

IT METRICS STRATEGIES

Helping Management Measure Software and Processes and their Business Value



Case Study: Benchmarking, Estimation, and Applications Outsourcing Gone Awry

by Michael Mah

When a company or department becomes single-mindedly obsessed with time to market, it sometimes acts in risky, self-destructive ways that result in the thing it fears most: blowing the delivery date even more than originally anticipated.

This article presents a real-life case study about outsourcing gone awry (names, places, and dates have been changed to protect the innocent — and the guilty). But it's not just about failed strategies, because locked within the story are "DNA blueprints" that reveal what might have been done differently. From the failure, we can learn and potentially mine some future successes.

The setting is a medical applications company. The actors are a newly hired CIO/VP of development, a director of IT who reports to him, the VP of marketing, the CEO, an archrival competitor, an outsourcer (waiting in the wings), internal IT development, and a metrics consultant.

Plot Summary

The newly hired CIO/VP of Bradley Davis Medical Systems (BDMS) was a man on a mission. He was brought in to take on the company's chief competitor, gain market share, and dramatically ratchet up revenue at

Continued on page 2.

executive summary

This issue of *ITMS* addresses the use of IT metrics in outsourcing and strategic partnering on "projects for hire," in which a client organization needs a system (fast — what else is there?) and an outsourcer or partner offers to build it.

IT applications outsourcing of this sort is becoming increasingly popular as demand for new functionality continues to outpace IT capacity, and companies rush to expand into new markets.

Process productivity limitations, combined with IT staff shortages, are forcing companies to increasingly hire outsiders to build applications for them. The problem is, how do you know if a contractor can deliver what it promises during the courtship (aka proposal) phase?

First, we have a case study that looks at a real company's outsourcing dilemma: a new CIO/VP of development; an inhouse development group; deadline pressure from a tough competitor in the marketplace; a marketing department that demanded a lot of function in a short time to increase market share; a project estimate by internal development staff that senior management felt was "too long"; an outsourcer waiting in the wings. What did the company choose?

We view these scenarios through the measurement lens and offer some ideas on what to do and what *not* to do. The moral of the story is this: incredibly diverse outcomes exist — some good, some really bad — and these are ultimately determined by the management choices that are made.

Returning to *ITMS* are Ware Myers and Larry Putnam with "Evaluating Bids: First the Facts, Then the Acts." Putnam and Myers describe a request for proposal that was met with two bids. One was US \$86 million; the other was \$147 million. What happened? What might have been done differently after the fact? Read their excellent treatment of this subject and find out.

Michael C. Mah, Editor

october 2000 vol. VI, no. 10

Case Study: Benchmarking, Estimation, and Applications Outsourcing Gone Awry	1
Evaluating Bids: First the Facts, Then the Acts	1



Evaluating Bids: First the Facts, Then the Acts

by Lawrence H. Putnam and Ware Myers

What happens when two organizations go bump in the night? Well, if they are boats, they spring leaks, take on water, miss arrival times, and sometimes sink. If they are a client organization needing software and a software organization providing software, they run into trouble, too. The sad experience in 1995 of the Los Angeles County Board of Supervisors and a software contractor reveals what happens when ignorance — actually, partial knowledge — bumps into partial knowledge.

What happened to the Los Angeles County Welfare Department back then is that the low bid for an

Continued on page 11.

Continued from page 1.

BDMS, all within one year. He had just come into the job two months earlier with a nice pay package, stock options, and an executive bonus tied to the price of the company's class B shares. All he had to do was deliver a new product (SmartMed 5000), cut overhead, raise profit margins, and successfully launch the product in tandem with an aggressive marketing campaign, led by the VP of marketing. Simple enough.

Alas, it wasn't that easy. Within weeks of the CIO/VP's arrival on the job, an archrival company sprang an ugly surprise on BDMS. In May, the rival preannounced the availability of a product that trumped the planned features of SmartMed 5000. It claimed it would reveal the new capabilities with great fanfare at an annual trade show in Geneva, Switzerland, the following January. This move was designed to embarrass BDMS and would surely steal its market share. This was bad — real bad.

Marketing at BDMS then decided on a pre-announcement of its own. It increased the promised features for the SmartMed 5000 product. BDMS presold it at a discount to major existing clients and made heady promises to potential new customers. All the internal IT group had to do was deliver. The problem was that the director of IT said the project was risky, especially with the expanded scope and the shortened deadline. The company might not make the date.

But the CIO/VP was determined to get his date (January's trade show) and the functionality no matter what. IT development made its best case. The date was risky; the only way to increase the odds was to scale back the features. The CIO/VP would have none of that. "Give me my date and all my features, or I'll find someone who will," he said.

Under that threat, IT had little choice but to cave in. Staff members started the project

and worked hard. But they were battling a set of shifting requirements from an indecisive and scared marketing department. Soon, the deadlines began to slip. The CIO/VP started to get angry. On the side, he initiated secret negotiations with an outside company — BuckStar International. BuckStar promised that it could deliver all the functionality at a fixed price. The CIO/VP quickly signed a deal, terminated the inhouse effort, fired most of the team to cut costs, retained a small staff to oversee the vendor, and, hence, outsourcing began. Marketing went full-steam ahead with its promotional campaign.

This strategy failed. BuckStar couldn't deliver by January. But the CIO/VP wasn't told until December, when it was too late to do anything else. BuckStar, embarrassed, said it could show a prototype at best, with one-third less functionality. The CIO/VP was furious. The contract was canceled, losses were written off, and BDMS filed a lawsuit against BuckStar for breach of contract.

Desperate, the director of IT tried to hire back the old team members, but they had scattered to competing companies. She managed to get back eight members of the original team, but it wasn't enough. The day of the trade show came, and BDMS had no SmartMed 5000 to show (then, or ever). The CEO stepped in, and the CIO/VP was abruptly fired.

Ironically, in the end, the competitor that started this chain of events with its pre-announcement didn't make the date either. When the Geneva trade show came, it had to resort to showing a simplified prototype. The company missed its planned ship date by more than six months.

All of this occurred because the CIO/VP refused to hear that the internal team's best estimate for a system with all the functionality

Editorial Office: Clocktower Business Park, 75 South Church Street, Suite 600, Pittsfield, MA 01201. Tel: +1 413 499 0988; Fax: +1 413 447 7322; E-mail: michaelm@qsm.com.

Circulation Office: IT Metrics Strategies® is published 12 times a year by Cutter Information Corp., 37 Broadway, Suite 1, Arlington, MA 02474-5552, USA. For information, contact Megan Niels, Tel: +1 781 641 5118 or, within North America, +1 800 964 5118, Fax: +1 781 648 1950 or, within North America, +1 800 888 1816, E-mail: info@cutter.com, Web site: www.cutter.com/consortium/.

Editor: Michael Mah. Publisher: Karen Fine Coburn. Group Publisher: Anne Mullaney, Tel: +1 781 641 5101, E-mail: amullaney@cutter.com. Production Editor: Lori Goldstein, +1 781 641 5112. Subscriptions: \$485 per year; \$545 outside North America. ©2000 by Cutter Information Corp. ISSN 1080-8647. All rights reserved. Unauthorized reproduction in any form, including photocopying, faxing, and image scanning, is against the law. Reprints, bulk purchases, past issues, and multiple subscription and site license rates are available on request.

being asked for was seven weeks later than what he wanted to hear.

