Presents
An IT Metrics and Productivity Journal Special Edition

Focus on Dr. Herb Krasner, University of Texas and Krasner Consulting
A CAI State of the Practice Interview
June 2005

Biography of Dr. Herb Krasner:

Herb Krasner is a senior faculty member at the University of Texas at Austin, and the Director of the Software Engineering Industry Affiliates Program. He is also a successful software excellence consultant. His personal mission, spanning several decades, has been to enable the development of superior software, and to stamp out poor quality software, wherever found. He is best known for his leading edge work on modeling the costs of software quality, reporting the ROI data for software process improvement, coaching organizational improvement programs and reporting the results from his empirical studies of professional programmers. He has published over 55 papers, articles and book sections, has spoken at many professional conferences and meetings, and is active in professional organizations and societies. His current research interest areas include empirical studies of software engineering, the human factors of software engineering (e.g. teamwork models), agile development methods, software design paradigms, software engineering process improvement, and software engineering best practices.

◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆ ◆

CAI: What in your opinion, if anything, is going to prevent all IT service operations from being offshored over the next 15 years?

KRASNER: First of all, I assume by the meaning of IT service operations that we're talking basically about IT organizations that enable companies to deliver products and services. I want to make sure that we have that context because normally I use the term 'IT' in a much broader scope.

That being said, I think the main things that are going to prevent all IT service operations from being offshored are things that we are already seeing. We are already seeing personnel shortages of experienced software engineers in the Indian marketplace. As it turns out, they can't hire and bring in people any faster or better than we can. And given the amount of work we've already offshored to them, the kinds of people they are bringing in now are not quite as talented. That's one point. Another related point is that, as a result of this, the good people are doing quite a bit of job hopping from place to place and asking for more money. And that just exacerbates an already diminishing cost advantage. It used to be a five-to-one cost advantage and now it's something like two-and-a-half to one.

A second issue is quality. We are simply seeing too many poor quality results in too many situations. I don't think the Indian success rate is that much better than ours.

A third issue revolves around something that I like to call 'domain knowledge'. Basically, for IT to be as effective as possible in supporting a business, IT has to understand the business. And the people who understand the business and the business processes are the ones back here, by and large. The expectation that somebody offshore would have the same kind of business domain knowledge is not very realistic.

You take all of these things together and it seems to me that the pace of IT offshoring will probably slow down and in some cases actually reverse, with a move towards onshoring, nearing, and insourcing. We're already seeing this, to some extent.

CAI: What, if anything, is going to prevent all business process outsourcing service operations from being offshored over the next 15 years?

KRASNER: I think that's basically the same issue. To the extent that IT services are closely coupled to business processes, I think that's exactly the same thing.

CAI: It's been 20 years since we first heard about SEI and CMM from Carnegie Mellon. What is your opinion of the progress that has been made to date, and what percentage of penetration do you see in large organizations?

KRASNER: First of all, you have to understand that the CMM was a model that was designed for particular situations and not all situations. For example, the CMM— as it is defined in the books and the technical reports— is not particularly well suited to maintenance organizations or to IT organizations. It has to be heavily adapted and modified in order to fit those kinds of organizations. In contrast to this, large government organizations and government contract organizations are actually very well suited to the CMM because that's basically who it was designed for. In these kinds of organizations, the penetration has been quite good and quite widespread. The problem has been that people hear about the SEI and the CMM and, without much understanding, they think they can take it and implement it into their own organizations. And then they fail. The folks at SEI used to say that the failure rate for applying the CMM was around 60-70%. And what they meant by this was that of organizations doing some kind of initial assessment using the CMM, 60-70% of did not show any improvement whatsoever. That's the bad news.

The good news is that the other 30-40% have in many cases shown significant improvements. If you read my paper *Accumulating the Body of Evidence for the Payoff of Software Process Improvement*, you will see some pretty amazing results. The nice part about this is that, as a result of the success of these organizations, everybody now seems to know about the CMM, at least in general. Whether people have tried it or

not, they seem to know about it. But the problem here is that these organizations will run into big problems if they think they can use the CMM in their own organizations regardless of whether or not it is well suited to them. They'll get lost in the details.

