

## Report Documentation Page

*Form Approved  
OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE <b>JUN 2008</b>	2. REPORT TYPE <b>N/A</b>	3. DATES COVERED <b>-</b>	
4. TITLE AND SUBTITLE <b>Quality Processes Yield Quality Products</b>		5a. CONTRACT NUMBER	
		5b. GRANT NUMBER	
		5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)		5d. PROJECT NUMBER	
		5e. TASK NUMBER	
		5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>MTC Technologies DTRA/RD-NTE 8725 JJ Kingman RD Ft Belvoir, VA 22060-6201</b>		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)	
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release, distribution unlimited</b>			
13. SUPPLEMENTARY NOTES <b>CROSSTALK The Journal of Defense Software Engineering June 2008</b>			
14. ABSTRACT			
15. SUBJECT TERMS			
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>	<b>SAR</b>
			18. NUMBER OF PAGES <b>3</b>
			19a. NAME OF RESPONSIBLE PERSON

# Quality Processes Yield Quality Products

Thomas D. Neff  
MTC Technologies

*Would your company like to save \$100,000 per day? Would you like to surge an urgent project's delivery time by 50 percent and deliver zero errors? Software organizations have done just that. In this article, I list small steps you can take that will lead your company toward similar results based on my 15 years of process improvement experience.*

If your company developed software that ran tools capable of propelling big objects long distances, measured accuracy in miles, and increased its accuracy to inches, you might save your customer millions of dollars. This actually happened [1]. If your company refined its software development processes so that your unit testing department found zero errors in a three-year period, you might eliminate unit testing and move those testers into other types of testing, saving many dollars with every release. This also happened [2]. If your customer asked you to speed up your next software delivery by 50 percent and *guarantee no flaws* in the delivered product, could you do it without incurring any extra costs or sacrificing other projects? One company did [3].

You may figure those goals are impossible for your organization to achieve or you do not have enough money to make it happen. If so, you are wrong. Right now open a Web browser, type <www.sei.cmu.edu>, and hit enter. If your organization does software development, search for Capability Maturity Model Integration for Development (CMMI-DEV). If you are an acquisition organization, search for CMMI-Acquisition (ACQ). All organizations should check out People Capability Maturity Model (P-CMM).

These models are all instantiations of Total Quality Management (TQM), the method that turned low-quality Japanese trinkets into high-quality automobiles, electronics, cameras, and many other products [4]. Because these models are different views of the same paradigm, you can also use them in other areas. One company used a predecessor of the CMMI-DEV and achieved the model's highest level of process maturity in software development. That company's hardware people realized the software folks really had their act together. They got jealous and sought the software secret. When they were shown the CMM, they said, "We could use that if we just change a few of the terms. Instead of talking about managing software requirements, we would

manage hardware requirements." Before long, that entire group was achieving record low manufacturing defects, record high profits, record high customer and employee satisfaction, record low employee turnover, and many more positive effects [3].

If the rewards from doing this are so great, why do so few companies achieve CMMI Level 5 – the highest level of process maturity? I contend it is because they do not execute their continuous process improvement (CPI) effort properly. There are many ways to do it right, but even more ways to do it wrong. If you would like to help ensure success in your CPI effort, read this article and get started. Before long you could very well be producing (or acquiring) software of exceptional quality, precisely meeting customer requirements, and incurring minimal maintenance costs.

Based on 15 years of CPI experience, here are some items you might consider when starting or reinvigorating your CPI effort. While they are no guarantee that you will reach the CMMI pinnacle, they can help you avoid pitfalls that snag many such efforts. (Throughout this list, *we* and *our* refer to the Nuclear Weapons Effects Division Process Improvement Team [PIT]):

- **Do not try to inspect in quality.** All too often, people believe they can have ad-hoc development processes, then use an inspection process *at the end* and effectively remove all defects, yielding a quality product. It just will not happen. My experience shows that only a small portion of defects are actually removed if the attempt is only at the end. Inspections in every phase of the process are good, just do not wait to the end and then do a lone inspection! Industry statistics indicate that for every four errors pulled out, one new error is injected. Hence, you must iterate many times to approach zero. Large expense, little return – not a good business decision.
- **Do not look for a quick fix.** I have learned to fear when a senior manager

goes to a conference where process improvement is discussed. Inevitably they return with the latest fad and want to implement it by week's end. It takes between two and three years to get CPI institutionalized. Your processes did not get screwed up overnight; they will not get fixed overnight, either.

