Defect Prioritization with the Risk Priority Number

featuring Julie Cohen and Will Hayes as Interviewed by Suzanne Miller

---

Suzanne Miller: Most software systems have defects that are identified by users, but it is not financially feasible to fix all of those defects as they arise. In fact, some defects may never be addressed. Organizations in government and industry must instead prioritize which defects to address and which to implement in the next incremental release, especially in a financially constrained environment. In today's podcast, we will introduce a technique that helps users, quantify the relative priority for addressing known failure sources. Welcome to the SEI Podcast Series, a production of the Carnegie Mellon Software Engineering Institute.

The SEI is a federally funded research and development center sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University. A transcript of today's podcast is posted on the SEI website at sei.cmu.edu/podcasts.

My name is Suzanne Miller. I am a principal researcher here at the SEI. Today, I am pleased to introduce you to two of my colleagues, Will Hayes and Julie Cohen. In today's podcast, we are going to be discussing their research on using the risk priority number, which can be used to help large enterprise organizations prioritize what defects to address first. But first, a little bit about our guests.

Will provides direct lifecycle management support to major software-intensive programs in government and military organizations. We actually worked together on one of those projects. He also researches and consults in the application of Agile methods in highly regulated settings and the innovative application of measurement and analysis methods. Will has been a previous guest on the podcast series where he discussed his research in Agile metrics.

Julie Cohen has been at the SEI for 12 years. She is currently involved in activities to support and improve acquisition practices in the Air Force and civil communities. She leads the SEI support for the Advanced, Extremely High Frequency (AEHF) satellite ground system software effort

. In addition, she is a major contributor to the Veterans Health Administration [VHA], including long-term support of efforts to modernize the VHA's scheduling system. She also contributes to other customer efforts and non-internal research projects like this one. Welcome, Will and Julie.

Will Hayes/Julie Cohen: Thank you.

Suzanne: Before you explain the technique that you are promoting, I would like you to give us an idea of the challenges that organizations face in prioritizing which defects to address. Why is this problem hard for organizations?

Julie Cohen: One reason that it can often be hard is because there can be lots of people who are concerned about the system. There are developers. There is the government organization that is sponsoring the work. There are users, and there can be many varieties of users. They all have their own priorities as to what they think should be fixed first. Often, those are in conflict.

Suzanne: So, in the case of a scheduling system, for example, a doctor may have a different sense of priority than a patient in some cases.

Julie: Absolutely. And, even a different sense of priority than the Veterans Health Administration…

Suzanne: Enterprise.

Julie: …enterprise and the IT group that is actually in charge of building the system.

Will: One of the sources for these differences is the different scenarios of usage that different groups might have for a common system. So my operational concepts and scenarios weave a particular path through the functionality of the system, and yours might weave a different path. So a defect that exists on my path is not really a high priority for you, but it is everything to me because it blocks my ability to use the system.

Suzanne: OK. That is a good way to put it. These are enterprise issues because we have got lots of systems that we have in play, and we only have a certain amount of money. So, it is not even just a matter for this system, we only have a certain amount of money to prioritize, but at an enterprise level, we often have to look at multiple systems. There is a scheduling system for VA and there are other systems, for example, that they have to deal with. So, this gets magnified. Am I correct that this gets magnified when you take it to the enterprise level?

Julie: Yes. Certainly at the enterprise level. Although I will say that many government and civil agencies don't generally take it up to this level. So, if you are working at a lower level, it works

just fine on a single project, but if you are at an enterprise level, it would certainly help a lot if you could use the same type of system across the enterprise, and then that would help clean up…

Suzanne: Then you have a common language.

Julie: Right. You have a common language to discuss what is important and why it is important.

Suzanne: OK. So, let's talk about the *it* in the room that we have just been dancing around. Tell us about the technique you are working with, where it comes from, and why you feel it can address the problem of prioritizing defects.