The IT Director's Story

Chris Ayari, the director of IT, had a bad feeling from the moment she found out that she had a new boss, CIO/VP Fred Alistair. Fred had a reputation as a slash-and-burn executive. For months, senior management had been hinting at dissatisfaction with the IT group. Management's complaints were centered on costs being too high and products often coming in later than promised.

Chris knew her team was incredibly talented — the best she'd worked with — and that team members had made major sacrifices over the last year to meet what they considered insane schedules. But this knowledge was largely intuitive; she had no metrics to prove it. Whenever she tried to make a subjective case to prove the levels of her team performance, she would get slammed with the history of missed dates. Although the missed dates were true, the real problem was that the projects were allowed to grow in scope with no change in already unrealistic deadlines.

Compounding things were the constant references from senior management about the high productivity of offshore developers being described at the time in airline magazines. She felt ill every time these glib comparisons were made.

One thing she knew for certain: the team worked hard on the last two product releases and managed to create success against serious odds. Toward the end of the last project, her team worked massive amounts of overtime, coding and testing throughout the Thanksgiving and Christmas/Chanukah holidays. Two senior staff members even canceled vacations, telling their children they'd make it up to them. Although the team was nervous about it, Chris was sure that if she did a metrics analysis, it would show high productivity and help them better estimate the SmartMed project.

Although she knew this should have been done sooner, she decided to benchmark the last two releases and present the findings to management. In the back of her mind, though, she still was suspicious of what Fred was up to, regardless of the numbers.

The Product History

Chris rounded up her team members and told them her idea about getting the numbers on the 8240 and 960 projects. For starters, she wanted to get all the schedule information for each of the major phases: the feasibility study, the specification and preliminary design, the main build, and the staffing history of each.

The team leader said that while this information existed for the 8240 project, the 960 was a reengineering of an existing product, so it skipped the specification and preliminary design phases. Chris replied that the data on the build phase was still useful. Another team member piped in, saying that in looking back, skipping the design phase was a mistake. The 960 project turned out to be more complex than 8240, as many of the early assumptions were incorrect. This made the build phase ugly. Many team members were afraid that the numbers would not look good in management's view, and they hoped Chris would bury the information.

Chris replied that she would want to see the numbers anyway and not prejudice what the outcome might look like. Even though the team members were nervous, they agreed to get the facts.

Schedule and staffing were just two of the needed metrics. It was agreed that they'd also quantify the amount of functionality they had built. They decided to inventory the number of C++ objects that were built new, plus the objects that were modified. This was easy. Moreover, the code sizes for all of the objects and classes came out of a standard counter within the configuration management system.

When a Really Bad Bug Happens, Someone Dies

The last metric was trickier: defects. Test logs existed for projects but were kept in a somewhat haphazard manner. However, on the 8240 project, team members were more rigorous about defect tracking because the project required US Food and Drug Administration (FDA) certification and had to have extremely high reliability. The major reliability risks were database corruptions. If bugs in the code resulted in these types of events, they were considered showstoppers.

In this case, database corruption resulted in wrong test results for a sick patient, and sometimes this was not immediately obvious. If that happened, a determination of a certain type of systemic bacterial infection could be wrong. If medical actions weren't taken immediately by the physician, a patient could die within 24 hours.

Aside from the human tragedy of such an event, it would certainly be bad publicity for the company. Marketing wouldn't like that scenario, Chris thought. She wondered if the company could be convinced of the importance of software quality with that as an example.

During the process, the team found itself aching to know what the metrics would show about development speed. It got the data in two days and plotted the benchmark results (see Figure 1).

What the Benchmark Meant to the Teams

The results were fascinating to the developers — it confirmed their intuitive beliefs about both projects. The 960 project had turned out to be more complex than what anyone expected at the time. When they viewed the chart, it made sense to them that

the project had plotted slightly longer than the average (center line) against an industry benchmark trend. It was a really hard project, and it needed more time.

The numbers on the 8240 project also made complete sense to them. Although it was technically less complex, it had a very tight deadline. In the last quarter of that fiscal year, everyone worked massive amounts of overtime during the critical test phase to get the system deployed. Eighteen-hour days were not uncommon, and this was during the holiday season. People became burned out, and the team was still feeling the repercussions of that experience. Two of its best people resigned when the project was over, and they would be hard to replace. Management's pizza party wasn't enough of a reward.

What It Meant to the IT Director

Chris felt she had something here. First, she could see that the original deadline for the 8240 project was way too aggressive against the benchmark chart. The actual project performance was clearly excellent, even though the project was two months "late." The team was being beaten up over how long the

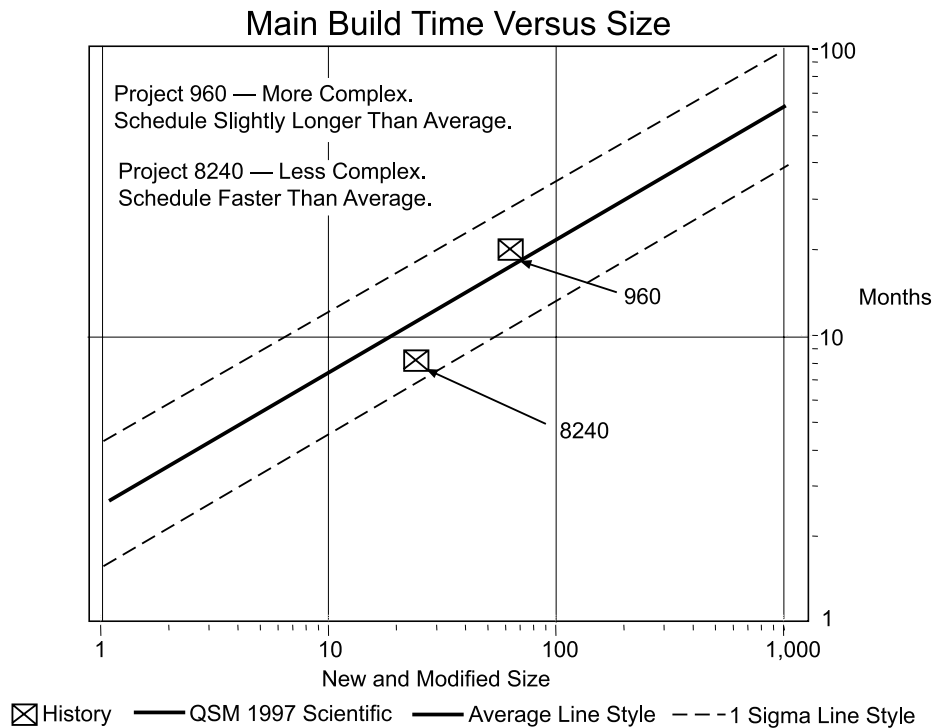


Figure 1 — Historical performance on last two BDMS projects.

project took, but the truth was that it was done *fast*. “Late” was against an arbitrary deadline — and one that was absurd. Rumor had it that it was timed for the last CIO/VP’s birthday.

