

By Rick Vanover

IT projects can be daunting, especially to the novice. It's relatively easy to propose a solution, but much more difficult to implement the desired performance levels on time for the right price. This list will help IT pros bring organization, professionalism, and goal-oriented progress to the projects they manage.

## Table of contents

Get professional .....	2
Identify the leadership roles .....	2
Focus on scope management.....	2
<i>Real-world example</i> .....	3
Create the project definition or charter .....	3
<i>Real-world example</i> .....	3
Identify the risks .....	4
Manage relationships with external parties.....	4
Maintain strong documentation standards .....	4
Build effective communication channels.....	5
Keep an eye on costs .....	5
Don't forget the closeout .....	6
<i>Additional resources</i> .....	7

## 1 Get professional

IT projects historically have a negative reputation for being over budget, late, and poorly implemented. Having a professional individual in charge of the project can add great organization and credibility to your efforts. If your project is of a size where a project manager role can be used, go for it.

Working with a [Project Management Institute](#) (PMP)-certified individual will greatly enhance the effectiveness of your software projects. The PMP is also a good benchmark across all project management disciplines and is a big credibility booster when a project integrates with non-IT individuals, external customers, business partners, or part of a larger project.

## 2 Identify the leadership roles

Having individuals responsible for specific metrics of the project is important. This should be done in a way that puts capable individuals in roles that are best suited for their talents but that doesn't overwhelm individual team members. IT projects often put too much emphasis on the technical contributions of a small number of individuals—or even just one person—and effectiveness is limited when these resources are maximized during the project cycle.

You should also ensure that individuals in charge of specific areas of the project do not hoard responsibility. For example, a person or small group may make great contributions to the progress of the project in regard to overall systems performance, not using so much time for the project (when working from a fixed-price/hours amount project), and getting finished ahead of schedule. But these efficiencies may come at the price of this individual or group not updating project documentation or ensuring revision control with authoritative instances of documents or code and possibly missing “the little things” in the project.

Individuals with leadership roles within the project can ensure that the project follow-through is done according to the required standards. Examples of this include roles such as Technical Lead, Project Lead, or Documentation Lead. These leadership roles can provide checks and balances in the event that a person becomes reassigned unexpectedly or leaves the organization. The continuity chain can be made stronger by tighter integration across individuals for progress points and ensuring the administrative follow-through of the project.

## 3 Focus on scope management

Scope management is one of the most important aspects of IT projects, and it's the team's responsibility to make sure that any scope changes are introduced in the correct forum. The project process should include procedures for making a scope change proposal.

It's also important to ensure that the official mechanism for project documentation maintains robust revision control, because scope can change functionality requirements and thus change the documentation that accompanies a project. In the event that a scope change is backed out, proper revision control will ensure that the original functional levels are available from a documentation standpoint.

## Real-world example

We solicited feedback from Bill Reits, a certified PMP at Siemens Logistics and Assembly Systems for some comments on scope management. He said that one of the most common and troublesome scope problems within IT projects is Gold Plating.

**Gold Plating** is adding undefined features to a project that were not within the agreed scope of the project. It's common in the software industry because programmers, software engineers, and IT pros decide on their own to add "cool features" that they determined would be fun to code, tools, or other benefits to the implementation project or customer's deliverable system. Although the intentions are often well meaning, Gold Plating can have the following costly consequences:

- The individual can underestimate the effort, get caught up in developing or showcasing unnecessary features, and end up taking a great deal of time that was not budgeted at the expense of deliverable requirements.
- Because the task was not planned, it often affects other areas of the project that were not considered. This can be negative performance impacts, unclear training materials that differ from practice, or other methods.
- If the tasks introduce a nonconformance (a.k.a. software bug), a great deal of warranty effort can be expended correcting something that was never within the scope of the project.
- When an individual adds a "feature" that was not in the scope of the project, additional work from other team members can be required. For example, the feature must be added to the master documentation, the functional specification, the operators manual, the unit test plans, the integration test plans, the acceptance test plans, the traceability matrix, etc. It should be obvious that one small easy-to-code feature can add many hours to a project.
- It may be possible, that the added feature is not desired by the customer, resulting in time and effort to remove it and in customer dissatisfaction. For instance, a "slick feature" may be added to a banking application that is against government regulations or bank policy.

## 4 Create the project definition or charter

Having the project clearly defined can pave the way for all subsequent aspects of the project to be implemented correctly. A well-defined project definition and corresponding processes gives the project a strong foundation.

The project definition will define an agreed-upon performance baseline, costs, efforts required, expected functionality, implementation requirements, and customer requirements, and it identifies the individuals and organizations involved in the project. Project definitions that include specific technology details on how a task is to be accomplished will benefit all stakeholders of the project.

## Real-world example

One TechRepublic member was implementing a project whose initial project definition referenced communication between two systems as the following:

"The host system automatically will send the order system the order information over the network using a standard interface."

This language spells trouble, since it could mean so many things: An EDI transaction, an FTP exchange between the two systems, two custom socket interfaces exchanging a messaging formats, an XML file, connectivity through a standard product like MQ series, SQL database replication, or any other number of ways of two systems exchanging data.

## 5 Identify the risks

IT projects can incur risk in unique ways, as IT projects make frequent use of vendors, consultants, and contractors. For example, if your organization contracts Acme IT Services to assist your IT staff in its upcoming Active Directory and Windows 2000 Professional to Windows XP Professional client migration, you may face the risk that Acme IT Services could go out of business, get a "more important" client, or do an inferior job.

