed. These data can then be used in future planning processes to improve planning a reduce risk.

A project retrospective should be conducted and actions taken on the suggestions to improve processes for the future. Lack of action will reduce participation in subsequent retrospectives.

Key Ideas: Closing Projects
Document the project results.
Recognize contributors.
Conduct a retrospective.

Conclusion

As mentioned in the summary, ***Identifying and Managing Project Risk,*** is most valuable to a junior Project Manager and, as Kendrick points out, can be used as a study text for the PMP exam. Unfortunately, the book has numerous typos that should have been caught in the editing process and tends to detract from the value of the content.

It contains a number of valuable "check lists" and other tools that can be used but glosses over some of the theory that might allow the reader to become knowledgeable on a subject.

The use of the PERIL database is interesting but gives the feel of objectivity where there should not be. He mentions the limitations of the data in Chapter 2, but neglects to qualify his use of data in later chapters.

His negative treatment of iterative (agile) approaches and complete neglect of Critical Chain omits numerous areas where risk can be addressed through process.

Resources and Templates

eCameron's website contains a large variety of reference materials on Project Management subjects. These include templates, processes and further discussions on a variety of topics. Please feel free to browse our site at http://www.ecaminc.com
Or contact: Todd C. Williams, Phone: 1.360.834.7361 e-mail: todd.williams@ecaminc.com

Subject	Description	Format	Location
Templates Homepage	Home page for the items listed below	Various	http://www.ecaminc.com/Templates/Templates.html
Change Management Process	A complete Change Management process document.	Word	http://www.ecaminc.com/Templates/CRProcessWord.html
Change Log Template	An Excel template for a change log.	Excel	http://www.ecaminc.com/Templates/CRLogXL.html
Change Request Template	A Word template change request form.	Word	http://www.ecaminc.com/Templates/CRFormWordDot.html
Estimation Template	An Excel template for estimating project changes.	Excel	http://www.ecaminc.com/Templates/EstimateXL.html
Executive Presentation Material	Various presentation ideas and templates for concisely expressing complex data to executives.	PowerPoint	http://www.ecaminc.com/Templates/ExecSummary.html
Generic Project Document Template	General project template to be used for specification and other control documents.	Word	http://www.ecaminc.com/Templates/ProjectDocDot.html
Meeting Minutes Template	A Word Template for Meeting minutes.	Word	http://www.ecaminc.com/Templates/MinutesWordDot.html
Risk Tool	A risk probability accumulator. Aggregates risk into a project level impact.	Excel	http://www.ecaminc.com/Templates/RiskToolXL.html
Time Reporting	Excel Template for tracking time reporting	Excel	http://www.ecaminc.com/Templates/TimesheetXL.html
Previous Newsletters published by eCameron.		PDF	http://www.ecaminc.com/ColProjMgmt/CPMIndex.html
General Discussion Topics		HTML	http://www.ecaminc.com/ColProjMgmt/ColProjMgmt.html
Recommended reading and Book Synopses		HTML	http://www.ecaminc.com/Books/BookHome.php
Seminars on Project Improvement		HTML	http://www.ecaminc.com/Workshops/ Workshops .html