Why Development Projects Succeed (or Fail)

I find the study of developing new products fascinating. Why do the majority of development projects fail to be completed within budget and on schedule, especially projects that entail new technology? It would seem that after decades of experience in the development of new products that somehow the appropriate recipe would now have been formalized on how to prevent cost overruns and schedule slips. Why is it so difficult to solve this problem? Certainly it is not because there are no benefits as it can be readily seen that the benefits are enormous.

From reading engineering development trade journals it would seem that the solutions to reducing development schedules and associated costs could be easily obtained. However, advertised tools, process models, and paradigms to solve schedule and cost problems seem to be much more plentiful than proof of their successes. A method to address the problem of schedule and cost overruns is direly needed in today's market that increasingly demands reduced time for development of increasingly more complex products.

Although I cannot offer a magic spell to solve this complex problem, I would like to offer some lessons learned from industry leaders who have tried. Most have achieved success to some degree although none would suggest that they have fully reached their target objectives. These lessons learned are offered in the hopes that the successes and failures may enhance the productivity of your own initiatives and reduce the false starts and failures.

Project Management

Project management is critical to the success of any new development project. As project management is responsible for planning and managing the project,

1. Any process change can be an improvement if you do not **measure**

Selection of the optimum development process seems to be an easy answer for the complex problem of how to increase productivity. Although it is recognized that adoption of any defined process is preferable to none, initial gains realized through introduction of a process must then be evaluated in terms of what incremental benefits are obtained through The latest process buzzword seems to be an easy fix to a complex problem. It seems that a new process and associated tools and text book is introduced about every five years from waterfall models to object oriented design to extreme programming. This is not to say that having and enforcing a process is not important as it is an essential ingredient to achieve any sustained productivity, however, only to recognize that the management of the process is more important than assuming that selection of the perfect process will solve all problems.

There is no one size fits all perfect process for engineering development that ever has or ever will be developed. The effectiveness of a process depends on the task, the people, the technology, and a myriad of other factors. Regardless of the process it must be continuously managed and adapted to circumstances. How can one determine if the process is optimum or requires change? The answer lies in the application of measures to evaluate the effectiveness of the process. With measures the cost benefit of any process change can be evaluated, did the

overhead associated with the new tasks result in commensurate productivity and quality improvement?

Managers are motivated to make decisions to please their own managers. Although no manager intentionally makes poor decisions, a lack of knowledge often results in changes that do not ensure the desired benefits. Development managers face many decisions through the life of the project that had significant on the success or failure of the project. These decisions include scheduling, assignment of personnel, purchase of tools, accepting or rejecting proposed changes, level of reviews and testing to be enforced, project audits to be conducted, technical design decisions, etc. When management decisions are not based on actual measurements from previous experience decisions tend to be shortsighted reflecting readily observed dependencies. Examples can include improper scheduling, inadequate testing or reviews, excessive documentation detail, etc. It is easy to select new tools, more people, and only when measures are applied can optimum decisions be made that recognize long-term benefits. In the absence of measurement many decisions

2. The delays get longer when problems are not confronted (**prototypes**)

Studies conducted by Barry Boehm have suggested that the most significant factor in predicting the success of projects (completion on time with acceptable quality) is the selection of the project manager. His studies also indicate that the most significant factor that distinguishes these managers is the ability to manage project risks. This means that the successful project managers are the ones that are able to assess potential project risks and apply mitigation strategies early on in the project to reduce these risks. A summary o this risk assessment and mitigation strategy is discussed in his paper A Spiral Model for Software Development.XX

As projects will always be subject to changes in requirements, identification of unknown problems, loss of resources and other changes, it becomes essential for the project manager to continually assess project status and to redirect the project to maintain the planned schedule. According to the research conducted by Boehm this is the most significant factors in ensuring project schedule is maintained.

3. Last 10% of the job requires 50% of the effort (**product versus prototype**)

The majority of errors experienced by products after release are related to so called "user errors" and seldom used features that were not anticipated by the development team. Product designers frequently address the normal cases of use and fail to consider all of the other cases that may be experienced by the product during use such as failure cases and abnormal conditions. The effort required to address these failure case and abnormal conditions is often times at least as significant as that required to address the normal course of action. Failure to recognize the added cost of this effort is another source of the inaccurate estimation of project schedules.

4. You can't get future products with historical designs (**design**)

To successfully complete a new product that does not sell or has a minimal life in the market is worse than a delayed release schedule. The definition of correct requirements is one of the most

critical elements of the product development process. Successful products have a long life in the market. With the rapid change of technology today it is important that new product designs integrate not only the customers current needs but also project future needs that can be enhanced through leveraging new technology. This emphasis can lead to the development of truly innovative products that can help increase market share and foster new markets. New development effort s should always address not only the current needs of the customer but also anticipate future needs.

5. You can be more productive if the product does not have to work (**quality**)

Productivity and quality are quite different measures. Productivity measures that amount of work performed but does not readily provide insight into the quality of the products produced, i.e., we could have an extremely high productivity but poor quality resulting in significant rework and poor customer satisfaction. On the other hand we could produce extremely high quality products but at the expense of schedule resulting in extremely poor productivity. Both attributes must be measured together.