Each element of risk—resources, schedule, performance, cost, etc.—should have assessments performed. These tasks are usually delegated to the project manager or individual most closely associated with that role. Periodic risk assessments and tracking are due diligence of the project process. Risks manifesting themselves in the project cycle should have recourses as well. For example, if Acme IT Services leaves your project for another client, ensure that there are recourses to working with this agency.

## 6 Manage relationships with external parties

IT projects will almost always have some level of involvement with external parties. These parties can be:

- Consultants
- Business partners
- Service providers
- Vendors
- Software publishers
- Equipment manufacturers

Having external parties involved in the project will add resources and ability to the appropriate deliverable of the project. However, ensure that each organization's role and need is clear. The project plan should identify an individual to be in charge of administering the relationship and availability of external parties. If your organization executes many projects at once, this individual may perform this function for all active projects.

## 7 Maintain strong documentation standards

Documentation is the key to a successful IT project, especially when changes need to be made after implementation. Ensure that your organization has clearly defined documentation expectations as well as standardized repositories for various types of documentation. Revision control mechanisms are also important if custom development is being performed.

In addition, it makes sense to have documentation that defines the documentation requirements. That may seem like overkill, but as a project scales in complexity, this becomes more valuable to the success of the project implementation and manageability.

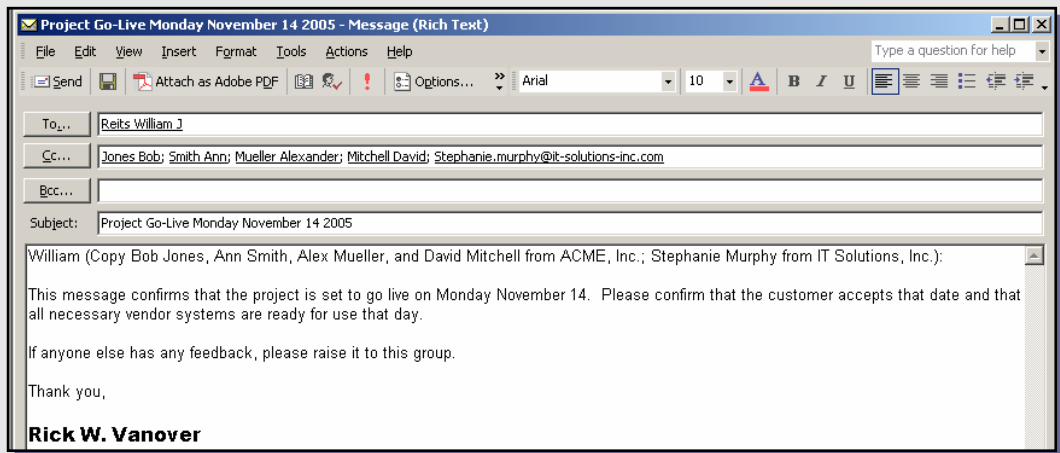
Strong documentation standards offer the following benefits to IT projects:

- New team members can assimilate more easily.
- Future work related to this effort are more easily started.
- Functionality changes are easier to stage or test.

## 8 Build effective communication channels

Project management should coordinate clear communications. E-mail seems to be the preferred mechanism for this, but it can easily become overwhelming and inefficient. One popular good practice is to identify specific individual(s) when a response is required. By using the TO: and CC: fields appropriately, you can avoid unclear messages about who needs to do what. The figure below shows a good example of an e-mail communication that outlines specific responsibilities.

This e-mail message clearly identifies that its target is William. If there are any issues with the topics presented, it is the primary responsibility of William to raise them. The other members are presented with an opportunity to raise concerns and to share them with the selected distribution.



Little habits can add great effectiveness to the communication patterns, especially when involving external parties. For instance, in the example above, members from each organization are grouped to give clarity to distribution. How many e-mail messages have you received where you aren't even sure whether you're being addressed, much less whom you should reply to?

Also make it a priority to communicate the schedule (and its changes), status reports, scope topics, and new issues that arise in the project process. Clear, concise, and targeted communications are all positive habits for IT projects.

## 9 Keep an eye on costs

The closer you are to the technology, the less pleasant the topic of cost becomes. Nevertheless, cost is among the most important aspects of the project process. Each project member should be aware of the costs associated with his or her aspects of the project. This also becomes important if it's determined that the scope of a project should be changed. For example, consider the following technology scenarios:

- A new version of a critical software component is released.
- A security risk for a software component is discovered.
- Newer or faster computer equipment is required or desired.

Scope change can address these topics, but there may be dependency scope changes that go with them, which can greatly increment the costs involved. Licensing, space concerns, "lost licensing" or unused equipment and software, and rework or lost time all can add to the cost of scope change.

Fear of the price impact should not deter scope change, but it's an element the project team must keep in mind.

## 10 Don't forget the closeout

Once the deliverables of the project have been met and all appropriate signoffs have been obtained, exert the same effort to correctly close the project. Depending on your project type and scope, the project's closeout and post-mortem are important to ensure that all project members have executed their required steps and that the customer (internal or external) is satisfied with the project results.

Depending on the scope and nature of the IT project, the closeout may be a required step to take the project (or customer) to "support mode." Project turnovers, closeouts, and other mechanisms to prepare the project for ongoing life are important to ensure that all the ends are in place so that when this topic arises again, there is a good reference point on the details of the project.



*Rick Vanover works for Siemens Logistics and Assembly Systems in Grand Rapids, MI, where he's part of a team of IT professionals that deploy custom software solutions to the material handling industry using a mix of current hardware and software products. Previously, he lived in Columbus, OH, and focused on client-server administration and support, project management, negotiating vendor agreements, and Internet connectivity and VPN. You can reach Rick at [b4real@usa.net](mailto:b4real@usa.net).*

Thanks to Bill Reits for his comments and other guidance in preparing this material.