

# Don't let project success slip through your fingers

Applies to: [Microsoft Office Project 2003](#)

By [Lonnie Pacelli](#)

Few things are more frustrating to a project manager than having a well-executed project fail because the project team tripped at the finish line. Whether it's because of taking on too much work, releasing resources too early, or coasting when victory seems imminent, having a successful project slip through your fingers at the last minute can feel like a knife cutting right through you. This article helps you, the project manager, spot key warning signs that indicate risk and helps you put practical strategies in place to ensure that you finish your projects successfully.

## Case study: How last-minute failure occurs

Andy Teal was a project manager tasked with implementing a new-order management system for Wingtip Toys. The project had been going well, with few issues along the way. With only five days left before going live with the new system, the team was feeling confident that the project would go into production without a hitch. During the weekly team meeting, a user asked Andy whether the team could implement a key enhancement that was originally requested but was later deemed out of scope. Andy assessed the remaining work and felt that there was enough team capacity to implement the enhancement before going live in five days. He agreed to the enhancement and released it to the development team.

Two days before go-live, Andy asked his lead developer, Sunil, how development on the enhancement was going. Sunil answered, "We're working through some bugs, but we think we'll be able to get it done by go-live." Andy was nervous, but told Sunil to keep going and to get it wrapped up. The day before go-live, the test manager, Jay, told Andy and Sunil that he was not going to accept the enhancement. "This thing is just too buggy!" Jay exclaimed. "If you put this into production now, we're going to have a lot of angry users!" Andy was now faced with the decision to keep going or to stop and roll back the changes. Andy decided to forge ahead.

Team members worked through the night to get the enhancement working and felt they got things to a point where the system could go into production. However, on go-live day, the system went into production and failed, when users attempted to perform a function that the test team had neglected to test. The production

implementation was stopped, and the users went back to using the old system while the development team fixed the problems. What had seemed like a successful project just five days earlier was now a failure.

One of the hardest failures to accept on a project is when you execute well through the life of the project, only to stumble at the end because you claimed victory too soon. I've learned (through failure) that in the last few weeks of the project, you must ensure very tight execution, daily (or more often) communication of activity status, and rapid resolution of any issues that may crop up.

## Why it happens

**Someone implements a last-minute product change that breaks something else** If you are a Monty Python fan, you may recall the comedy routine where a man gorges himself on food; but when he is offered a mint, he literally explodes. This is a great analogy for what can happen on a project when last-minute changes are put into the nearly finished product.

On a recent project, where I was the business owner, I mandated to the project team during the final weeks that they change the product only when absolutely necessary. However, one of the customers convinced a developer to make an "easy" product change. The developer made the change and broke a key feature in the product that was previously working. Not only was the "easy" change not implemented in the final product, but precious resources also were consumed during the attempt to implement the change and then diagnose what went wrong.

**Project communication among team members isn't timely** As I mentioned previously, I find it necessary to focus on regular team communication as I move into the final weeks of a project. I'm a big fan of holding daily morning meetings, which highlight the activities of the previous day, outstanding issues, and action plans for the current day. It's also critical to make sure that all project roles are represented at the daily meetings so that team members can assess activities from their own vantage point and understand how they may be affected. I also believe in keeping the daily meeting short — typically no more than 30 minutes.

**The project starts shutting down prior to completion** As a consultant, if I were experiencing budget pressures on a project, I would constantly assess who I could roll off the project early to save money. This can be good cost management if the person's tasks on the project are finished. However, cases in which the tasks aren't completely finished and where the rest of the team is expected to pull

up the slack, you've now created a personnel resourcing risk that you didn't have before. In reality, you're most likely not saving much money (if any) if someone still needs to complete the tasks.

## Warning signs

**Team members are reassigned to other projects** As a project nears completion, other project managers will likely pressure you to reassign your project team members to their projects. Reassigning some people may be in order, but you should ensure that people aren't leaving too early. I also try to keep some reassigned project team members tethered to my project, in case they are needed to help with a critical task.

**Customers demand last-minute nonessential product changes** Whenever you hear "but this is so simple, it should take only a few minutes to complete," warning bells should go off in your head. True, a feature change may be simple on the surface, but it could be more complex when you consider all the factors that ensure the product feature is implemented correctly.

**Project team communication has dropped off or is nonexistent** When the project team isn't in frequent communication toward the end of the project, team missteps likely occur because team members don't understand last-minute issues or decisions. I prefer to err on the side of overcommunicating during this time, just so that the team is fully aware of any developments.

## Turning it around

**Keep the focus** Keep the project team together, focused, and current on what is happening in real time. Make sure that the team knows which things need to happen each day and who is responsible for driving key activities, issues, and decisions. Be cautious of project team members being pulled onto other projects before your project is completed.

**Drive the team to communicate** My preference is to have shorter, more frequent status meetings — at either the beginning or the end of each day — to keep everyone current. On some projects, I've set up a "war room," where we kept current tasks, issues, and decisions on whiteboards. We've even had key project team members camp in the war room so that if impromptu discussions were needed, the right people were assembled to facilitate discussions.

**Stabilize the product** As you get closer to product release, be very careful about implementing product changes that aren't absolutely essential to the product's functionality. Anytime

you introduce a change, you stand the chance of unintentionally disturbing something that you don't mean to disturb. Don't feel obligated to fill each minute of the project by adding more product features.

## Finishing strong

It's frustrating to a project manager when a potentially successful project turns into a failure at the last minute. The old phrase "Don't count your chickens before they hatch" couldn't be more true here. Don't do anything unnecessary at the last minute that could put your project at risk. It's better to finish strong with fewer features than to trip before the finish line trying to squeeze in that one last piece of work.

**About the author** [Lonnie Pacelli](#) is a business owner, consultant, and author with more than 20 years of experience as an executive, project manager, developer, tester, analyst, trainer, and business owner. He is author of *The Project Management Advisor: 18 Major Project Screw-ups, and How to Cut Them Off at the Pass* (Prentice Hall, 2004).