White Paper presented by

Analytic Data Solutions, LLC

Title: The Requirements Driven Project Planning Process (RDP3)

Author: By John Majarwitz

Date Written: 3/5/2011

Date Posted (to website): 12/19/2011

Brief Description:

This paper addresses the project planning/creation process for software systems and technology projects and offers an approach to project planning that recognizes that planning a project is different than managing a project once the plan is in place.

The Requirements Driven Project Planning Process (RDP3)

An Integrated Project Planning Method For Information Technology Solutions

By John Majarwitz, Principal Consultant, End To End Information Solutions, Inc.

3/5/2011

The Requirements Driven Project Planning Process (RDP3)

Introduction

The Difference Between Project Planning and Project Execution

Project Management activities during the planning phase of a project is very different than project management activities during the execution phase of a project. In the planning phase, a project is being *created* whereas in the execution phase a project plan is being *followed* (and modified as needed). In many respects, the actions and skills needed to create a project are different than the actions and skills needed to ensure a project is being properly executed. This paper addresses the project planning/creation process for software systems and technology projects and offers an approach to project planning that recognizes that planning a project is different than managing a project once the plan is in place.

Weaknesses in Traditional Project Planning Approaches

In many traditional project management approaches for software systems and information technology, although there is a distinction drawn between project planning and project execution, there is not an integrated methodology to include the diverse types of resources and skill sets that are needed in the planning phase. Under the traditional approach, project planning is considered a project management activity and project management activities are primarily to be led, produced, arbitrated, and accounted for by the project manager. Yet equating project management to the project manager during the planning phase of the project is too limiting because of the breadth of requirements and architecture complexity involved in the creation of well-formed project plans for software systems and technology projects. Additionally, there is typically a lot of time pressure during this phase, resulting in limited time to think critically through the complexity.

Equating project management to the project manager during the planning phase of the project is too limiting because of the breadth of requirements and architecture complexity involved in the creation of well-formed project plans for software systems and technology projects.

The result is that the project manager often times "recycles" a prior plan (usually from MS-Project) and edits it in an attempt to fit the new project. Meanwhile, the analyst team documents requirements that are not closely coupled to the project plan while the architecture/design team documents the technology approach which is also not well integrated with the project plan. This produces a hodge-podge of deliverables, which are not well connected by the project plan. While each deliverable under this approach may be well formed, the total work package is not always as good as it should be to drive the project forward. The breakdown of tasks and activities and the estimate of effort may not be well aligned to the functional

decomposition of the business requirements and the technology requirements of the solution architecture.

This causes the project execution phase to be burdened with trying to "figure out" the project plan and results in additional project planning "rework" as a result of not doing it well enough the first time. As the project team expands in size, the problem is exacerbated because the core nucleus of the team is not able to communicate the plan with enough precision to effectively engage others.

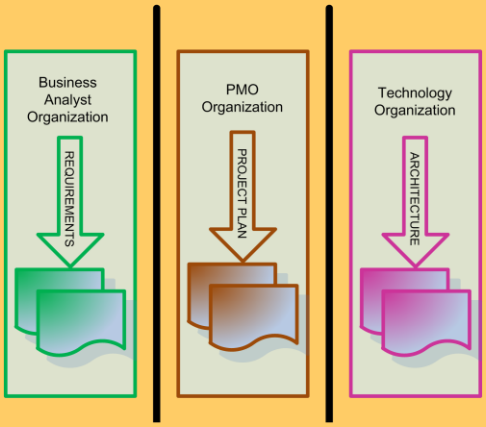
At the heart of this problem is the estimation process, which requires a thorough understanding of the business requirements and solution architecture/design but is used to determine the cost and schedule. Poor integration of work product during the planning phase leads to poor estimation. And once a project is established based on a poor estimate, much of the effort spent from that point forward will be in treating the symptoms (explaining and justifying why there are increases in cost and delays in deliverables or shortcuts in quality) rather than on fixing the root cause (going back and re-estimating the project and creating a revised plan).

Strengths Of The Requirements Driven Project Planning Process

To address this problem, the Requirements Driven Project Planning Process (RDP3) is described in this paper. This process connects the project plan to the major artifacts of the project: the functional /business requirements; the technical/architectural requirements; and the components of the solution architecture. It achieves this by ordering the work activities needed to create the project plan and by encouraging collaboration between the three main resource types needed to assemble the plan: the project manager; the analyst; and the architect. Each resource type is responsible for specific areas of advocacy while working with the other resource types in a peer based setting. This creates a positive "tension" as the advocacies are weighed and reconciled during the process. Tradeoffs, assumptions, and priorities are documented as part of the

The Impact Of Organizational Structure

Larger organizations tend to structure themselves along functional lines. This creates strong affinity within the function and looser affinity between functions.