Will: The technique called risk priority number comes from a background of failure modes and effects analysis, which is an element of Six Sigma. The reason we found it to be compelling for our purpose is that it allows us to put on a common playing field, if you will, approach to quantifying the risk or the priority associated with risk from a defect existing in the system. So, whereas your use of the system may be affected greatly by one particular kind of defect, and my use of the system be affected equally strongly by the existence of quite a different defect, there has been some challenge at an enterprise level to come up with a fair comparison or a way of comparing with respect to things that matter at the system level. Often conversations about how to prioritize the defect that you have and the defect that I have as a priority, those have often resulted in people advocating. And, it is the strength of advocacy that kind of helps break the tie.

With the approach to using risk priority number, we have created an opportunity for a common language to pervade. That common language is driven by the severity of the consequence should the defect exist in operation, the likelihood of and the frequency of its occurrence in operational use, as well as the difficulty in detecting whether or not the defect is causing a disruption to the function of the system. Those three metrics—severity, occurrence, and detection—are the fundamental elements of risk priority number, and it comes from failure modes and effects analysis that I described earlier where the origin was to diagnosis performance of a process.

What we have been able to do is to apply that concept to prioritizing and comparing defects in a way that gives us a quantum of numbers to work with. So, if we remove these three defects, we will reduce the total sum of RPN by a certain amount, and we can contrast that with removing five alternative defects. Does that remove a larger quantum of risk or priority of risk to the system? Then obviously we could look at the cost of doing the three versus the cost of doing the five and see what the tradeoff is.

That has been the benefit that organizations we have tried this with have seen in that it really structures that conversation for them. It is not a magic number that you would elevate above other priorities like cost and the capability that is enabled in a particular subsystem by resolving all defects in that subsystem. Those priorities may still exist, but in the context of those, if you

have a difficult choice to make, we help organizations come out of the strength of advocacy as the way to make the decision and give them a more elaborated basis for the conversation.

Suzanne: So, my guess is that when you do this, you come up with what I will call some counterintuitive results. You might go in thinking that, *This is the obvious candidate. Why doesn't everybody understand that this is the obvious candidate for removing this set of defects?* But, when you actually apply this index, you actually come out with a different result. Have you seen that in action?

Julie: Yes, we have. It is not always counterintuitive, but sometimes it is. What we found was that even some of the users who had said, *Oh, this is really important.* When we actually make them think about, *What is the severity? Are you really not going to be able to do your mission at all? Might you have to delay your mission? Might it just have a small impact on the mission?*

And, then when we ask, *Well, how often does this occur?*

And, they say, *Well, it occurs every time we do this, and that's about….*

How often is that?

Well, that is about once a month. And, you can see the wheels turning. *Well, it's only once a month, and it might delay the mission.*

Can you actually tell when the issue?

Oh, yes. Every time it happens, we know.

And, all of the sudden, they realize that, *So, maybe this isn't such a big deal after all. It only happens once a month. We know when it happens. We can fix it in a half an hour.*

Suzanne: Yes, but back in the '80s, we used operational profiles, which is a software reliability kind of construct, which is essentially, *How frequently is it used?* But, we didn't have the severity aspect. So, you had some things that only occurred if there was a big error or degradation in the hardware. So that is very severe because it would actually send the component offline. But, that wasn't figured in. So, we would have problems sometimes when these very, very low frequency events that had very high severity, they were kind of pushed off to the side. So, I can see where this would actually keep that from happening so that you get both the severity and the frequency of use included in that discussion.

Julie: And the users, the government program office and also the users were involved in helping to develop and then tweak the scales: *How important is severity? How important is occurrence? How important is detection?*

We actually broke down severity into several other categories to make it easier for the users to grade exactly what had gone wrong with the system and what the impact of that was.

Suzanne: Those categories could be set up differently for different types of systems, I'm assuming.

Will: One of the really interesting lessons along the way was when we actually started assigning values to individual defects, we started to hear that the workarounds they had for defects that affect the operational mission had an effect on their perception of the severity. So if a workaround relied on a particular type of staff person who was available very infrequently, then the risk associated with the defect persisting would elevate. If there is a help desk to support the workaround process, even though that process might be more involved, the availability of the help desk helped alleviate some of the priority and the risk posed by that defect. So there are things that just reading about risk priority number, you wouldn't think of. That interaction with the operational users and being in context really made a big difference for this.

