

# How to Complete Technology Projects On Time



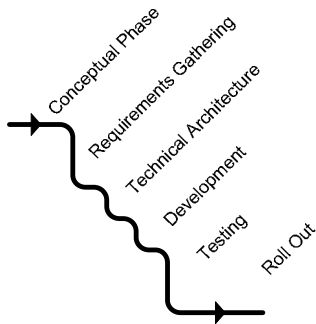
There are two general schools of thought when it comes to carrying projects from Conceptual Brainstorming to Production Roll-out.

Both of the following approaches are valid, proven, and can work:

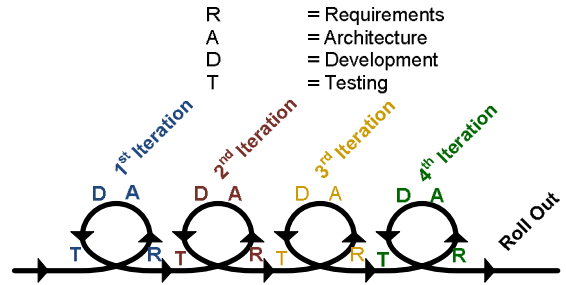
www.jasonwisdom.com

1.212.706.3071 P

1.212.706.3072 F



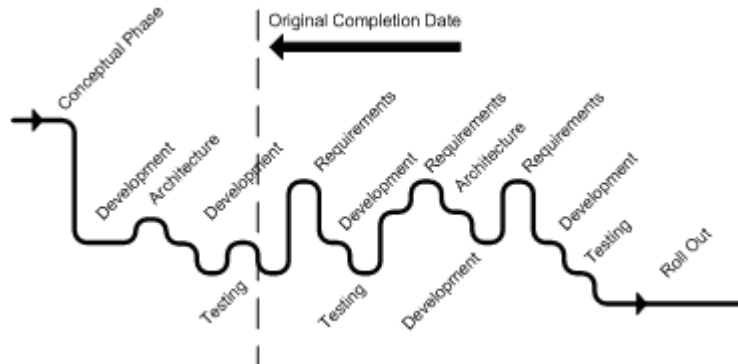
**Waterfall Approach**



**Agile Approach**

Unfortunately, what happens with many projects is that efforts go from Conceptual Phase straight to Development, pushing an aggressive timeline because “It has to get done”, and expecting the project to go straight to Roll Out on its own.

Instead, what happens is more like the following . . .



. . . and what was a 3-month project ends up dragging out over a year. Pictured above is an example of “Cowboy Coding”, and it is **not** Agile.

There are three main reasons Technology Projects run past their expected delivery date:

- 1) **Weak Requirements** – There seems to be a strange, widespread attraction to the “We’ll figure it out as we go” approach. While this can be very noble and is the storyline behind many movies, when 3 or more departments each issue change requests on a weekly basis, the project is going to be delayed, cancelled, over budget, and/or function with a fraction of the originally-intended functionality.
- 2) **Inadequate Testing** – Testing and QA require more time than 10 days tacked on to the end of a 6-month project. If a major bug is found, it can take *weeks* to correct. When adequate time is not budgeted, the only alternative to delaying the roll-out is demanding 7-day overtime work schedules, which is when *real* disasters occur.
- 3) **Technical Incompetence** – The technical lead, business analyst, project manager or at least one senior programmer is a total dolt.

Fortunately, each of the above reasons has a simple resolution:

- 1) **Fixed, finalized Requirements** *must* be prioritized, *before development begins*. I have seen far too many technical leads waving their arms in frustration, “When are we going to stop getting scope changes?!?!” and hearing only the howling wind in reply. What about change that happens “at the speed of business”? There is a difference between changes caused by market conditions, versus changes caused by business stakeholders dragging their feet. The latter *must* be clamped down on, by top executive ranks on down.
- 2) **Testing and QA** should be given one-third of the project timeline. If a Waterfall project takes 6 months, the finished Beta test should be finished within 4 months. If each Agile iteration lasts 3 weeks, each 3<sup>rd</sup> week should be frozen to new development.
- 3) This one is easy. **Replace the dolt**. You simply cannot afford to keep perpetually bad staff on board – competition will eat you alive.