During the execution of projects, agile methods try to address this issue by creating small cross-functional teams in order to have stronger team affinity. One way to look at RDP3 is as a means to create stronger team affinity between the Analysis, Project Management, and Architecture functions during the project planning phase. This is done by having the RDP3 process unite the organizations rather than suggesting alternative organizational structures.

process and scenarios can be observed that match various project alternatives such as the additional effort required based on expanding scope or the impact of a waterfall development process to total cost and schedule. Altogether, RDP3 allows for better informed decisions to be made by the project stakeholders prior to investing significant resources in the development of the project.

Altogether, RDP3 allows for better informed decisions to be made by the project stakeholders prior to investing significant resources in the development of the project.

This approach is tied to the general principles established as part of the project management discipline. It offers a practical means to achieve these principles based on a step by step process. As part of the PMI's PMBOK 4.0, the project model is characterized by the evolution of the original "triple" constraint, which has evolved to recognize six factors that need to be balanced, as illustrated in the diagram below [1,2]:

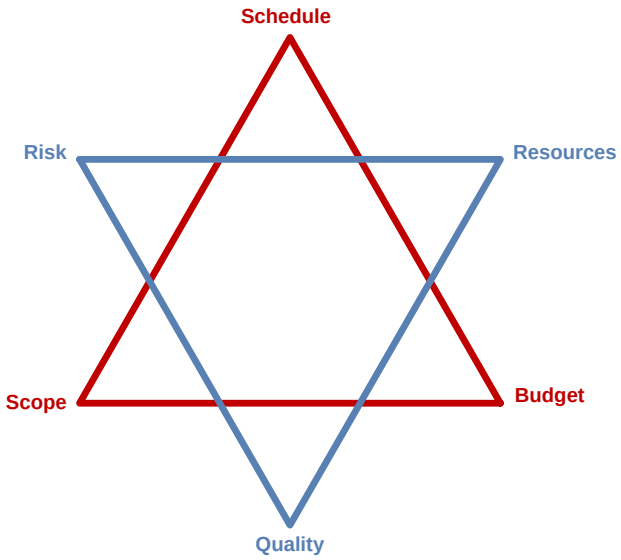


Figure 1: The Six Factors To Balance in Project Management[3]

In this diagram, the classical triple constraint is shown in red, acknowledging the balance between Scope, Budget, and Schedule that all projects need to achieve. Overlaying this is the set of primary underlying factors that directly influence the triple constraint, and correspondingly must be taken into account when managing the triple constraint: Quality, Risk, and Resources. This model is generally well accepted and helps govern the basic set of project management operating principles for most Information Technology divisions in corporations today.

The challenge with these principles is not the model itself, which is an excellent characterization of the goal of project management. Rather, the challenge lies with how to go about producing such a model for a given project. Each factor, in and by itself, is complex, dealing with a variety of underlying

components. There are also interdependencies between the factors which cause additional complication as the factors are adjusted. Trying to define and manage all six factors at the "same time" during the project planning process is ineffective because of this complexity and interdependency.

RDP3 addresses these complexity and interdependency issues by separating these factors and describing the activities and deliverables that are produced to account for each factor. By using a defined method that first identifies all the work steps and then ties the work together, an integrated project plan is produced which is cross referenced to business and technical requirements and also cross referenced to the solutions architecture. This "dual path" cross referencing establishes end to end linkage and traceability between the activities and tasks on the project plan and the purpose they fulfill in meeting the requirements and conforming to the solution architecture. This allows the properties of necessity and sufficiency to be validated for the project plan. All the tasks included in the plan are *necessary* because they reference requirements and solution architecture components (there should not be any tasks that have no references) and the project plan is *sufficient* because it fulfills all the business and technical requirements (there are no requirements that are not referenced) and it fulfills all the components of the solution architecture (there are no components that are not referenced).

By using a defined method that first identifies all the work steps and then ties the work together, an integrated project plan is produced which is cross referenced to business and technical requirements and also cross referenced to the solutions architecture … This allows the properties of necessity and sufficiency to be validated for the project plan.