The really bad news was that the SmartMed 5000 deadline was also almost off the charts based on the functional size that marketing wanted combined with the Geneva trade show deadline (see Figure 2). The team was being set up to do it all over again but without two of its stars who quit after the 8240 pizza party. Fred Alistair was going to really destroy morale with this one. Unless he could be convinced otherwise.

The Metrics Consultant

Steven Martens, the metrics consultant hired by Fred Alistair and Chris Ayari, had seen this before at other companies. He knew this was an organization that had much larger forces in play than was immediately visible. Simply having a metrics benchmark and estimate analysis was only part of the solution. The other parts would play themselves out as the drama unfolded and the agendas of the different players came into light.

Fred was the wildcard. Meetings with him were, on the surface, beneficial. But the fact that he was a new executive meant that no one at BDMS really knew his track record, aside from his reputation. The CEO seemed to place all of his trust in his new number 2 person.

In observing the organization, Steven sensed that decisions were already made in the minds of many of the players — that the deadline and the promised functionality for SmartMed was a foregone conclusion. But Steven decided to proceed anyway and see what he could tease out of the players during the interviews and management briefings.

Marketing’s Dilemma

In his first meetings with marketing, Steven could see that this was an organization drenched in fear. Employees spoke of the company’s archrival as though it was the “Terminator” of the medical products industry. Marketing’s relationship with its developers was tense and adversarial — marketing felt that development didn’t understand what was at stake. All marketing kept hearing were complaints about requirements being inadequately defined and changing. Didn’t

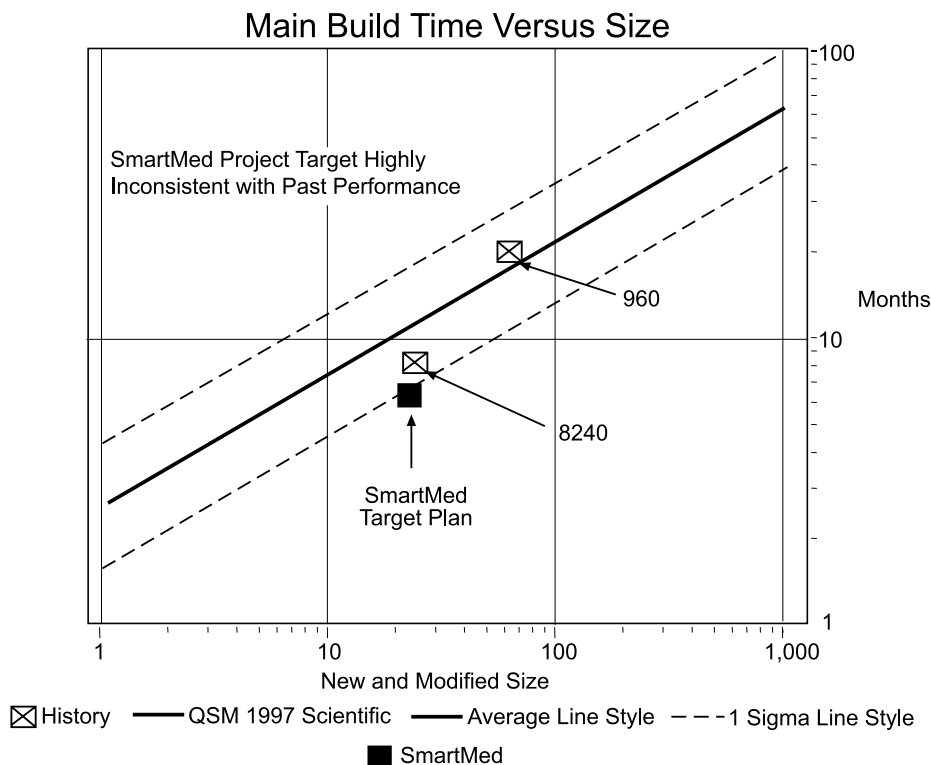


Figure 2 — SmartMed project’s anticipated deadline compared to other historical performance.

developers understand that that was the nature of their business?

Executive staff meetings in the past were mainly screaming matches between the last CIO/VP and the marketing VP. The two had a nasty rivalry and were constantly trying to win influence and favor over the other in the eyes of the CEO. It felt like one big dysfunctional family.

When Fred's predecessor left, marketing was somewhat relieved. However, although Fred came in pledging a new partnership, the staff still didn't trust him. He seemed to be the CEO's new golden boy. Rumor had it that Fred was aiming to make his mark with the SmartMed project, create a hero persona, and align himself to be the CEO's successor. Everyone was suspicious — even the people who worked for Fred.

Those kinds of politics made Steven anxious about how a metrics analysis might be received. He was aware that people often view "facts" through the biases of their own political agendas and personal history. Nevertheless, more of the story would have to reveal itself through day-to-day conversations. Steven felt that being on-site was vital to "taking the pulse" of the situation. He couldn't get as good a feel for how people felt through faxes and phone calls. He decided that doing a series of on-site interviews and assessment meetings would be a good idea.

Pulling Teeth to Get Requirements

With the history of the 960 and 8240 projects in hand, Steven and his team began to make inquiries about how the organization had estimated projects in the past. He asked key project leaders how they transformed marketing requests into time and staffing estimates.

Pretty simple, they explained. They looked at the requirements that were given to them by marketing. Most of the time these were in the form of interoffice memos. Steven asked how complete they were on the 8240 project. Shelly, the project manager, pulled out a yellow file folder. It wasn't very thick. She laid out the memos on the conference table. The first memo was one and a half pages. Steven read it; a shiver went down his spine. "How long into the requirements

phase was this given to your team?" he asked. Shelly looked at the date. "One week before we were supposed to finish the preliminary design," she replied. Steven then asked, "When did the next level of detail come to you?" Shelly replied, "Two months into the detailed design and coding."

"So what were you coding at that time?" Steven asked. She replied, "What we thought they wanted. We wrote a loose spec and sent it to them for review. It was based on our best guess. We kept asking them to make up their minds, but they replied that they weren't making the progress they hoped for at customer focus group sessions. So, we did our best even though they wouldn't give us review comments to the spec." Steven said, "Okay. What came next?"

The Notorious Homegrown Estimating Spreadsheet

"We would take the project requests and compile a running list," Shelly said. "For each request, we would tally what we needed to do to make it happen — 20 person-hours for this task, 40 person-hours for that, and so on. When we added it up, it would come out to some number, say 13,750 person-hours, or about 80 person-months."

Steven commented, "So, for example, that could be expended by eight people working over a period of 10 months. What was the schedule?"

"Not 10 months, that's for sure! What a luxury *that* would have been!" Shelly said. Steven noticed that about four people started laughing.

Bob, one of Shelly's best engineers, said with a straight face, "They told us we had six months for the design, code, and test — the main build phase. No more. We decided to divide 80 person-months by six months, and got 13.3 people as the answer for the number of staff to assign to the project. We rounded up 14 people."

Steven cringed. "What if they told you to do it in four months?"

"Eighty divided by four, I guess," Bob said. "We would have tried to get 20 people on the project. But that would have been *really* unrealistic. The four months, I mean."

Four Key Elements of a Successful Outsourcing Relationship

Today's "e-economy" is driving more and more companies to use outsourcing for competitive advantage. Of course, whenever new relationships are created, conflict can result.

