



Going Forward while Looking Back!

Applying one Project Manager's Lessons Learned

Larry Dribin, Ph.D.

December 10, 2014 - V1.0

larry.dribin@pearlstreetgroup.com

847-807-7390

June 2011, I spoke at PMI Chicago on Cloud Computing so before Lessons Learned a Cloud Computing Update

- ◆ Public Cloud Computing is rapidly growing, especially among start-ups:

It is used for servers and for key applications

- The Good – low cost entry for high quality solutions
- The Bad – piecemeal solution with many unintegrated solutions

- ◆ Slower mainstream adoption than anticipated in 2011

Several Reasons:

- ◆ Tremendous competition & confusion
- ◆ The rise of Private Clouds
- ◆ Security Issues – Many companies think that the Cloud is more vulnerable to online security issues than their own Data Centers



Sony, Target, Chase, Home Depot, & Jewel – Are they right?

Lessons Learned

- ◆ Are information gathered during a project that can improve the current project or future projects
- ◆ Often problems on a project are the result of forgetting or ignoring several key lessons learned from prior projects
- ◆ Should be a form of continuous improvement
- ◆ Can occur anytime during a project, but are often conducted at the end of a project – Post Project Reviews
- ◆ Use Lessons Learned to improve the current project, not just at the end of the current project
 - Project teams often disperse to other projects
 - It is difficult to utilize “lesson learned” from one project to another

Agile Scrum Retrospective – occurs after every sprint as a means of improving the current project

Example - F500 PMO - Don't do as they did!

◆ Background

- PMO at Large Fortune 500 did a Lessons Learned – Post Project Review of 12 projects finished that year
- 4 successful project and 8 unsuccessful projects

◆ Analysis

- All 4 successful projects followed the companies “complex/heavy” project management methodology
- Key comment the methodology was heavy which added to project cost
- All 8 unsuccessful projects did not use the methodology – use a hodge-podge of methodologies including a couple of agile variants

◆ Result – Nothing Happened

- No mandate that projects use the company methodology
- No change, except a side project about a light project management methodology that was never implemented

Lets get started and look at my Top Ten list of Lessons Learned.

10. Manage Issues & Risks

- ◆ Issues are events/situations which happen during the project which are not on a project plan – Examples:
 - Can't find a time to meet with a key stakeholder
 - A late deliverable
 - Require an Action Plan to resolve
 - Challenge – It is difficult to track & resolve issues
- ◆ Risks are potential events/situations which could jeopardize the project – not issues till they occur
 - You need to estimate a Risk's probability of occurrence and impact
 - Require a plan to mitigate the Risk should it occur
 - Challenge – It is difficult to estimate probability and impact of a Risk
- ◆ Example:
 - Had a complex difficult to use Issue Management System
 - Result: Team couldn't keep up with issues

09 - Use Change Control to Manage Changes

- ◆ All project have changes – not all projects manage change
- ◆ Unmanaged changes lead to scope creep
- ◆ In order to minimize scope creep you must manage changes
 - Agree on scope
 - Define a simple and flexible Change Control Process
 - Use the Change Control Process to track potential changes, assess impact, manage and determine how to manage change

- ◆ Examples:
 - One of my PM Mentors said:
 - “Always start a project with a change!”
 - On a \$2 million project had over 900 managed changes -> success
 - On a similar \$2 million project – unmanaged changes -> Lawsuit

08 - Version Control all Documents

- ◆ Generally called Configuration Management
- ◆ You need to version control both:
 - Draft Versions
 - Final Versions
- ◆ Never use Now() for the date in a MS/Word document
- ◆ If your organization doesn't have a standard convention for version control, develop your own for both Drafts & Finals
- ◆ Plenty of free version control systems available
 - SharePoint, Subversion, Git

