

## PM WORLD TODAY – PM TIPS &amp; TECHNIQUES – MARCH 2009

## We Didn't Do a Good Project Schedule

*By Lonnie Pacelli*

I can remember vividly my very first project schedule. My manager gave me the mission statement and an overall timeframe he thought it should take for me to complete the project. I diligently broke the schedule down to lower levels of detail. I continued to divide the overall timeframe among the tasks and assigned people to the tasks. I worked for days on end with my face buried in a computer screen developing the schedule. What I ended up with was a horrendously detailed project plan that had no logical dependencies identified, people being asked to complete 40-hour tasks in 15 minutes, and some people being asked to work 200 hours per week to get their work done. But by golly, the schedule met my manager's timeframe request.

Sadly enough (for me), this is a very true story but one that I don't think is too terribly uncommon. It's pretty easy to ignore reality at times when you're developing a schedule and to skip some fundamental steps in completing your schedule. You may get everything to look good on paper, but the result may deviate significantly from reality.

***How it happens:***

**The project schedule was either too detailed or not detailed enough** – A project schedule is only effective when it is able to help you know that everything is on track and that you're going to be able to complete the work on time. When your activities are at too high a level, you risk losing accountability, missing out on key dependencies or expose yourself to "90% complete syndrome" when the team reports progress that is not real. When your activities are at too low a level, you can frustrate your team members by unduly micro-managing them, creating a greater administrative headache for yourself, and confusing the team with an excessive number of activities to manage. Either of these can spell schedule slippage and can severely impact successful project completion.

I've learned to use two rules of thumb when defining the appropriate level of detail for a project plan:

- Can the activity be assigned to a single person to complete the activity?
- Can the activity be completed in less than 40 hours?

Let me explain my question rationale. In the first question, I have found that explicit, clear lines of ownership are vital to ensuring that activities are completed. Whenever there is an activity

assigned to “the team” or some other group of people there is no single point of accountability thus no one truly owns the task. Therefore each and every task should have a named person that takes the heat if the task isn’t completed on time.

In the second question, the more time an activity is given to complete the greater the likelihood that you will be surprised at the last minute that the activity was not completed on time. I’ve gotten burned way too many times on an activity getting to 90% very quickly then taking twice the amount of time to finish the last 10%. Now, it’s not that I’m a distrustful person or that I think that people are overtly trying to deceive me. No one wants to miss a deadline and thus will continue to report that they are on target and hope that everything falls into place if things start going awry. Sometimes it works; sometimes it doesn’t. I prefer to leave as little to chance as possible. So, I’ve zeroed in on a 40-hour rule of thumb because it allows you to divert train wrecks on time but also doesn’t micromanage the team member. Depending on your environment, you may want to use something other than 40 hours, just be definitive and consistent in what you use.

**The project schedule didn’t correctly address dependencies between tasks** - When designing your project schedule, you need to keep in mind how those activities relate to other activities and define them accordingly. Establishing clear dependencies between tasks and having a true understanding of the critical path (the string of tasks that are the longest point between the start and finish of the project) is in my view one of the most important components of your project schedule. As you’re designing your schedule activities it’s helpful to keep dependencies clean by defining clear finish-to-start relationships. There are ways to accommodate this using most common project management software packages, but I recommend keeping your dependencies simple to understand and manage.

**The project duration was too long** – When designing your schedule, keep specific focus on the length of time that you go between celebrating successes. I’ve become a strong advocate of keeping project phases to no longer than three months in duration. This is not to say that, if you are implementing an Enterprise Resource Planning system that you should try to do the entire implementation from software selection to implemented system in that three-month timeframe. What I am saying is that you should phase the project in such a way that there is a defined beginning and end to the phase within three months. Why would I say such a thing? Simple; people (particularly management) live in a short attention span theater world and over time will become discouraged and lose interest if a project drags on too long. So ok, you’ve figured me out, I’m just breaking a \$10 bill into two \$5 bills, but I’ve seen teams perform better when they are able to have mini celebrations at the successful end of each phase because at any point in time the end of the phase of work is no more than three months away. This also gives the team and management logical review points to look at the project’s mission and ensure that it is in sync with management’s current priorities.

**Some of the tasks didn't produce useful deliverables** – When you're defining your project schedule, make sure you're continually asking yourself these questions:

- What is the deliverable that will be produced out of this activity?
- What will it look like?
- What happens if we don't do it?

If you don't have satisfactory answers to each of these questions, then seriously consider whether or not the activity is necessary. Remember, every activity that you do should be getting you one step closer to successful completion. If you can't articulate what the activity is supposed to produce then chances are you don't need to do it.