- **Hold people accountable.** This is the biggest key to any CPI effort. If you create a meager CPI plan complete with a feedback loop for improvements, then hold people accountable to following it – you will make great progress in relatively little time. I have experienced both sides of this and can vouch that *not* holding them accountable will guarantee failure, and always holding them accountable is more likely to guarantee success. However, you cannot hold them accountable for six months and then give up because it is not working. Refer to the second point above.
- **Do not aim for a certain level of improvement.** Never state, "We want to achieve CMMI Level 3 by \_\_\_ date." What matters are the qualities exhibited, not the score obtained. Your primary emphasis must be to institutionalize CPI. Once that is accomplished, the rest will fall into place. If your aim is Level 3, once you reach it, you will not have any objective left and you will begin backsliding. However, if you emphasize CPI, once you reach Level 5, you will be thinking about what Level 6 (if there was one) would look like or you will seek other company areas that could benefit from your CPI attitude. Levels are just indicators of your progress.
- **Do not follow the CMMIs in the order they are written.** They are written so that *one size fits all*. As you and I know, even though one size fits all, it rarely looks good. You are much better off finding those areas of the CMMI currently giving you the most headaches and work on those first. If that does not work for you or you have many headaches, take a new project

and use it to pilot the CMMI. As you work through that project, write the necessary standard operating instructions (SOI)/standard operating procedures (SOPs), as identified by the CMMI, and test them with that project. Once they are acceptable, publish them as an example of how your organization does business. Of course, these are living documents and as you mature, your processes must evolve with you.

- **Perfect is the enemy of good enough.** If you are looking to produce perfect processes, you will never get there. Aim for the 80 percent solution. While that might seem pretty low, remember that each process has a feedback loop whereby improvements can easily and frequently be made. I know of no one who has ever gone to work thinking, "I want to do worse today than yesterday." Most employees want to do a better job. The problem is that they do not always know how, but processes can give them a framework. Their experience and intuition will help fill in the details on how to improve.
- **Jealousy is a great thing.** Do not let the lack of senior management support stop you. All you need is *any* manager to support CPI to get it going. Once you are making progress, others will see something is different with your manager: projects are being produced on time, on budget, and/or with greater quality. The other managers will become jealous and want to achieve the same success.
- **Do not try to end world hunger.** Aim low and reach your target. If you try to fix your whole company, you will likely spend most of your time negotiating, selling, and/or compromising. It is better to fix your little niche and make others jealous (see above). Allow them to modify your processes to fit their needs. They will already have incentive to ensure they are successful (jealousy still reigns). If they fail, they will come back since they will become jealous over something else you are doing better than them (and reaping tangible rewards such as a bigger budget or additional people). They might even stumble onto a better process than yours – great! Ask to use it and make it work for you. Now you have a strong ally.
- **Two heads are better than one.** Create a PIT encompassing each work unit in your area. As a PIT leader, you do not have the corner on good ideas. The more people you include (up to

seven, plus or minus two), the better. However, you do not want just anyone. You want people who share your enthusiasm for process improvement and who see the big picture. If you have the wrong team members, it can be detrimental because you will spend 80 percent of your time educating 20 percent of them.

- **Every team needs cheerleaders.** If you document/improve all processes but no one knows about them, you have accomplished nothing. Find the opinion leaders in each work area and get them involved. If they are not on the PIT, try to include them on the occasional work group or have them write an article for the PIT newsletter. That newsletter can be another good

---

**“One company used a predecessor of the CMMI-DEV and achieved the model’s highest level of process maturity in software development. That company’s hardware people realized the software folks had their act together ... When they were shown the CMM, they said, “We could use that ....”**

---

cheerleader. It is your first opportunity to provide training snippets on new processes as well as keeping everyone informed of your current CPI status.