Suzanne: So if you were to consult with an organization on using this after you have done this a few times now, what kinds of advice would you give them if they wanted to adopt risk priority number as one of their ways of addressing system defect prioritization?

Julie: I think the first thing is that, as Will stated, it's not a magic number. It won't answer all of your…

Suzanne: Once again, no silver bullet.

Julie: No, it won't answer all of your problems about how to prioritize which defects to fix first. The other thing I would let them know is that it will take a little bit of time and effort on the part of the program office, on the part of the users, to develop the rating scales.

Suzanne: Their scale.

Julie: Their own scale and how to best weight things and prioritize things so that it meets their system need. And then, even once you get that done, and you can compute a number, how to use that number in relationship to actually scheduling software development that will then address those defects is another very important concept of the whole way that you would implement RPN.

Suzanne: Defect management.

Julie: Right. So if you have different modules of code to say, *These are the top 10 defects we have, but they are all in a separate module of code. Let's open up all 10 modules and fix these 10. Then, when we have enough money, we will do the next 10, which are in those same 10*

modules again, and we're going to open up all those modules again. That may not be the most effective way to do it. So, there is still some thought to do once you get the number about how best to actually schedule these defects to be fixed.

Suzanne: The thing that I am hearing is there may be other factors that—depending on the complexity of the system and things like how the architecture is set up—you may have another factor in there in addition to those three, which may be not just difficulty to detect, but sort of effect on the system just by touching the system, right? If I have a brittle legacy system, I definitely want to treat that differently than something that is very loosely coupled and modern architecture, service orientation kind-of-thing. There may be other factors that you could bring into this index if that was really turning out to be appropriate.

Will: I think one of the most important things that we saw in the particular use we are thinking of is the process we went through to establish the credible way of computing this involved really establishing trust with a diverse user community. Because these tend to be very intelligent people, if you give them a system of numbers that are going to drive decisions, any one of them could easily find a way to work that system to their favor. What we needed to do, and I think we did very well, was to get them all to be in the same lifeboat with us, if you will allow that analogy. People understood that these things have a particular meaning and to just max out the scale because I want high numbers for my defects really works against the purpose that we are trying to pursue here.

Julie: Also, usually, in most government organizations, there is some boarding process for determining what the ranking of a DR [Defect Report] is and whether you accept a user's priority or not. Those types of things. One thing this helps with is it helps take that from a purely, *Well, it has a lot of impact on me and you need to believe me* to *Well, it's going to make me not be able to do a mission*. Then somebody can say, *Well, how does it stop the mission?*

Well, it stops the mission because I can't do x, y, and z when it happens.

Also, it becomes very clear how often it happens, right? So, if you see something that's happening every day that is going to stop somebody from doing a mission, it becomes much clearer to the other people sitting at the table that, *Yes, that probably is something high priority* rather than this user who claims that everything they own is high priority, saying, *Oh, but I have to have this one*. Well, now there is a way to discuss that such that you are discussing all 10 that are before the board using the same language and it's not one user saying, *Oh, this is really important*, and the next user saying…

Suzanne: It is no longer force of personality.

Will: If you think about the role of a change control board, it is their obligation to protect the integrity of the decision process as well as the future of the system. So if the conversation that happens on the change control board can be more robust, can reflect…

Suzanne: And more objective.

Will: Yes, a broader understanding of the system. It really is progress in other ways as well.

Suzanne: OK, I like this stuff. They know I like this stuff and I'm really glad to see us applying some of these techniques in our field work and I look forward to what you guys come up with with other uses. You have shared with us before, and Will and Julie have also recently presented a webinar on this work, the risk priority number. To view that webinar, please visit the SEI Digital Library at http://resources.sei.cmu.edu/library/. In the search field, type *risk priority number* to view all related resources on this topic.

Julie: Just one more thing before we go. I think we really do need to mention the name of our colleague who helped us do this. Bob Ferguson was instrumental in helping us develop this, the risk priority number.

Suzanne: Another one of our podcast interviewees from time to time.

This podcast is available on the SEI website at sei.cmu.edu/podcasts and on Carnegie Mellon University's iTunes U site. As always, if you have any questions, please don't hesitate to email us at info@sei.cmu.edu. Thank you.