Corporate cultures may intersect or conflict, interests can be divergent, and patterns of communication are likely to be different at best. Time pressures amplify these problems. A conflict that can be managed and resolved given enough time might explode when harsh deadlines are imposed.

Companies that succeed will be those that manage their strategic relationships the best. Core competencies differ from organization to organization, but relationship management must be a core competence for every company.

Here are four key elements to help maximize your chance of success in your outsourcing relationships.

1. No Shotgun Weddings.

Take the time to get your contract right. Time pressures often cause people to rush a deal, which can create structural flaws in a relationship at its very onset.

Be thoughtful about service-level agreements (SLAs). Bad SLAs from ambiguous contracts can create the basis for future disputes. If the relationship sours, it is usually because a client accuses the supplier of breach for delivering a project with less functionality than agreed to, and/or one that's late, over budget, or with unacceptable reliability. Suppliers often accuse clients of having changed the scope beyond the initial contract and say that the requirements were either ambiguous or late.

You can minimize the risk of this happening by taking the time that you need up front, so you don't suffer worse consequences later.

2. Build a Credible Metrics Framework into Your Contract.

Many contracts are woefully lacking in the subject of metrics to quantify expectations and commitments. This is especially disastrous when application development is involved. You'll need to include a core group of measures, such as the Carnegie Mellon/Software Engineering Institute Minimum Data Set, to establish service levels and promised deliveries.

Why? A supplier may commit to a contract price and date, but you'll have to be explicit about commitments on the amount of delivered functionality and quality levels.

Ask prospective suppliers for these metrics on a few of their historic projects from other client engagements, even if they're different than what they're doing for you. Two to four historic profiles will give you a sense of their demonstrated capabilities. Sophisticated, experienced suppliers will have this data. If they manage projects and don't know this at their projects' completion, then by definition they have been out of control.

You'll need the same information for their bid. The promised date, the committed effort and cost, the amount of delivered

functionality, and the expected reliability. You'll then want to ascertain if the bid is in line with past performance. If it isn't, that's cause for discussion.

3. Don't Treat Outsourcing Strictly as a Transaction.

Outsourcing might be seen as simply a business transaction. Yet when we examine deals that have failed, a theme that emerges is how the "relationship" failed. If you imagine that an outsourcing deal is about friendship rather than business, you're setting yourself up for a fall. You may give away too much on a particular issue in order to preserve the relationship, at a time when a hard-headed business decision is needed.

It is equally dangerous to treat outsourcing deals as if they are only deals, without considering human relationship aspects. For example, people who feel valued perform more effectively. Relationships marked by clear and frequent communication are more durable and less prone to conflict.

Consider some kind of training on interpersonal communication to help with those difficult conversations. Why? Because people in technology, although trained in science, might need skills in human interaction and solving interpersonal and intercompany conflicts that were not covered in degreed engineering programs.

4. Include Dispute Resolution Frameworks as Part of Relationship Management.

When you look at mechanisms in business agreements to resolve disputes, you don't often see processes on how to manage conflict — what you find are ways to escalate them.

Good dispute-resolution clauses do involve escalation, in the sense of sending a dispute that can't be resolved at one level up to the next. But all too often, kicking the problem upstairs also means kicking conflict into high gear. The boilerplate language used in some dispute-resolution clauses doesn't address how to de-escalate tension. In the absence of conflict management procedures and practices, the executives brought in to manage the conflict might not solve the problem, and that could lead to arbitration and litigation.

Examine how you plan to manage conflict. Try to understand the causes of conflict, the communication (or lack thereof) that tends to sustain and deepen it, and methods for managing it most effectively.

Consider employing dispute resolution that focuses on joint problem solving around both the substantive/structural aspects and the relationship aspects. You can target the areas directly using a number of intervention tactics and lower your risk of being sapped by conflict and dragging your outsourcing vendor (or being dragged by them) into court.

by Michael Mah and Doug Stone, partner at Triad Consulting Group and Lecturer on Law, Harvard Law School

“Well, how do you know even six months was unrealistic?” Steven asked.

The team looked at each other. No one said a word. Finally Bob said, “We just felt that way. But it doesn’t matter. Around here, the deadline is the estimate. When they tell us six months, we don’t feel we have the leverage to change it. That’s just the way it is.”

**The SmartMed Estimate:
Good, Fast, Cheap — Pick Two**

The team decided to share its findings with Fred at the next development management meeting. Members were hopeful this would help Fred understand the situation and maybe lead him to helping the team negotiate with marketing on the best strategy for SmartMed. That was part one of the plan.

Part two would not be so easy. Even based on the demonstrated productivity history of the 960 and 8240 projects, what marketing wanted IT to achieve on SmartMed was unprecedented. Making matters worse was the loss of two of the group’s most skilled people, plus the fact that, once again, requirements from marketing were in a state of flux. The team had succeeded in achieving high productivity on 960 and 8240, but it was only the result of superhuman efforts and taking the team to the edge of burnout.

IT was in a tough spot. The team ran a computer simulation of the SmartMed project

plan using a macro-estimation model, calibrated with the 960 and 8240 project data, and presented it in a summary slide to Fred (see Figure 3).

These were tough tradeoffs. With the existing team, the January trade show date would likely be missed by seven weeks. There was only a 10% probability of delivering all the functionality by January. Using an alpha version of the software in January (about when testing would be slated to begin) risked an embarrassing series of system crashes in full view of clients and the press at the Geneva show.

The good thing about this scenario was that the costs would be kept down, and when the system was later released in March, it would operate virtually trouble-free with high mean-time-to-defects rate: more than 10 days on average.

To make the deadline, the staff would have to be ramped up to 30 people. Although IT would make the date, that scenario showed only one-fourth of the reliability, with the cost doubled. The January deadline could be made, but with four times the bugs in the system from a chaotic rush of design and coding with an unwieldy team.

Many team members were concerned that the January deadline would be forced upon them without any additional staff. If the team deployed the system too soon, there was a risk that the system would be woefully premature. A reliability forecast of the number of defects and how they were expected to ramp down during the later stages of the project was shown to Fred (see Figure 4).

Most were worried that one of these nasty bugs would sneak by them and make it into a production system out in the field. Although it might make FDA certification by a nose, there were worries that if one of those database corruptions occurred, a patient could be at risk.

One of the team members was especially afraid of that, based on personal experience. He knew firsthand how important these life-and-death decisions were. Last year, he had contracted an infection that made it into his bloodstream. Delirious with a 105°F fever, he was rushed to a hospital. The tests that were run on him helped diagnose the problem and probably saved his life. The lab that

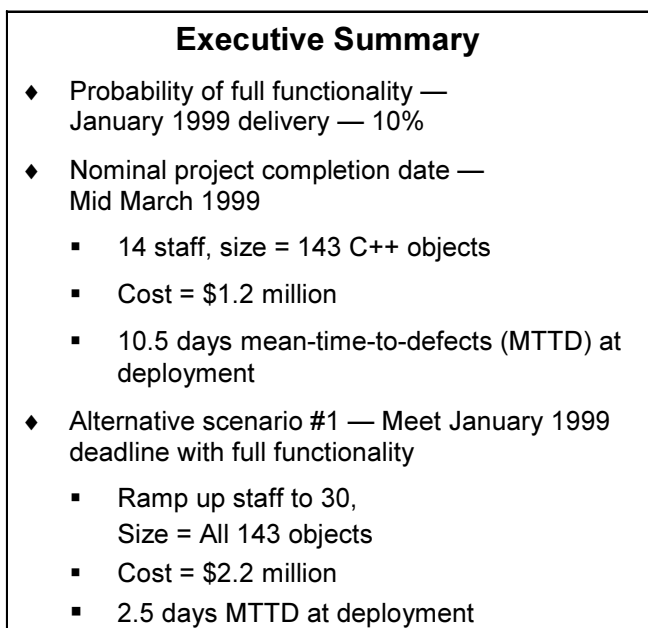


Figure 3 — Executive summary slide of the SmartMed estimate.