An overview of RDP3 is provided in the next section. The methodology for performing the work and creating the deliverables that describe each of the six factors is referenced from this overview and described in separate documents.

Overview of the Requirements Driven Project Planning Process

RDP3 is summarized in the following sections, based on organizing the work according to the original three primary factors from the PMI triple constraint and the three subsequent derivative factors. The primary roles in RDP3 consist of the Analyst, the Architect, and the Project Manager, working in collaboration, with each role responsible for specific advocacies, meaning that each of these roles is responsible for assuring that the needs they represent are accounted for in the project plan while also working collaboratively with the other two roles to assure the project plan is balanced, comprehensive, and well formed. The diagram below shows these three roles overlaid upon the PMI triple constrain diagram:

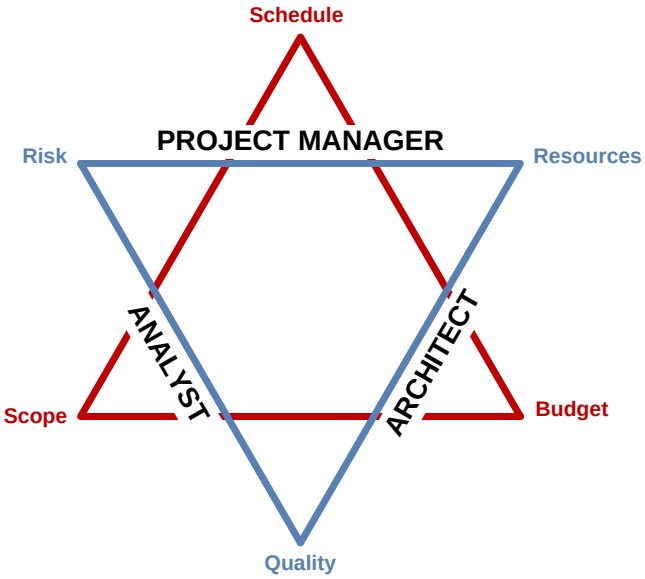


Figure 2: The RDP3 roles and advocacies overlaid on the PMI triple constraint

The Analyst role is an advocate for Scope, assuring that the project is governed only by the necessary business drivers as elaborated by the business requirements, balancing Quality and Risk factors.

The Architect role is an advocate for Budget, assuring that the estimate of effort and the types of resource skills needed are sufficient to develop the solution, balancing Quality and Resource factors.

The Project Manager role is an advocate for Schedule, assuring that the timeframe for delivering the project is achievable, balancing Resource and Risk factors.

All three roles are highly interdependent and the intent is to have a system of checks and balances in place through the advocacy responsibility of each role.

Scope

The starting point for the RDP3 begins with Scope. Of all the factors, Scope is the most important because it influences every other factor. It also provides an excellent forum to establish the right kind of

working relationship with the project stakeholders/customers, although this topic is beyond the scope of this paper. The RDP3 establishes scope in several areas: the functional business requirements; the technical system requirements/constraints and the solution architecture. Together, the information from each of these areas is synthesized to create a high level work breakdown structure that will serve as the model for how all work within the project will be tied together.

As part of the functional business requirements, there is typically a requirements document that is created by business analysts working with business stakeholders. This document can be fairly high level (bullet points of business needs) or fairly detailed depending on the type of project. But it must be complete in the sense that it accounts for all the major business functions the solution is responsible for. These business requirements are categorized, enumerated, and then referenced by the work breakdown structure.

The technical system requirements and constraints are usually defined by a systems analyst or a systems architect and are gathered from a variety of sources. This establishes the non-functional requirements, including performance and availability. And this also usually constrains the technology platforms based on organizational standards. This is usually documented as a supplement to the business requirements and hence is sometimes referred to as supplemental requirements. These supplemental requirements are categorized, enumerated, and then referenced by the work breakdown structure.

The solution architecture is described in a two stage process with the first stage being a logical architecture describing the functions of the overall solutions and how they are related in terms of data flow and control couplings. The second stage maps the logical architecture to the systems and technology platforms that will be used in order to create a systems component based architecture. These components implement the functions described in the logical architecture and are referenced by the work breakdown structure.

RDP3 sets guidelines for the organization of the work breakdown structure that are based on using the functional requirements to identify activities that describe system features and using the solution architecture to identify activities that account for shared components and services.