What is the optimum tradeoff between productivity and quality? Is it okay to sacrifice quality for productivity? There is no easy answer to these questions, the product characteristics, the customer expectations, and desired results are critical to determining the optimum integration of these attributes.

6. You cannot double the size of the project without changing the schedule (**scheduling**)

It seems silly to suggest that schedules must be realistic in order for a project to have a chance for success in terms of budget and schedule but for many projects this seems to be the case. The schedules and budgets are not derived from analysis of the effort required or based on comparison with like projects but based on the wishful desires. The recommendation here is obvious; ensure that schedule and budget estimates are based on analysis of effort required and experience with previous projects.

Even if the project schedule and budget are correct for the initiation of a project, they will quickly become obsolete if changes are allowed to the project without adjustment. Changes to a project must be assessed in terms of the resources and schedule consequences before accepted. Failure to enforce management of project changes may very well be the principal cause of project overruns. Changes to projects must be assessed in terms of cost and schedule impact before being accepted for implementation.

It would be nice if a project could be carefully planned and all activities would follow according to the planned schedule, however, in new development projects there are too many unknowns that preclude the ability to precisely plan schedules and budgets. It is the responsibility of the project manager to track progress and anticipate potential problems (risks). This risk management process is essential to ensure that potential problems are minimized and delays are prevented. The project manager must carefully monitor progress to the planned schedule and adjust resources and project priorities to address critical risk areas.

7. If nobody works on the job it will take longer (**resources**)

This is another statement that seems to be self-evident but is not so easily carried out in practice. Often resources that are planned for project implementation are devoted to other tasks such as correction of problems for existing projects, addressing problems that arise in other development projects, research into new project ideas, and even personal issues such as vacation and sickness. Although these are real problems that are faced by developers they are not easily communicated to customers as an excuse for a delayed release.

Seldom are project managers given the luxury of full time resources as planned for the duration of a development project. Even when the physical resources are available often times these resources do not have the experience level needed to be fully productive, a considerable amount of effort is devoted to education on the current state of the project, communication with the other developers to verify interfaces, and communication associated with understanding the project status (Mythical man month).

How to combat this problem? The simplistic answer might be to assume that the project manager has the authority to ensure that resources are devoted to his/her project alone, the realistic answer is to ensure that ant project schedule has a buffer to accommodate the unavailability of resources and other contingencies that tend to arise through the duration of the project.

8. A perfect process cannot correct the wrong product definition (**requirements management**)

Today's quality systems papers seem to focus on the concept of process improvement and associated initiatives such as six sigma programs. These programs focus on the reduction of variation and optimization of processes as the solution to quality problems. Although these techniques clearly yield benefits, they might not be the correct answer to the current problems being experienced. For example, if our product is not properly designed to meet the needs of the customer, perfecting the manufacturing process will not help.

9. Projects executed using parallel tasking can be completed faster than those executed in a sequential fashion.

When a project is near completion it is too late to consider what tasks could be executed in parallel, however, implementing parallelism early in the development process can yield insight into optimum designs and significantly reduce the overall schedule. Project schedules tend to always take longer than anticipated. To minimize the risk of schedule delays it is incumbent on the project manager to attempt to introduce as much parallelism into the development project as possible. This includes early prototype development during the definition of requirements, development and execution of test procedures during implementation, and emphasis on stress testing as the initial tests executed during validation testing

10. Preventing problems is easier than fixing them (**reviews**)

The earlier in a process that a problem is discovered the less costly it is to correct. Studies have shown that finding and fixing a problem in the definition of requirements ion the requirements review stage can be 10 to 100 times less costly to correct. This suggests that effort expended in up front review times can yield significant pay back.

Design reviews include reviews of not only final product implementation such as source code or schematics but also review of specifications produced during early design stages. Finding errors early in the development process can significantly reduce the cost of correction. Benefits of the design process include a reduction in errors but also increased communication between developers, education of less experienced personnel, and conformance to standards. Like any imitative, reviews can become excessive, the benefits of design reviews should be monitored to ensure that reviews are conducted at the appropriate level of detail and for the right subsystems.

Why Development Projects Succeed (or Fail)

Project delays

Fishbone (Ishikawa) diagram with the following branches and causes:

Problems encountered during development
- Technology (how to implement)
- Resource conflicts (people)
- Poor team communication
- Poor customer communication
- Unfamiliar with system

Requirements changes
- Improve performance
- Easier to use
- Optimized
- New (changed) interfaces
- Added more detail
- Added features
  - Market changes
- Initial requirements not understood

Requirements incomplete
- Inexperienced people
  - Inadequate training
  - Hiring practices
- Inadequate customer communication
  - Unwillingly to devote time
  - Inexperience
- Lack of system understanding
- Insufficient review time
  - Low priority
  - Poor scheduling

Poor development process
- Tools not used efficiently
- Documentation did not contribute to efficiency
- Defined process steps (reviews and test) not followed

Design not optimal
- Architecture not defined up front
- Relied on OTS components
- Poorly documented
- Inexperienced designers
- Poor scheduling of time for design

Coding errors
- Inexperienced developers
  - Training
  - Standards
- Process not followed
  - No audits
  - No measurements
- Ineffective reviews
  - Inadequate time allowed
  - Training