CAI: If the CMM is better suited to large organizations, then what can one do to help improve the processes of a small organization?

KRASNER: Although I started my career with very large organizations (e.g. Harris, Lockheed), I now spend most of my time with smaller, more entrepreneurial organizations. They are usually under severe resource constraints and delivery pressures, their process problems are dominated by people issues, and their organizations tend to have a preference for informality. So there is usually no defined process at all, nor any focus on process, and the prevailing attitude tends to be that process only sells as a means to getting short term results; in other words, the payoff better be real close to the bottom line in terms of profitability, customer satisfaction, and quality concerns. In these cases, process definitions can and should be very small and CMM principles must be boiled down to practical nuggets of useful information.

Sometimes the biggest quick win in these cases is to help write good job descriptions for new software staff positions, help them staff up, and help them align existing skill sets with a new organizational approach to development. It's here that I often help bring key people into key roles, i.e. properly qualified people who will bring the process with them. Since such experts tend to dominate team approaches in a small organization, improvement progress can be rapid. The challenges, as always, lie in changing the commitment behavior of executives and of marketing and product line managers who may not understand either software engineering or disciplined process management concepts.

CAI: Could you tell us a little bit about the historical development of the CMM and how we got to where we are today?

KRASNER: The CMM has actually undergone significant transformations over twenty years. In 1986 the CMM started off as just a simple questionnaire. It consisted of 101 yes/no questions without an explicit model of best practices behind it. It existed like that for four or five years until the early 1990s when the CMM staged maturity model was described and published. This first model was used strictly for project-based, new software development.

In the mid 1990's several other non-software maturity models were developed at the SEI, and all were brought together (in 2002) in the latest CMMI suite of models. This new suite is much more complex and much more specific to the needs of the Department of Defense, the SEI's customer. Over time, the CMMI has evolved into something much more suitable to large military/government contractor type of

organizations and therefore much less suitable to commercial and IT shops.

Nevertheless, the original CMM (for software) is still in widespread use by industry. And even though it is no longer supported by the SEI, it has become a world-wide standard for guiding software process improvement programs.

It's important to understand this evolutionary context when you talk about the CMM – because it is not just one thing.

CAI: Are you surprised by the number of organizations still struggling at Level 1 and Level 2?

KRASNER: No, I am definitely not surprised. I deal with a lot of them. I deal here in Austin, for example, with a very entrepreneurial software community. We don't have huge manufacturing companies or huge government contractors. When a software company starts up they typically start up without thinking too much about the processes they are going to use. They hire a bunch of good people, they turn them loose on a first product, and then they worry about process five years later. Most organizations actually start up in this manner, at CMM Level 1 (ad hoc) or CMM Level 0 (ad lib), and then later on figure out that they need discipline to continue being successful. It's at that point that they start worrying about process and how to become more mature.

There are a couple of exceptions to this. For example, in the mid 1990's when Motorola started their offshore software factories, particularly in India, they started these organizations at CMM Level 3 and then hired people into the process. So with some planning and forethought they were able to create CMM Level 3 factories from scratch. They were able to do this because Motorola, as an organization, had the process maturity and experience behind them already so that when they began creating new organizations in India and Australia and some other places they had top level management policies and plans already in place to enable CMM Level 3 organizations. They essentially had the blueprints for CMM Level 3 organizations and when they hired their directors and managers they handed them these blueprints and told them that this was the kind of organization they wanted to create. And then they went out and hired people to fit into that mold. No retrofitting was necessary.

CAI: For those IT organizations that have had a reasonably strong CMM program and CMM implementation, what would be your characterization — in aggregate — of the resulting improvements in productivity, professionalism, and effectiveness?

KRASNER: Since your question revolves around IT organizations that have had a reasonably strong CMM program and implementation, let me first say that we don't know about too many of those. That's because the CMM has not been particularly well

received by IT organizations. IT organizations are involved in a broader variety of activities. They are not necessarily focused on new system or new product development. They do a lot of legacy work and CMM is very weak in this area.

