Work Breakdown Structure: Project Design Issue or Clerical Task?

By Dick Billows, PMP, GCA

Summary: The work breakdown structure is often the launching pad for project failure because people think it is a "to do" list and that bigger is better.

It's amazing how often people ask us questions like: "How many tasks should this project have?" or, "How much detail should I have in the project plan?"

The usual mistake PMs make is to lay out too many tasks; listing hundreds or thousands of things-to-do starting with the first thing and ending with the last and listing one-hour or even 15 minute things because we don't want to forget anything. It's easy to get caught up in the idea that a project plan should detail everything everybody should do on the project. That, so the thinking goes, will protect us from people forgetting or skipping an item because they are lazy or sloppy. It also frees us from having to rely on the thinking or creativity of the team; they can just put their heads down and follow the "to do" list.

All this springs from the screwy logic that a project manager's job is to walk around with a checklist of 17,432 items and tick each item off as people complete them. This view is usually linked with another fallacy - namely, that the project plan should be a step-by-step procedure for doing everything.

Sponsors encourage these fallacies by marveling at monstrous project plans because they make it seem that the PM has "thought of everything and is really in control of this project." Soexecutives often judge the quality of the project planning by the size of the WBS.

The result of these fallacies is that PMs produce project plans with hundreds or even thousands of tasks. Many of them have durations of a few hours. Others may be more brief like a 1 hour meeting. This may sound good to you and these arguments persuade many project managers to take this approach. But there are some important questions we need to ask, such as:

Does this level of detail give us better control and lead to successful projects?

What is the impact on the project team's performance?

Micromanagement Works Some of the Time

First, the laundry list approach leads to, and even encourages, micromanagement of the people working on the project. That consists of PMs making all the decisions, discouraging team member creativity and innovation and insisting that everyone follow the "to do" list exactly as written.

Micromanagement is appropriate when you have slackers and nincompoops working for you, but few project teams are composed entirely of poor performers. The majority of your project team members will not thrive under micromanagement. This style tends to encourage dependency on

the project manager rather than independence. It also creates team members who have no accountability for results; all they have to do is follow the list of activities. Micromanagement works on small projects where the project manager knows more about all the work than the team does. It also works on very simple projects like cleaning out all the trash from a vacant lot where we can "see" everything that has to be done and all the work is physical with little thinking or creating required. Activities like, "Carry the three tires, one at a time, to the truck and load them into the back of the truck," cover everything that must be done. That activity is also a reasonable basis for estimating the work which will allow us to track progress and tick off that the tires are in the truck.

Maintaining the "To Do" List WBS Takes Forever

The laundry list approach is hard to maintain because this WBS requires a lot of maintenance. Every time a micro-task changes we need to update the WBS. That can require dozens of changes each week by the PM. The people working on the project have to report on 5-15 tasks each week. That volume decreases the odds of receiving accurate and timely status reports. The PM, with or without clerical support, has a great deal of data entry to do to input all this status data.

The inevitable result is that tracking falls behind and so does updating the schedule. There are simply not enough hours. Usually within a few weeks the PM stops updating the schedule because it takes too much time. This may sound like a stupid and improbable reaction, but we see it with alarming frequency even on large and important projects. The logic is, "No one is looking at all that detail anyway, so why spend all that time to catch up?"

Micromanagement Doesn't Work Most of the Time

Micromanagement does not work on projects that require complex judgments and creative thinking. On these projects, much of the work is cerebral and it is impossible to specify everything that must be done. More importantly it is stupid to try and specify it. Say we have task like "design the payment input screen (GUI)for the billing system." That relatively small task will require:

- meetings to gather information about requirements from the people who will use it
- listing all the required information for the GUI
- thinking about how to arrange the data elements on the screen for data entry efficiency while still leaving enough space for really long names
- writing a layout document for the screen
- meeting with users to get approval
- meetings to secure approval of the rough design
- more thinking about the requested modifications
- and so on.

We could list all those activities and more in the WBS but do we want the engineer doing the work to just follow all the steps? Do we want the engineer to ignore a payment GUI that just

won a design competition? How about if another analyst had several payment GUIs for another division? Clearly we do not want the engineer to skip all those opportunities to do it faster and better and just blindly follow the "to do" list. No, we want to engineer to figure out the best way to do the design.

We also need to think about the motivational impact of micromanaging the engineer with an activity list like the one above. How will the engineer respond to having all those micro-tasks that cover a week's work? Will we get the engineer's best effort? Or will we get an engineer who simply follows the "to do" list with little concern for the outcome.

Accountability for End Results Always Works Best

Professionals who manage projects for a living and use the best practices in project management all specify that the WBS should be composed of deliverables, not the activities needed to produce them. With this approach, the engineer is assigned the end result of "department manager signs off on the GUI design for entering payments." We leave it to the engineer to schedule the meetings, discuss changes and persuade the user manager to sign off. We base our estimating on that one deliverable and that's what we track each week.

Assigning accountabilities for end results gives us a smaller WBS, easier reporting of actuals and many fewer changes to make to keep the schedule current. We also get more ownership of the results from the engineer who is given an achievement to deliver. Best of all, when we base our assignments and monitoring on well conceived and measurable achievements, no one loses sight of the desired end result.

Summary

Your work breakdown structure (WBS) is your design for making assignment, holding people accountable and monitoring process. Done properly, your schedule will be easy to maintain and your people will be responsible for their work product. Learn to do a WBS properly as part of our basic and advanced project management courses. You'll learn how to break down work into "packets" of achievements for which you will hold people accountable.