Cumulative Defects Remaining Plan by Category
(Expected 50%)

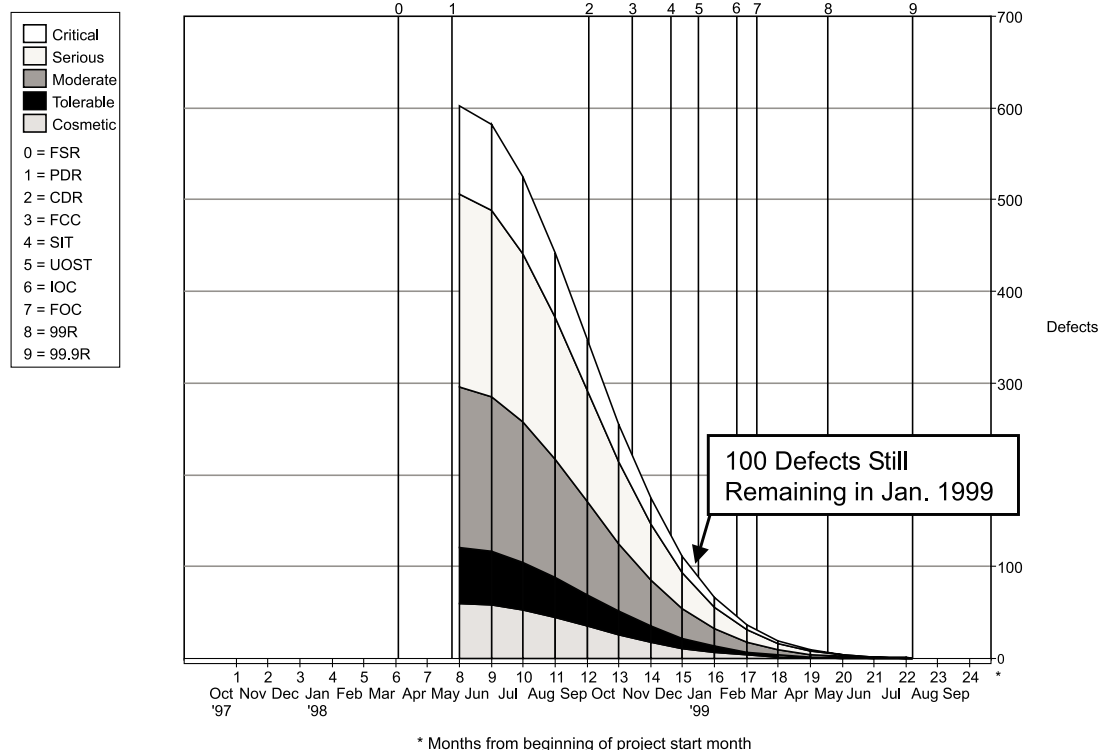


Figure 4 — Defect curve for the SmartMed project.

analyzed his test data was a customer of the 8240 system that he helped design. In a way, he and his team helped save his own life.

If SmartMed was released with 100 known defects in January, it would mean big trouble. One fail-safe was the FDA certification, which was designed to avoid having an unreliable system released into operation. Still, the team didn't want to risk anything getting through. The team clearly needed the additional seven weeks of testing; members figured they could flush out 70% of the remaining code bugs with the extra time.

The Internal Negotiation

This set the stage for the core internal negotiations and the tradeoffs that would ensue. Scenario 1 was to let the project evolve to a March delivery and miss the Geneva trade show. But Fred didn't even trust the March date.

Scenario 2 was to increase staff and ante up an additional \$1 million or more of development cost. The downside to this was that, in

addition to the money, there would be a reliability penalty (higher defect rates from throwing more people on the project with an accelerated deadline). Although this scenario met the deadline requirement, two other parameters suffered.

Scenario 3 was to scale back the features and get critical functionality to market, providing additional features in a subsequent release. That would satisfy the budget, the deadline, and the reliability, but it would deliver less features (temporarily). Marketing would have none of that — it believed the company had to deliver more than its competition, not less.

The best chance of having their cake and eating it too was to achieve an unprecedented level of productivity. This was a gamble — it required, at a minimum, a cooperative effort to keep the project in scope and minimize requirements churn. This would be marketing's primary responsibility. Development stressed that this was a critical issue in the past and it would need to be solved if there was any hope of achieving higher output. It was especially critical because the team had suffered the loss of two

of its stars. The rest of the team would feel that loss and have to make up for it.

As a risk buffer, it was determined that, if necessary, the wish list of noncritical features could be scaled back.

What the CIO/VP Had in Mind

Fred didn't like any of what he heard. None of the scenarios allowed him to feel he had a high likelihood of making his mark in the eyes of the CEO. All of them required a tradeoff in a dimension he had not counted on.

What's more, Fred felt his authority was compromised — whatever he decided would be subject to debate. For the time being, he went with a plan that sought to deliver full functionality even though it was risky, hoping for productivity to hit the levels that would be needed.

However, he decided that if the early milestones were missed, he would not hesitate to act swiftly and decisively. He began discussions with an outsource supplier.

When the Going Gets Tough, the Tough Decide to Outsource (the Wrong Way)

The company decided to place a bet on none of the three primary scenarios. Instead, Fred decided to “shoot the moon,” modestly ramping up the project staff to 20 people. While he was at it, Fred decided to discontinue the metrics analysis, since he felt he might not be able to control the information that would be derived from it.

In a short time, it became clear that despite the best efforts of the team, the aggressive milestones were not being met. Requirements from marketing remained sketchy well into the build phase. On the side, a company called BuckStar International was asked to submit a proposal to build SmartMed under Fred's terms. Full functionality, January delivery, high reliability, the right price. “Of course, we can,” proclaimed BuckStar, and the deal was done. With that bold contract stroke, the internal SmartMed initiative was terminated. Most of the SmartMed team was fired in an effort to cut expenses.

Steven Martens found this out after the fact in a phone call to Chris Ayari. Chris explained that she was allowed to keep a

core team of four people to oversee the BuckStar outsourcing project. The difficult task of laying off people was given to Chris (by Fred).

The Crucial Moments

As described earlier, BuckStar did not deliver as promised and wound up in a legal dispute with BDMS. Fred Alistair was fired in disgrace by the CEO, who was absent from the scene until the very end. Chris was left with the task of desperately trying to recover the situation, to no avail.

Looking at this scenario in hindsight, it becomes clear that there were several critical junctures in the story. First and foremost was the application of metrics — too little, too late. The organization perceived itself as having failed to deliver on two previous projects. As it turned out, these “failures” were subjective perceptions; actual performance was quite good when held up to the light of a benchmark, after the fact.