- **Fail and get over it.** As humans we are imperfect. Do not worry about failure. The only failure is one where you learn nothing. If you are unsuccessful and learn from it then it was a great learning experience, not a failure because you now know at least one way *not* to do it.
- **Start with the obvious.** The CMMI-ACQ has a lot of information about what your acquisition program should

contain. It does not tell you *how* to do it but there is plenty of *what*. Do not wait until you have the *how* to get started. Take the *what* (i.e., CMMI) and turn it into a policy statement (SOI). Then, when it is time to create the *how* (SOP), people will know which *how* to develop. You will have already added some structure to your process improvement effort.

- **Keep focused.** When writing an SOI, do not delve into *how* people should do something. You want to focus on *what* they are to do and, on occasion, *why*. You can even describe a little of *who* or *when*. Once an SOP is created then it is time to describe *how* to do the job. These SOIs/SOPs should not be written for a three-year-old, but they also should not be written for a brain surgeon (unless it is an SOP on brain surgery). You should rarely include any *why* material in an SOP. If the worker does not know *why* they are doing their job, they have bigger issues. SOPs should be written by those already doing the job.
- **We don’t need no stinkin’ tools.** Just as everyone wants instant gratification, we also want a super tool to make our jobs easier, thus solving all of our problems. Come back to reality. That tool does not exist. I have found that if you buy a tool to solve your problems, you are more likely to get a failed CPI effort – and be poorer to boot. Because you do not have a repeatable process, the tool only lets you make mistakes faster, easier, and with greater impact. Of course, this frustrates people and they will quit using the tool. They will not realize it was the lack of process that caused the problems, not the tool. First, create a process and *then* introduce a tool to help people perform the process faster, cheaper, and better.
- **Sometimes status quo is good.** Remember, people want to do their jobs better – they just do not want to change to do that. The mere act of documenting your current (probably flawed) processes is a huge improvement over undocumented processes. At least now you *could* repeat the process twice in a row. It is better to get the early buy-in than try to perfect the process too quickly. There will be plenty of opportunity to improve the process as people use it.
- **Sometimes status quo is bad.** Hopefully you will never hear *we do it this way because it is how we have always done it*. However, someone is thinking

it. My experience is that if people do not know why they are doing something, they are also ripe for the suggestion that there might be a better way, especially if it means less work. Many of the *always done it this way* processes can be reduced in effort by 50 percent or more. Often, some work products are used by no one. If a product has no customer (user), eliminate it. You will earn many new friends. If there was a hidden customer, they will eventually figure out something changed and come to you to explain why they need what you eliminated. Then you will know why it is needed.

- **Keep it short.** We keep SOIs to no more than three pages. Most are one to one-and-a-half pages, with the shortest being two paragraphs. SOPs are longer, but we still try to keep them to about four pages. If attachments are added, we do not count those against the four-page goal. A short document will get read, but a long one will not. Our plan is to write 100 short documents instead of one all-encompassing volume.
- **Do not get hung up on training.** Some people feel they need training on everything. At some level, I agree. However, it is just as bad to do too much training as not enough. No one *needs* training on our SOIs. Even most SOPs are written so that anyone sufficiently educated could pick up an SOP and determine how to perform its task. Use screen captures, pictures, and flowcharts – some people like words, some need pictures. Cater to both but keep it short, and provide training as needed or requested.
- **A hyperlink is your friend.** Ample hyperlinking avoids redundancy and inaccuracy. For instance, we have an SOI describing acronyms and definitions. All acronyms and definitions used in our SOIs/SOPs are included here. We then name the definition as a bookmark and hyperlink upon its first use in each document. That way we ensure the proper definition is used and we do not have to spell it out, which keeps our documents shorter.
- **Procrastination is your enemy.** There is no bad place to start a CPI effort except to not start at all. I do not know how many times I have been asked how to start a CPI effort. I answer, “It doesn’t matter.” Start where you feel most comfortable, with what causes the most headaches, with what will give the best return on investment, or you can use any other criteria. As Nike said, *just do it*.

