Credibility and Commitment

Contents

Bill, a software manager, had only been in the job for about a month. He was now the manager of a very large software project and his 12 top managers were all in the room. This programming group had not made a schedule in recorded history and now the company really needed their product fast. Bill had successfully managed several previous projects and he had been selected to straighten this one out.

Vin, the company senior vice president, had called a meeting with Bill and his top software managers. Vin spent the first 20 minutes chewing them all out. The company was ready to ship hardware and the software was not ready. They had promised to deliver months ago and had missed every single date. Now, nobody believed them. Vin demanded a delivery schedule in two weeks.

When Vin stopped talking, everyone looked at Bill. This was the time to put up or shut up. Bill knew there was no way to produce a project plan in two weeks. Several hundred people were working on this project and it would take a week just to tell everybody what to do. Bill knew that a schedule without a plan would be a guess and the last thing he wanted to do now was to guess. The new dates would then be no better than all the others and it would only be a matter of time before Bill was on the scrap heap with all the other failed software managers. Bill didn't know how Vin would react but he did know that if he didn't take a stand right now, he would never have a better chance. He figured that since he had just been put in this job, they wouldn't fire him this soon.

Bill told Vin, "We could make a guess, and give it to you today. But if you want a date we will actually meet, we must make a plan, and that will take 60 days."

Vin was surprised. He looked around the room and asked the managers what they thought. Each manager, when asked, said, "Yup, it would take 60 days to make a plan."

In the end, Vin agreed and gave the project the 60 days they needed to produce a plan and a schedule.

After the meeting, Bill was hailed as a hero. For the first time, their boss had stood up to management pressure and had gotten the time to make a real plan. Bill never knew whether Vin

had staged the meeting to test him, but it didn't matter, it had a marvelous effect. The managers were now committed to producing a plan in 60 days, and they intended to make a plan they would actually meet.

They did in fact finish the plan in 60 days. With the time to thoughtfully make a plan, the managers compared their new work with what they had done before. By using historical data, they were much more realistic. This made the schedules longer than anyone wanted, and many of the key functions were spread over several releases. What surprised everyone was that with realistic schedules, they could now think about their work instead of reacting. They found they were meeting their interim dates and they could count on their coworkers to meet theirs. As a result, they delivered the first release ahead of schedule and didn't miss a single date for the next two and a half years.

The difference between a guess and a commitment

There is a fundamental difference between a guess and a commitment. Even when they both produce the same date, the guess has no substance. With a commitment, the engineers have thoughtfully considered the job and how to do it. They have compared this new job with data on their previous work and have committed themselves to the plan. They have thoughtfully made the plan and fully intend to deliver the product when they say they will. Its a matter of personal integrity.

With a guess, nobody is on the hook. The manager has pulled the date out of a hat. While it may be a plausible date, and the manager may be publicly committed to it, none of the engineers have thought about the job. They have not planned how to do the work and their integrity is not on the line. Most important, only the manager owns the date. The engineers' attitude is best summarized in the words of a disgruntled engineer who once said "It's his date, I wish him luck."

In planning, the best rule is: if you don't work hard to produce a delivery date, you won't work hard to meet it.

Believing in magic

To see what planning accomplishes, consider the following hypothetical case. You have a large program to develop, you are the manager, and you have been assigned a development team to do the work. Management, however, requires that you first produce a plan and schedule for the work.

In making this plan, assume you have a magic planning tool. It will tell you the time the development will take and the size of the finished product. All you need to do is to enter the requirements data, and voila, out comes the size of the ultimate product and the time, in

programmer months, to do the work. There is, however, one limitation. This tool will only tell you the best possible size and development time. It assumes that you have the world's championship development team, and that they will produce the absolute best system. It also assumes that nothing goes wrong, that you get all the resources you need when you need them, and that your people will build just the right system the first time.

Even if you could get such a magic tool, you would still have to figure out how this ideal product differs from the one your team will produce. You must then estimate how much longer your team will take to do this work. Also, you need to consider what could go wrong, how much time each mishap will take, and what contingencies are needed for requirements changes, staffing delays, testing, rework, vacations, sickness, meetings, etc.

This means that a magic planning tool would not really help that much. While it could tell if the requested date was better than the world record, it would not tell when you could realistically expect to finish the job. This suggests that the search for magic planning tools, models, or methods is a waste of time. Even when they operate properly, they will only give theoretical figures that do not represent the reality of this team for this project.

So what is left?

What is left is the need to make a plan. This requires a detailed study of this project. You can then judge the likely size of the product and compare it with prior projects. With an estimate of the work to be done, you can lay out the tasks, judge how much time each is likely to take, and set the schedule for the work. Once this plan is complete, you can negotiate a realistic commitment.

Just what is a commitment?