While the requirements and the solution architecture can be produced quasi-independently, the key part of the RDP3 is the joint evaluation of both of these documents when assembling the work breakdown structure. For any given project, there are many possible ways to express the work but RDP3 sets guidelines for the organization of the work breakdown structure that are based on using the functional requirements to identify activities that describe system features and using the solution architecture to identify activities that account for shared components and services. The RDP3's work breakdown structure is also organized based on the major software development lifecycle phases that will be used in the development process (e.g., Analysis, Design, Build, Test, etc.). The resulting work breakdown structure is a grid laid out as shown in the schematic diagram below. This is usually documented as an Excel spreadsheet.

More information is available in the *RDP3 Scope Factor - Guidelines, Practices, and Standards* document.

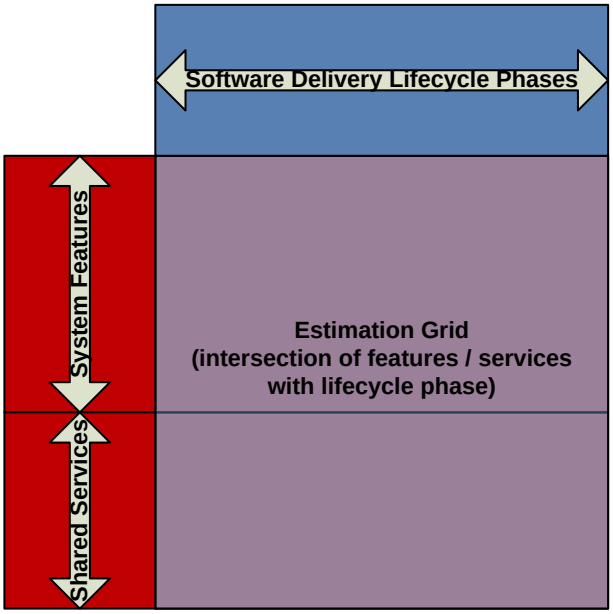


Figure 3: Grid Based Organization of The Work Breakdown Structure

Budget

Budget for the RDP3 is driven by effort based estimation, driven from the work breakdown structure established from the Scope factor discussed above. The emphasis is on evaluating the work that is being requested to be performed and identifying the effort required to accomplish it. This is a difficult task and it is often biased by pre-conditions that establish available funds before the work breakdown structure and estimate are ever performed! The RDP3 establishes guidelines that de-emphasize external budgetary constraints during the initial estimation process. This is crucial for establishing independent estimation of the actual work to be performed regardless of the availability of funds. Once the estimate is initially created, the RDP3 provides approaches for evaluating alternatives based on funding limitations, but this is a step to be performed *after* the initial estimate so that artificial bias is not introduced into the estimation process.

One of the key principles of RDP3 is to establish multiple estimates as independently as possible and then compare these to see their degree of consistency, which can help predict how reliable the estimate is.

There are several estimation techniques that RDP3 employs, covering top-down, bottom-up, meet in the middle, history based, expert based, and swag ☺. Industry based approaches such as function point analysis, COCOMO II[4], or others can also be used as well as any internal estimation methods your company may use. Estimation is a difficult process and there is always uncertainty (variance) involved in any estimate. One of the key principles of RDP3 is to establish multiple estimates as independently as possible and then compare these to see their degree of consistency, which can help predict how reliable the estimate is.

Once the initial estimate is created, the work breakdown structure is tied to the effort required to fulfill all stated tasks across all stated phases of the development lifecycle. The cross-references to the requirements and solution architecture are double checked as part of the estimation process. Assumptions, dependencies, and open issues are also identified and recorded as part of this process.

The RDP3 defines effort in units of time rather than in units of dollars at this point. This is because it is more precise to review and evaluate the work in terms of how long it will take to do something rather than in how much it will cost. Assumptions need to be made about the personnel and their skill sets in order to produce the estimate as well as around the definition of a "unit of labor". The RDP3 establishes guidelines for these.

At a later point in the process, the effort is converted into units of dollars to establish project cost, but this occurs only after the scheduling factor is addressed.

More information is available in the *RDP3 Budget Factor - Guidelines, Practices, and Standards* document.

Schedule

Once the effort is identified for the activities/tasks in the work breakdown structure, the activities need to be scheduled. There are several considerations that come into play. First, dependencies between tasks need to be accounted for. Then the type of development method needs to be taken into account. For example waterfall methods result in ordering of tasks by development phase whereas iterative methods allow for a greater degree of concurrency between phases. Additionally, the assignment of tasks to resources needs to be accounted for including level smoothing and resource ramp up and ramp down. Finally, the degree of "aggressiveness" of the schedule needs to be considered. All of these considerations are addressed by RDP3.