- **I think I can, I think I can.** The Little Engine That Could ran uphill for a long time. It was about out of steam when it crested the hill and things became easier. So it is with CPI. You *will* face an uphill battle for at least six months – and probably more. However, at some point (that point will be different for each organization) you will crest the hill and gain momentum. At that point, no one can stop your CPI effort. It will be institutionalized and no longer dependent on individuals, becoming an integral part of your organization’s business practices. As long as you have steam, you *must* keep chugging uphill. Set your sights just over the crest and you will get there.
- **This is not three-card monte. Pick a model, any model.** There are many process models from which to choose (CMM, CMMI, International Organization for Standardization [ISO] 9000, TQM, Lean, Six Sigma, Lean Six Sigma, etc.). Which should you use? When you are getting started, it does not matter. Just pick one and go. Any improvement is better than none. You may even choose bits and pieces of several models. Having said that, I believe the CMM and CMMI models are the most comprehensive and take you farther than the others. For instance, ISO 9000 takes you to about a CMMI Level 2. The Lean and Six Sigma models require you to document your process first, so you can determine just how much it has improved. Since most organizations just starting CPI do not have documented processes, it seems the CMM/CMMI might be best for starting because they provide guidance on what should be in your key processes. As your processes mature, you will likely incorporate other models into your CPI effort to speed your progress or improve the quality. Use whatever works for you.
- **Do not reinvent the wheel.** *Reuse* is your friend. Build on others’ successes. Learn from them. Never embrace *not invented here* syndrome. The Software Engineering Institute already developed all the tools you need to start making significant leaps in the quality of your processes. Their CMM and CMMI models describe every characteristic your organization should exhibit at various levels of process maturity. Use the models – they work. They will give you quality processes leading to quality products. From what I have seen, most failed

CPI efforts could not figure out where to begin, lost steam before starting, could not get any management support (usually tried at too high a level), focused too much on tools versus processes, could not find a quick fix and quit, or tried to solve world hunger and gave up.

Based on my 15 years in process improvement, I suspect that if you follow these suggestions, sticking with it at least two years, you will be successful in your CPI effort.

If you have CPI lessons learned, I would enjoy hearing them. ♦

## References

1. Yamamura, George, and Gary B. Wigle. “SEI CMM Level 5: For the Right Reasons,” *CROSSTALK* Aug. 1997 <[www.stsc.hill.af.mil/crosstalk/frames.asp?uri=1997/08/seicmm5.asp](http://www.stsc.hill.af.mil/crosstalk/frames.asp?uri=1997/08/seicmm5.asp)>.
2. Billings, C., J. Clifton, B. Kolkhorst, E. Lee, and W.B. Wingert. “Journey to a Mature Software Process.” *IBM Systems Journal*. Vol. 33, No. 1. 1994: pp. 46-61.
3. Vu, John D. “Presentation to CIO’s Office.” *National Reconnaissance Office*. Chantilly, VA., Mar 2001.
4. Deming, W. Edwards. *Out of the Crisis*. MIT Press. 1986.

## About the Author



**Thomas D. Neff, Lt Col, U.S. Air Force (Ret.)** spent most of his Air Force career in software development and project management.

Currently, he works for MTC Technologies supporting the Defense Threat Reduction Agency’s Nuclear Weapons Effects Division as a process manager, and uses the CMMI-ACQ and P-CMM as guides for that effort. Neff is a frequent speaker on process improvement at information technology conferences. He has a Master of Computer Science from Texas A&M, which helped steer him toward process improvement.

**DTRA/RD-NTE**

**8725 JJ Kingman RD**

**Ft Belvoir, VA 22060-6201**

**Phone: (703) 767-4106**

**Fax: (703) 767-9844**

**E-mail: [thomas.neff\\_contractor](mailto:thomas.neff_contractor@dtra.mil)**

**@dtra.mil**