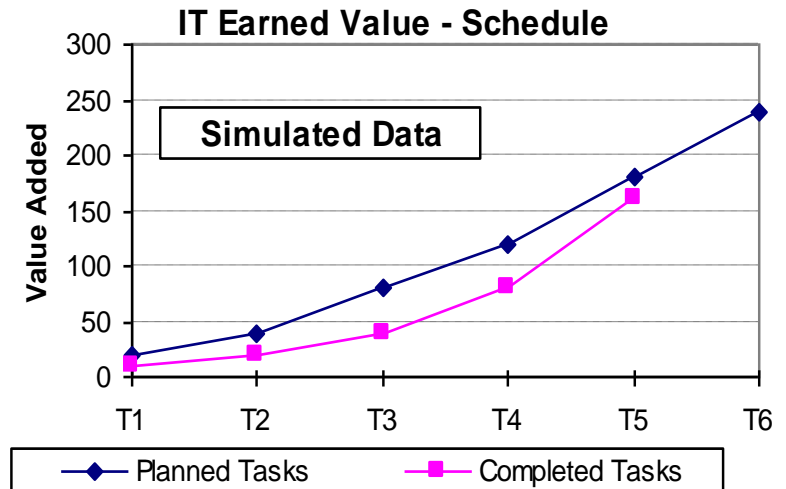
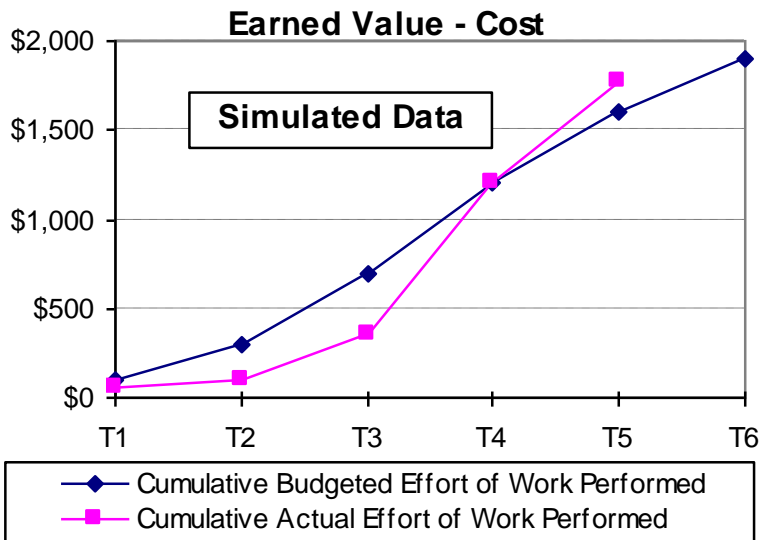
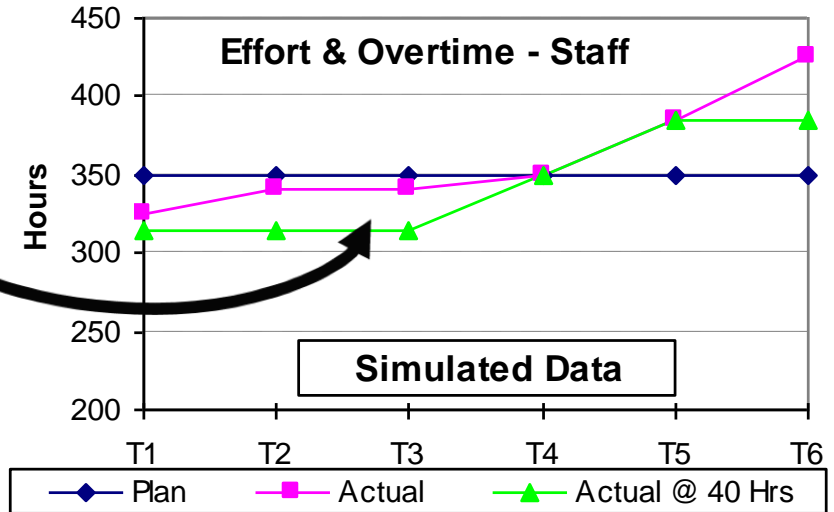
Poor Configuration Management is one of the biggest time wasters and sources of errors in a project!