**The team didn't understand the plan** – Your project team needs to have complete buy-in on the tasks, the durations, the team assignments, the dependencies, and the deliverables. What I've seen work well is doing shorter, more frequent informal reviews with team members while you're developing the schedule. I've seen project managers hold themselves up in an office or conference room for days on end, emerge from their cave with the schedule to end all schedules, and then have the other team members storm the Bastille because they don't see how they're possibly going to be able to accomplish what the project manager expects (recall my opening story about my unrealistic schedule). It's days like those that the project manager wonders why he or she didn't take over the family delicatessen instead of doing this stupid project manager job. Get the buy-in along the way; it helps you avoid rework, allows the team members to feel more included in the process, and will produce a better quality plan that the team will believe they can achieve.

### ***Warning Signs:***

**Tasks aren't getting done on time** – Chronically missed deadlines on tasks can be due to unrealistic task durations, poorly defined dependencies, poorly defined work assignments, or project distractions. Diagnose the reasons for the missed deadlines immediately before the snowball rolling down the hill turns into an avalanche.

**Tasks assigned to "the team" or some other group of people aren't getting done** – Any time that a specific name is not attached to a task, it is easy for team members to assume that someone else will do the task because *no one is specifically held accountable for task completion*. If you want things to get done, make sure that there are specific names beside each of your task and that each team member feels personal accountability for getting their work done.

**Team members aren't aware that they were supposed to be working on a task** - It's an ugly

situation when you're getting status updates from team members and a task that was supposed to be completed last week wasn't even started because the team member didn't even know they were supposed to be working on the task. Make sure that work assignments are crystal clear and that team members know what tasks they are supposed to have completed by when.

**Team members are confused as to what they are supposed to produce for a task** – So you assign a task to a team member and the day that the task is due the team member produces a deliverable that looks exactly *nothing* like you were expecting it to look like. The deliverable now needs to be reworked which means other tasks are going to slip as a result. Be clear about what deliverable needs to be produced and ensure that you and the team member have a common understanding of what it needs to look like.

***Turning it around:***

**Get real with the schedule, and fast** – Don't delay; get the schedule in shape quickly making sure that you've defined all the right tasks, durations, dependencies, and resources to get it done. More importantly, don't go into a cave for days on end to do; make sure you are getting input on the schedule frequently to avoid an unrealistic schedule.

**Do focused reviews with team members** – On some projects, I have developed supplemental documentation which explains key tasks that might be confusing and even gone so far as to produce a template of what the deliverable out of the task needed to look like. I prefer to do mini reviews as the plan is being developed to ensure that the team is putting their thumbprint on the plan and that any confusion points can be addressed early.

**Keep dependencies simple** – While it's great to clearly understand dependencies between tasks, I've also seen plans that are overly complicated because the project manager built in serial dependencies between tasks that could actually be performed in parallel. This could be due to an assumed dependency between the tasks or because the project manager is anticipating that one person will be doing both tasks. Before defining a dependency, put rigor into making sure that the tasks are truly reliant on being performed serially.

**Highlight tasks which are due in the next 1-2 weeks** – I've learned through experience that solely relying on the project schedule as the communication vehicle for a project team is not always the most effective way of ensuring tasks get done on time. Depending on the experience of your team, they may not understand how to read the schedule and may miss some key tasks that need to get done. I've learned to use either status reports or e-mail reminders to individual team members reminding them of what they need to do and when it needs to be done. It puts a bit more work on the project manager, but it better ensures the team member knows what needs to be done by when.

***Take aways:***

- Make sure all your lowest level activities have a sole owner and are no longer than 40 hours in duration
- Break your project into phases that don't exceed three schedule months
- Know the critical path of tasks through the project and keep clean finish-to-start dependencies between activities
- Make sure that your activities have associated deliverables that are relevant
- Ensure the team buys into the plan along the way; don't do a grand reveal when the plan is complete.
- Remind and highlight team members of tasks needing to be completed within 1-2 weeks

This article was excerpted from the book:

**[The Project Management Advisor - 18 Major Project Screw-Ups And How To Cut Them Off At The Pass](#)** (Prentice Hall, 2004) <http://www.project-management-books.com/Advisor/>.



## About the Author:



### ***Lonnie Pacelli***

*Author*



**Lonnie Pacelli** is an internationally recognized project management and leadership author and consultant. Lonnie has over 20 year's leadership expertise as an executive, project manager, developer, tester, analyst, trainer, consultant, and business owner. During his 11 years at Accenture he built leadership expertise consulting with many Fortune 500 companies including Motorola, Hughes Electronics, and Northrop-Grumman. During his nine years at Microsoft he continued building leadership expertise through development of some of Microsoft's internal systems, led their Corporate Procurement group, managed their Corporate Planning group, and led company-wide initiatives on Continuous Fiscal Improvement and Training Process Optimization. He has successfully implemented projects ranging from complex IT systems to process re-engineering to business strategies. Read more about Lonnie, subscribe to his newsletter, see his books and articles, and get lots of free self-study seminars, webcasts and resources at <http://www.leadingonedge.com>.