The failure perception caused critical information that could have led to future success to become buried. Management and team members were not anxious to perform deliberate postmortems for fear of what might be discovered. Fear of being exposed became the predominant emotion, and coverups ensued.

When the information was finally produced, there was a chance for success. But the hiring of Fred Alistair destroyed that chance. Fred's personal agenda would lead to the suppression of key information that, ironically, could have been leveraged for his success and the success of the company.

Imagine if he had used the metrics analysis to execute a development strategy that maximized the internal IT productivity of the BDMS team. Further analysis could have been undertaken to promote the organizational learning, with data being used in feedback loops to improve practices that result in reduced cycle time. Some of these findings were beginning to emerge, but they were prematurely truncated.

Fred could also have used the metrics to benchmark BuckStar's proposal to BDMS. It would have been possible to identify whether the proposal was a fairy tale

designed to win the contract, or if the company had a track record that demonstrated the capacity and speed that BDMS needed. But Fred acted without this kind of analysis to support his decision, discontinuing the metrics initiative when he began to view it as a threat. Had Fred applied the concepts of metrics for bid evaluation (see “Evaluating Bids: First the Facts, Then the Acts,” on page 1), the outcome could have been entirely different.

Summary

- **The setting was an ad-hoc culture with no history of measurement.** When measures emerged, an antilearning dynamic set in at senior management levels because of politics and fear.
- **The setting also involved an internal rivalry between marketing and development.** This was compounded by a lack of leadership at the CEO level. Thus, a CIO/VP with personal ambitions was able to disregard any wise advice that was offered.
- **Warring factions** meant that there was never any interest-based negotiation designed for collective benefit.
- **The company employed a non-democratic, authoritarian decision-making process.** With the power of decisionmaking comes a responsibility to use that power wisely; this does not always occur.
- **An obsession with time to market resulted in loss of time to market.** Demanding a “perfect” answer led to a worse-than-mediocre outcome because of self destructive behaviors by the organization.

The following should have been done differently:

- **Use metrics for risk management.** The business benefit of this would have been *huge*. In the end, whether the company outsourced or not, the result could have been a system that gave it market share instead of a canceled project.
- **Require outsourcers to provide factual basis for proposals.** This would have been in the interests of both the supplier and BDMS. In the end, signing up BuckStar was not in either party’s interest. BuckStar’s proposal had no validity behind it, yet management (in the form of Fred Alistair) bet a critical product on it.
- **Provide application development life-cycle training for project managers.** This would have avoided the faulty estimation processes from the simple spreadsheets that were initially in use. Poor estimates on past projects were the result of this flawed logic.
- **Encourage active leadership from the CEO.** The most senior executive was asleep at the switch. Through his inaction, key pieces of knowledge from the teams were not leveraged, processes suffered, and divisive energy split the company. The CEO could have intervened at critical decision points.
- **Realize that schedule conflicts reveal deeper organization issues.** First-order problems of missed deadlines underscore more significant second-order intergroup and interpersonal conflicts. Attending to these fractures through training in interest-based bargaining and dispute resolution might have diffused the tensions that crippled teamwork and sapped the energy of an already weary organization.

Evaluating Bids

Continued from page 1.

extensive software system came in at US \$86 million. The next-highest bid was \$147 million. Was the high bidder realistic? Or gold plating? Was the low bidder realistic or just exceptionally hungry? Because of the size of the contract, final approval had to come from

the board of elected politicians. Now, we make a lot of fun of elected officials, and they are forever getting caught as disasters cave in on them. Still, to endure, most of them have at least a measure of “low cunning.”

“The difference was so substantial, it looked bizarre,” one of the county supervisors recalled at a 1998 hearing called to add \$52 million to the project. But, he went on, the less expensive bidder had assured the board it could bring the project in on budget.¹ Of course, in 1998, the Welfare Department was still in need of the system. By then it was apparent that the higher bidder had been the realistic one. The county had little choice but to grant the additional millions and more time.

This gloomy scenario is played out hundreds of times each day somewhere in the world, although usually for lesser amounts of money. There are still software organizations without much knowledge of project estimation, yet which need business to survive. There are still customers, lacking knowledge of the sort we outline in this article, who feel pressured to accept the low bid. When these two parties go bump, odds are that when the time and effort set in this low bid are nearing exhaustion, the project will be far from complete.

Then the two damaged organizations must search for reality, very likely in a bitter atmosphere, possibly even in a courtroom. At best, it is a place and time for the skillful negotiation that Michael Mah champions! Obviously, it would be better to handle the up-front relationship in a more skillful way, so as to avoid this down-rear confrontation altogether.

Client and Contractor Must Cooperate

There is some kind of conspiracy to keep the facts about software development a secret. The conspirators tell us, “Just get coding under way, and all will be well!”

The anticonspirators contend that software development is a complicated activity. The elements of requirements, functionality, architecture, and size are uncertain far into the early phases. Moreover, many of those elements are the responsibility of the customer organization. The rest involve the technical and operational experience of the software organization.

¹Nicholas Riccardi, “County Computer Contractor Assailed,” *Los Angeles Times*, 21 December 1998.

To be brief, the initial uncertainties encompass the scope and definition of the project and the substantial risks that might derail the project. These uncertainties make it difficult for the client and the contractor to define the functionality to be provided by the eventual system. Lacking a grasp of functionality, it is difficult to estimate the likely size of the system — the first fact in estimating time and effort. In turn, the estimate of time and effort underlies the contractual relationship — the bid — between the client and the contractor.

In this maze, the only basis for success is cooperation between customer and client — a deeper cooperation than the arm’s-length relationship often found. The first step toward this collaboration can occur in the bidding relationship. The client seeks from each potential bidder the core metrics on its recent projects against which to validate the bid on the current project. In other words, the core metrics being bid are calibrated against the core metrics representing the contractor’s actual experience.

There are only four of these core metrics:

1. Size of each past project in source lines of code, function points, or other measure the contractor might use
2. Time, that is, schedule length for the main build, or beginning date and ending date
3. Effort, in person-months, devoted to the main build
4. Defects, such as number identified and removed during the project, or estimated number remaining at delivery

Some bidders may object to providing data such as these on the grounds that it is confidential information. The general answer to this objection is “cooperation.” In the first place, it goes almost without saying that the client will hold this data confidential. In particular, it will not be shared with other bidders. In the second place, the client will use this data only to assess whether or not bids are realistic.

It is not in a contractor’s interest to win a contract at a cost too low and a schedule too short to provide a product with the needed

capabilities and appropriate reliability. It is not in the client's interest to not get the product at all (as all too many surveys have pointed out). It is not in the client's interest to not get it at the point in time coordinated with the client's other business activities. It is not in the client's interest to have to come up with a few million unplanned dollars to complete the needed work.

Some bidders may simply be unable to provide these metrics. That is good for the client to find out up front. How can a contractor make a valid bid if it doesn't know what it has done before? If you have a little piece of creative work, where schedule and cost are not critical, you might throw it this bone. Such a bidder is not a good choice for anything serious.