A commitment is an agreement to accomplish something. Say, for example, Mr. A agrees to develop a software product for Ms. B by a specific date. That is a commitment. The quality of the commitment is determined by the degree to which Mr. A responsibly sets the date and the degree to which Ms. B realistically represents what she wants.

For a commitment to be meaningful, the person making the commitment must intend to perform as committed. That is, when Mr. A agrees to complete the development on a date, he must really intend to do so. To responsibly do this, he must think through the job, estimate the work involved, and then set the completion date.

The elements of a commitment are then:

- one person who wants a task performed
- another person who agrees to perform the task
- a price and a date they both agree to for the job

While there are often contractual and technical specifications for the work, the essence of a commitment is an agreement on a task specification, a schedule, and a price.

The commitment negotiation

In the simplest case, a commitment is an agreement between two parties to accomplish something important to them both. Neither one wants an unrealistic date. In our example, Ms. B wants the shortest date she can get A to agree to. While she would probably take the earliest date she could get, her real interest is in having A work as hard as he can to complete the specified job as quickly as possible.

Mr. A, on the other hand, would also like to complete the job as quickly as possible. He, however, would like some room in the schedule in case something goes wrong. He is not sure how big the job really is, and would like a cushion in case it is bigger than expected. He thus wants a realistic schedule but one that is shaded on the high side.

With these opposing objectives, it is clear that commitments must be negotiated. Mr. A wants the longest date he can reasonably get, and Ms. B wants the shortest. Using whatever data, arguments, or authority they can each muster, they must somehow establish a mutually agreeable date. The point is that when making a commitment, neither side knows how long the job will take. While the manager has authority and influence and the engineer has job knowledge, they each need the other's agreement.

These negotiations might seem to be evenly matched, but the manager has an important advantage. Assuming Mr. A has no clear idea of how long the job will take, he must depend on his intuition and his powers of persuasion. If he is dealing with his manager, he is at a serious disadvantage. He knows that Ms. B's opinion of his work is critical to his future. Whether realistic or not, he always has the worry that these negotiations could affect his performance rating, promotion prospects, or salary.

To help in this commitment debate, Mr. A needs some tools. The only effective tools, however, are planning skills, data on the job size, the cost of prior similar jobs, and the plan for doing this job. Thus Mr. A's bargaining power depends almost entirely on data he can muster on this job and other similar work.

Ms. B generally has authority as Mr. A's manager. She has leverage over his salary and could select someone else to do the work. Unless he has other projects to work on, A could even lose his job. Ms. B thus has tremendous negotiating power over Mr. A. Even in negotiating with customers, engineers are relatively powerless. The customer could select other vendors, insist on painful contractual terms, or cause other unpleasantness. While these actions may not directly impact the engineers, they will impact their managers who will then put pressure on the engineers. Without the leverage of a plan that is based on credible data, engineers have almost no power in these schedule debates.

The advantage of credibility

There is, however, one exception to the engineer's lack of power. That is, if Mr. A has credibility. Bill, for example, had previously completed several projects on schedule and Vin trusted him and believed what he said. Suppose, for example, that Mr. A has already completed several projects on or ahead of schedule. Now, if Mr. A says that a job will take four months, Ms. B probably would not argue. She might ask for supporting data but she has learned that Mr. A does what he says he will do, and he does it on his committed schedule.

Credibility is powerful. Once engineers have it, they can debate schedules from a position of strength. When they give a date, everybody believes them. In fact, the engineer with credibility is like a magic planning tool, only one that is specially tailored to the work to be done. The managers just ask the credible engineer how much time and resource the job will require, and voila, they get a date they can count on. While it may take a little time for the engineer to produce a plan, when they get it, they know they have something worth waiting for.

Earning credibility

In our business in particular, no one believes software schedules. Even the leading software firms routinely slip their dates several times before they make a delivery. As software engineers, we start at a disadvantage. No one believes us. We don't have credibility. To get credibility, we must earn it. The only way we can earn credibility is to start consistently meeting commitments.

While your credibility will withstand an occasional misfortune, it is tenuous. It is like a bank account. If you make enough deposits, you can make an occasional withdrawal. Unfortunately, when you make a withdrawal, you take the balance to zero. To keep your credibility, you must continue to deliver your work on schedule, and if you miss once, you must start to earn your credibility all over again.

So, to earn credibility, you must plan your work, and you must make plans that are realistically based on your historical performance. Then you only make commitments when you have a plan. When you do this, you will make responsible commitments, and you will more consistently meet them. Only then will you be respected as a competent software practitioner. You will get more satisfaction and pleasure from your work, and you will have credibility.

An invitation to readers

In these columns, I discuss software process issues and the impact of processes on engineers. I am, however, most interested in addressing the issues you feel are important. So please drop me a note with your comments and suggestions. Depending on the mail volume, I may not be able to answer you directly, but I will read your notes and consider them when I plan future columns.

Thanks for your attention and please stay tuned in.

Watts S. Humphrey
watts@sei.cmu.edu