07 - Set Targets & Measure Progress

- ◆ In addition to managing the Project Schedule develop key metrics with targets to measure project progress
- ◆ Measures some form of easily understood Earned Value
 - Example Agile – Burn Down Chart
- ◆ Develop a Dashboard containing multiple measure
- ◆ Look for trends not just Green/Yellow/Red status

Example Dashboard

- Project starts out understaffed
- PM tries to catch up via overtime & hiring additional resources (Green)
- Project is getting back on plan
- But starts to run over budget because of the overtime



06 - Don't Do it Yourself - Collaborate

- ◆ As project manager, you and your team are the architects of the project design – how the project will be implemented
- ◆ Don't design the project by yourself
- ◆ Collaborate with your team to design your project
 - Core project team
 - Extended Stakeholders
- ◆ Get buy-in from your team
- ◆ Work with the project solution architects to develop an easily executable and flexible project plan
- ◆ Agile proponents often depict the PM as a “lone wolf – autocratic manager” vs. the Scrum Master who acts as a team facilitator

Which role do you think best represents you?

05 - Use an Agile Approach to Planning

- ◆ PMBOK → Progressive Elaboration
- ◆ Plan projects using an Agile approach
 - The way you plan a driving vacation – Instead of planning each day in detail in advance, develop a high level plan and adjust as you go
 - Weather, traffic, unplanned side trips, stay another day, etc.
 - Add more details near term and fewer details further out
 - Don't get too detailed or you will spend too much time administering
 - Task Length – 1 to 2 weeks min / max
 - Don't be afraid to re-plan as you learn more about the project
- ◆ Example:
 - When I strictly adhere to a detailed original plan – things don't always work out
 - Often about 1/3 of the way through a project I understand what really needs to be accomplished and re-plan the project

04 - Be Ready for Unintended Consequences

- ◆ Plans do not always go as planned

“In preparing for battle I have always found that plans are useless, but planning is indispensable”. Dwight D. Eisenhower

- ◆ Be prepared for unintended consequences to your actions
 - Always look for unintended consequences
 - Be ready to quickly revise your plan, your approach, your organization or your team if you start to see unintended consequences
- ◆ Example:
 - Assigned an individual to a task who became extremely stressed at critical milestones

03 - Learn thy Organization and thy Customer

Learn everything you can about your organization:

- Business you are in
- Competition
- Strategies
- Policies and Procedures
- Culture
- Financials
- Management
- Communications mechanism
- How to get things done
- ...

Learn everything you can about your customer:

- Business they are in
- Competition
- Strategies
- Policies and Procedures
- Culture
- Financials
- Management & Buyer Network
- Communications mechanisms
- How to get things done
- ...

Learn all about the project you are on – problem & solution environment

Example:

Large Pharma – New Drug Study – Team didn't listen to experience!

02 - Availability is not a Skill Set

- ◆ Often available resources are not the required resources
 - They may not have the required skill, knowledge or experience
 - They may not fit the culture of the project team
 - May have too much skill, resulting in: *Too many cooks syndrome*
 - It takes time to on-board new resources and if resources can not be productive quickly, they may slow the project down versus speed it up

- ◆ Examples:

From my initial Project Management Mentor

- His company provided several equally qualified PMs for any project and let the customer select the PM they would like to work with
- I have been on numerous projects where I was given available resources which were not qualified to do the work

How about you?

01 - You Are Not Always Right

- ◆ You are not infallible and you will make mistakes
 - Learn from them
 - Correct as fast a possible
- ◆ Not all mistakes have the same impact, so...
 - You can make some mistakes and recover
 - Other mistakes are “fatal mistakes”
- ◆ Try not to make “fatal mistakes” for your project, your customer, your organization, your team or yourself
- ◆ Example:
 - Focused on the wrong solution
 - Forgot to measure performance
 - Don't perform robust issue management

What are some of Your Lessons Learned...

Summary of Larry's Top Ten Lessons Learned

1. You are not always right, you may make mistakes
2. Availability is not a skill set
3. Learn thy Organization
4. Learn thy Customer
5. Be ready for unintended consequences
6. Collaborate with your project team to design an agile plan
7. Set measurable targets and measure progress
8. Version Control key documents
9. Use Change Control for key changes
10. Manage Issues and Risks

Keep it simple and hold Lessons Learned sessions throughout the project not just at the end!

Going Forward while Looking Back!

Applying one Project Manager's Lessons Learned

Larry Dribin, Ph.D.

December 10, 2014 - V1.0

larry.dribin@pearlstreetgroup.com

847-807-7390