How Software Development Behaves

Let's say we have a bidder that wants to cooperate. It provides core metrics on three or four recent projects with its bid. From the client's point of view, we want to know two things:

1. Is the bidder's performance on recent projects reasonable?
2. Is the new bid in line with this performance?

Answering the first question calls for some knowledge of how software development behaves. Figure 5 illustrates the schedule behavior of the more than 5,000 projects in our database. Note that both axes of this diagram are logarithmic, enabling us to cram a vast range of size and schedule on a small figure. There are at least three points of interest to the student of software behavior:

1. There is an "impossible zone" — a short development period in which no one has ever completed a project.
2. There is a very great range in schedule length (the vertical dimension on the figure) at each system size. The bidder, at the very least, ought to project a time duration longer than the minimum at the estimated size.
3. Categorizing the data points by the three major application types, we find that information systems are completed in less time than engineering systems or real-time systems. Depending on the application being bid, the schedule ought to fall within one of these ellipses.

When we plot effort against size, or a defect metric against size, we get figures that look much the same as Figure 5. That is, they show a vast range of effort or defects at each size.

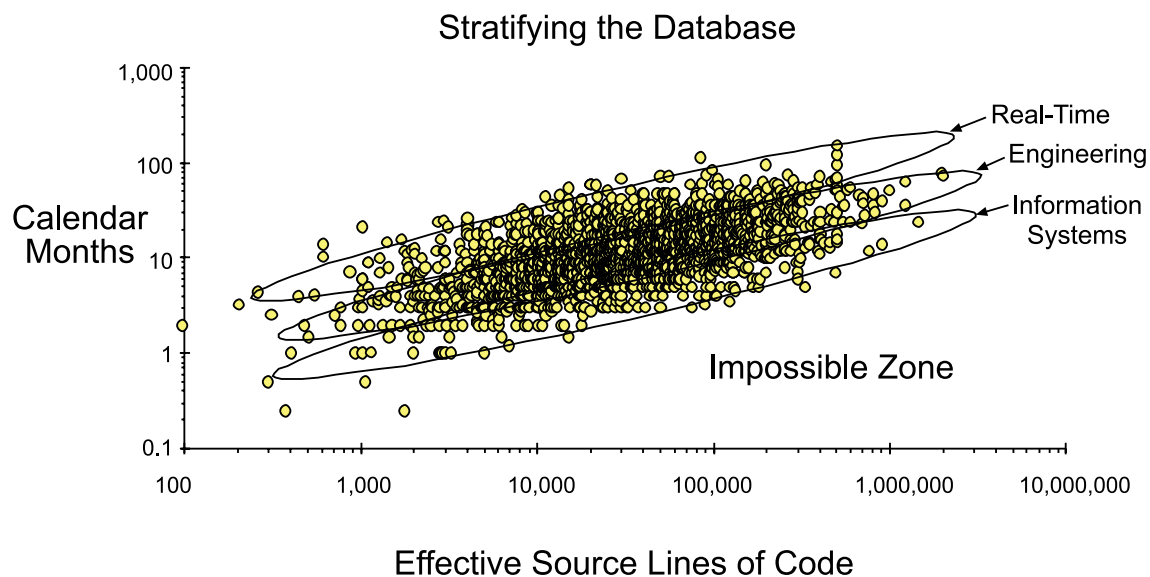


Figure 5 — According to our data, thousands of projects behaved in this way. The bidder's historic projects and its bid ought to be within this "ballpark." (Source: QSM mixed application database)

Again, the bidder's data on effort or defects ought to fall within the corresponding ellipses representing the range of effort or defects.

We can draw a centerline through the large ballpark of Figure 5, as shown in Figure 6. The centerline is the mean of the schedules represented in the database. The upper and lower lines (each one standard deviation from the mean) enclose about two-thirds of the projects. In effect, these are the projects that developers accomplished at a productivity rate for their process somewhere near average. They represent what a client might reasonably expect from the average bidder.

In Figure 7, we erase all the database points and show the mean and standard-deviation lines. Two of the small circles represent the bidding software organization's historic schedule data. It has done better than average — that is, its schedules were shorter than the mean (but well out of the impossible zone). The third circle is the bid. It shows the same level of schedule performance as the historic projects. The client may reasonably conclude that the bid is realistic with respect to schedule. The client can use similar figures to evaluate effort and defects.

Evaluating the Bid

The traditional thing bidders do in a proposal is embroider it with lots of fluff. At least that is the way the plethora of glowing adjectives comes through to the jaded consumer of proposals. Getting bidders to submit their core metrics brings a whiff of fact into bid consideration. Then, locating those facts on historically based diagrams, such as Figure 7, injects a big dose of realism into bid evaluation.

The historical data points of some bidders may fall well outside the upper and lower boundary lines of Figure 7. Those on the more productive side of the boundary (lower, in the case of schedule) are most welcome. Those on the less productive side are questionable. Most will be in the middle area and are acceptable.

The next step is for the evaluators to note where the project being bid falls. It should be near the bidder's historic data points. Suppose it is in a much more favorable location. Why would a bidder do this? It may be trying to buy its way into the field by submitting a bid that is low on cost, effort, or schedule. That may be good for the client, if the bidder can afford it. It may also lead to trouble later on — a la the Los Angeles County example.

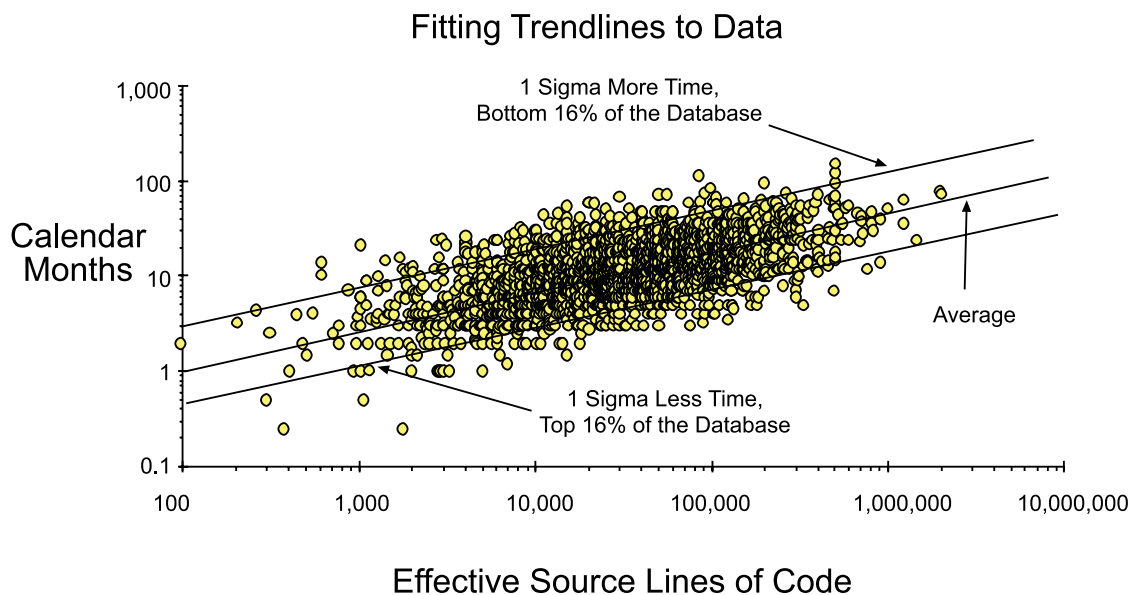


Figure 6 — The bid schedule might fall within the upper and lower lines (one standard deviation each) if the bidder's process productivity is near average and the bid is realistic. (Source: QSM mixed application database)