Once scheduling is complete and there is a level smoothed resource plan then the total dollar cost of the project can be determined based on a level smoothed plan and cost rates based upon the resource type.

More information is available in the *RDP3 Schedule Factor - Guidelines, Practices, and Standards* document.

Resources

Scheduling and Resources are highly interdependent. A project schedule that simply orders tasks and sets begin and end dates without assigning resources is not as useful since the plan cannot drive work assignments. Resource considerations also influence the estimation process as assumptions are made (either explicitly or implicitly) on the strength and breadth of skills and experience.

RDP3 handles resource considerations at two levels: the initial level is person independent and sets "abstract" resources types for purposes of establishing skills, experience, allocation to task, determination of start and end dates and level smoothing; the subsequent level addresses the assignment of personnel to fulfill the roles required by the abstract resource types. By splitting resource

considerations into two levels the planning process can move forward while deferring the need for personnel selection, if necessary.

More information is available in the *RDP3 Resource Factor - Guidelines, Practices, and Standards* document.

Quality

Quality considerations are included in many ways. Task identification in the work breakdown structure, estimation of work effort, technical requirements, degree of sophistication of the solution architecture, software development lifecycle and strength of resources that are identified as needed are some of the key areas where quality assumptions are being made. Additionally, the stability and accuracy of the requirements are probably the most important consideration since this captures the business problem to be solved.

It is common for there to be a certain degree of "cognitive dissonance" during the planning process with respect to understanding the quality of the solution relative to the effort required to deliver the solution … This leads to the tendency to oversell quality as a justification for cost.

It is common for there to be a certain degree of "cognitive dissonance" during the planning process with respect to understanding the quality of the solution relative to the effort required to deliver the solution. Since this is the planning process for the project, the "work" is not yet real and it is easy to rationalize that quality will be present, whereas the "cost" of the project tends to be more real at this point in time since one objective of the planning process is to (ultimately) bind the stakeholders to a financial commitment. This leads to the tendency to oversell quality as a justification for cost. The RDP3 provides guidance on handling this type of issue, which is part of overall expectation setting.

More information is available in the *RDP3 Quality Factor - Guidelines, Practices, and Standards* document.

Risk

Awareness of risk needs to be consciously present in the minds of the project managers, analysts, and architects who collectively conduct project planning processes. At this point in the project, there are so many possibilities for issues that it can be almost overwhelming to consider risk. It can also be relatively unproductive to focus too much discussion on possible risk scenarios that are unlikely to occur. However, a good risk scorecard can help add perspective to the project, especially as it might compare to similar projects and experiences.

Equally important is the documentation of assumptions and open issues that are identified during the planning process. Although assumptions and open issues are a common aspect of good project management during project execution, it is frequently overlooked as a disciplined practice during the project planning phase. Ironically, "project management" of the project planning phase can sometimes

be lax. RDP3 supports risk mitigation by providing a risk scorecard and by establishing a method for documenting assumptions, questions, and open issues during the project planning activities that will follow through to the project execution phase.

More information is available in the *RDP3 Risk Factor - Guidelines, Practices, and Standards* document.

Conclusion

Information technology projects require careful planning. RDP3 establishes a structured and repeatable method for creating technical project plans for complex IT projects. Planning well at the beginning of a project can be invaluable, as the best problems to solve are the ones that do not occur in the first place.

At an organizational level, establishing a repeatable method for planning projects also provides an easy means to review and decision proposed project plans since the artifacts and deliverables produced by the method will be consistently formed for all projects.

About End To End Information Solutions

End To End Information Solutions (E2E) specializes in information technology solutions, architecture and design for data warehouses and distributed systems including large scale data management and service oriented architecture projects, application and project assessments, project planning, solution architecture, data architecture, performance tuning and project execution and delivery. Contact us at 904 612 1124.

References

1. *The Project Management Book of Knowledge (PMBOK), Edition 4.0. The Project Management Institute. ISBN: 1933890754 / 9781933890753.*
2. *http://www.projectsmart.co.uk/pmbok-guide-fourth-edition-changes-chapter-by-chapter.html*
3. *http://en.wikipedia.org/wiki/project_management_triangle*
4. *Constructive Cost Model II (COCOMO II). University of Southern California Center For Systems and Software Engineering. http://sunset.usc.edu/csse/research/COCOMOII/cocomo_main.html*