Nevertheless, for those organizations that have figured out how to augment the CMM to fit their own particular situation I have seen a variety of different things. I have seen modest gains in productivity improvement. I have seen complete turnarounds. And I have seen huge gains such as 500-600% improvements in ROI. So it's hard to characterize the aggregate because the variation is so huge. But generally speaking those organizations that are serious about it and that understand and are committed to continuous improvement are going to see an order of magnitude gain over a five to ten year period.

CAI: Are there any other key critical success factors that you can think of for organizations that undertake the CMM?

KRASNER: The most important thing is how committed you are to continuous improvement and that is something that can only come from senior management sponsorship. One of the things we always notice is that when senior management changes and somebody comes in who is not so committed to the improvement program it dies pretty quickly.

Another primary success factor revolves around the way in which you approach this. You are basically asking people to change their behavior. And people change pretty slowly. And there are known techniques for facilitating change. And if you don't have internal expertise in organizational change you better go find it somewhere or your organization is simply not going to change.

A third success factor involves the right improvement incentive. The biggest success stories we have seen have been in organizations that have failed miserably before they began their improvement programs. Pain leads to change. In this sense, huge project failures can be tremendous motivators.

Another factor is the measurement discipline. If you don't see the change, as demonstrated in real metrics and measurement, then how can the change ever get reinforced? And if it doesn't get reinforced how will it ever get institutionalized? This is something that differentiates the very successful from the unsuccessful organizations.

CAI: Given all of the advances in methodologies and CMM tools we still have data* that points to the fact that IT software productivity has remained flat for 10 years. Why do you think that is? (*source: IT Metrics Strategies @2001 Cutter Information Group)

KRASNER: I think it's probably true that IT productivity has remained relatively flat

over the past 10 years. But let's interpret this in the context of the fact that the demands on software functionality have grown by an order of magnitude in these past 10 years. What we are being asked to do is so much more complex that what it used to be. On top of that, the technology that is underneath all of this is much more complex. We've got layers and layers of software in modern systems. And the systems we have to interface are more complex, too. So complexity has gone up exponentially and yet productivity has remained flat. I see that as great news. What this says is that our strategies and methodologies for managing increasing complexity are working.

Just look at the size of the systems we are tackling these days. They are millions of lines of code. 10 years ago they were hundreds of thousands of lines of code. If you have some way of measuring size, and I am not suggesting that lines of code is the correct approach, but whether it is function points or object code bytes if you take a hard look you will see that the size of these things is growing wildly. The data alone that we store in our databases has grown 100 fold over the past 10 years.

The fact is, if we hadn't had engaged in process improvement activities and if we hadn't evolved our methodologies and created technologies to help build these systems our productivity would have gone down dramatically. And that's the reason process is so important, because as the size and complexity of these systems continues to grow our processes are going to have to scale up. And most organizations are just not good at scaling these things up.

CAI: In the next 10 years what do you feel has to happen in IT for us to see significant advancement?

KRASNER: First of all, I'm going to focus on two key words in that question: "to see". I firmly believe that we have to see things to recognize that we've had advancements. In order to see the advancements or the gains we've got to measure. That means we've got to have improved measurement discipline. And most software organizations don't measure worth a hoot. That's my main point.

My second point is that the software process improvement initiatives in an IT organization are going to have to be coupled more closely to the business process improvement initiatives. As software process improvement and business process improvement get closer together I think we are going to see extraordinary advancements.

I also think we need to focus more on the development of talent, of people. You can never get away from the fact that it's the best people who develop the best software. So I think that educating people, accelerating their learning strategies, and helping them improve their own careers is really key. From my point of view, the organizations that will succeed in the next 10 years will be the ones that learn the fastest. So, to the extent that we can enable the accelerated learning processes that must go on in IT organizations, I think we will see advancements. Unfortunately, most IT organizations don't do a very good job of this today. They don't invest enough in developing their

talent, their people. The assumption seems to be that you hire a good person and they have all the knowledge they need to do the job but the fact is, the technology is simply changing too fast.

[The opinions expressed in this interview are those of the interviewee and should not be construed in any way as the opinions of the University of Texas]

Questions? Suggestions? Comments? Please contact the IT Metrics and Productivity Journal Editor at michael_milutis@compaid.com.