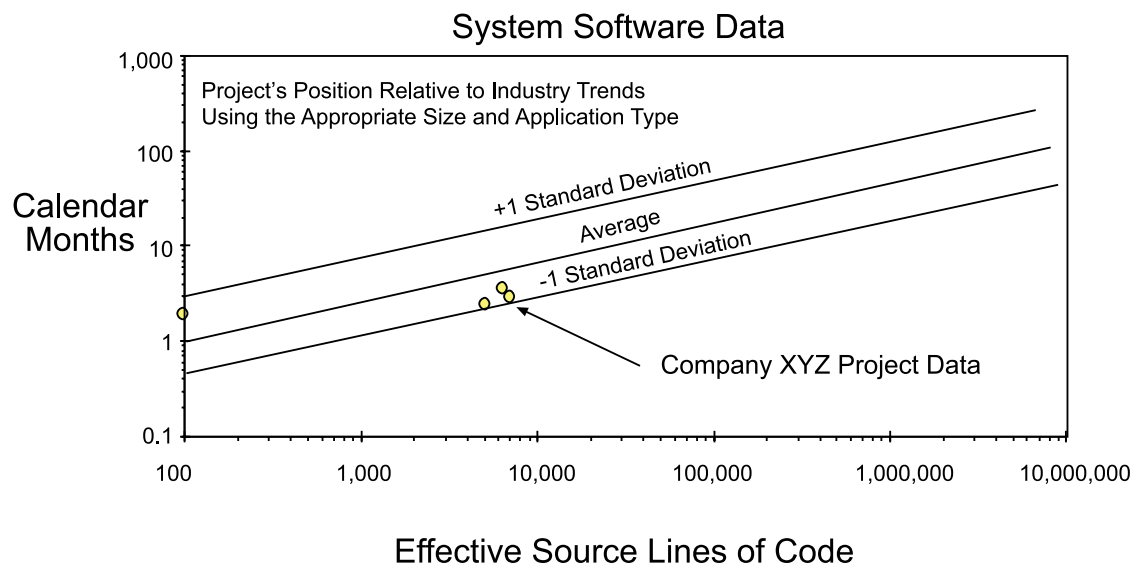


Figure 7 — The bidder’s data, shown in small circles, is better than average on schedule length.

Another answer may be that the bidder is overestimating the process productivity it will achieve in the coming period. That’s a red flag. Our experience in measuring process productivity over more than 20 years is that it improves slowly. Significant improvement, even when it is worked at assiduously, takes several years.

Still a third answer is that the bidder has underestimated the amount of functionality to be provided. That translates into a smaller size estimate and that, in turn, into a lower bid. That is an alarm signal to engage in a little of the cooperation we urged earlier!

On the other hand, suppose the bid data is in a much less favorable (to the client) location. The bidder has “gold plated” the project. It is unlikely that the bidder expects its process productivity to decline, although in the real world that sometimes happens. One possibility is that its schedule is crowded, it doesn’t much want the job, but it will accommodate the project at a price high enough to cover the costs of expansion. Another possibility, of course, is that the bidder has overestimated the amount of functionality; it is contemplating more work than the client is. It is a good idea to bring these conflicting points of view into harmony early, rather than later.

A large set of reasons for high bids or low bids may turn up, but at least they are grounded in the facts provided by the four core metrics. The client can locate the bid in relation to the bidder’s recent history. The client can see where it lies in relation to the background data provided by a large database. The client and the contractor are immersed in reality, not in fancy language. The bid-evaluation team can smoke out the games that vendors play with the proposal process.

At this point, the evaluators can put aside the bids that are unrealistically high or low. The remaining bids are realistic and reasonable. The evaluators can compare them to see which offers the lowest cost (effort), the fastest schedule, the highest reliability, or the combination of these elements best suiting the client’s situation. The evaluation team can weigh these factors along with other factors it normally considers. In considering all these factors, the team has the facts in hand that enable it to give proper credence to the realities of schedule, cost, and reliability. It can assess these realities in light of what each vendor has been able to deliver in the past.

Factual Evaluation Is a Proven Practice

We have been privileged to help with a number of software bid evaluations over a period of more than 10 years. To show our level of experience, here are a few examples:

- The now privatized telephone company serving the Netherlands
- The Scottish Criminal Records System (Jim Greene of our London office)
- A procurement run by the Naval Research Laboratory
- US Navy Stand-off Land Attack Missile
- US Navy E2C radar patrol aircraft
- The Joint Mission Planning System (US Navy and Air Force mission planning software)

With metrics, realism gets into the bid-evaluation process. Historic fact replaces storytelling in the proposal. The selection board deals with fact rather than fiction. The client, with its control of the “money button,” occupies a commanding position!

About the Authors

Ware Myers has a bachelor’s degree in engineering from Case Institute of Technology and a master’s in management from the University of Southern California. For a decade he was an extension instructor in personnel management and training, and for another decade he was a lecturer in engineering organization

and administration at the University of California at Los Angeles. Along the way he put in stints as a safety engineer, sales engineer, Labor Department representative, naval officer, training officer, vocational advisor, production control manager, systems analyst, drafting instructor, personnel director, engineering writer, contributing editor, and writer of software books. As a production control manager, he learned that estimates often fail to work out as hoped. So when he first heard Larry Putnam in 1977 explain his views of software estimating, he applauded their merit in the form of a long article in *Computer* magazine in December 1978. That led to their collaboration on three books and many articles and columns. Mr. Myers can be reached at 1271 North College Avenue, Claremont, CA 91711, USA. Tel: +1 909 621 7082; Fax: +1 909 948 8613; E-mail: myersware@cs.com.

Lawrence H. Putnam has more than 25 years of research experience in software metrics, measurement, estimating, and control. He founded Quantitative Software Management in 1978 and continues today as its president. His most recent work employs statistical quality control to gain effective control of in-progress projects and to measure the commercial benefits of process improvement. Mr. Putnam has a bachelor’s degree from the US Military Academy at West Point and a master’s degree in physics from the US Naval Postgraduate School. He can be reached at larry_putnam_sr@qsm.com.

Please start my subscription to *IT Metrics Strategies*® for one year at \$485, or US \$545 outside North America. Phone Megan Nields at +1 781 641 5118, or fax +1 781 648 1950, or e-mail info@cutter.com.

Please renew my subscription.

Name _____

Title _____

Organization _____

Dept. _____

Address _____

City _____ State/Province _____

Zip/Postal Code _____ Country _____

Tel. _____ Fax _____

E-mail _____

Payment or purchase order enclosed

Please bill my organization

Charge my Mastercard, Visa, American Express, Diners Club, or Carte Blanche

220*5ITS

Card no. _____

Expiration Date _____

Signature _____

**Web site: www.cutter.com/itms/
Cutter Information Corp.
Suite 1, 37 Broadway
Arlington, MA 02474-5552 USA**