

# Does anyone still believe in the Capability Maturity Model for Software?

Asked 16 years, 3 months ago   Modified 9 years, 9 months ago

Viewed 5k times



22



Ten years ago when I first encountered the [CMM for software](#) I was, I suppose like many, struck by how accurately it seemed to describe the chaotic "level one" state of software development in many businesses, particularly with its reference to reliance on heroes. It also seemed to provide realistic guidance for an organisation to progress up the levels improving their processes.

But while it seemed to provide a good model and realistic guidance for improvement, I never really witnessed an adherence to CMM having a significant positive impact on any organisation I have worked for, or with. I know of one large software consultancy that claims CMM level 5 - the highest level - when I can see first hand that their processes are as chaotic, and the quality of their software products as varied, as other, non-CMM businesses.

So I'm wondering, has anyone seen a real, tangible benefit from adherence to process improvement according to CMM?

And if you have seen improvement, do you think that the improvement was specifically attributable to CMM, or would an alternative approach (such as [six-sigma](#)) have been equally or more beneficial?

Does anyone still believe?

As an aside, for those who haven't yet seen it, check out this funny-because-its-true [parody](#)

process-management

cmmi

Share Follow

edited Sep 13, 2012 at 9:05

asked Sep 15, 2008 at 18:12



Ed Guinness

35.1k ● 16 ● 113 ● 149

---

A company I worked for long ago had a particular division get up to CMM level 5 so that they could advertise (or at least dangle that carrot on sales calls) that their organization was CMM level 5. Not exactly the most honest way to proceed, but don't be surprised if that is what you see. – [torial](#) Oct 27, 2008 at 20:50

---

- 1 I would also add that having worked with IBM, which is process heavy, that the highly talented people there seemed afraid to do anything lest they be buried under paperwork. Not sure what scheme they try... but process bloat can be dangerous! – [torial](#) Oct 27, 2008 at 20:51
-

## 22 Answers

Sorted by:

Highest score (default)



At the heart of the matter lies this problem, neatly described by the CMM guidance itself...

15



*"...Sound judgment is necessary to use the CMM correctly and with insight. Intelligence, experience and knowledge must shape an appropriate interpretation of the CMM in a specific environment. That interpretation should be based on the business needs and objectives of the organization and the projects. A rote, checklist-oriented application of the CMM has the potential to harm an organization rather than help it..."*



From Page 14, section 1.6 of *The Capability Maturity Model, Guidelines for Improving the Software Process* by the Carnegie Mellon University Software Engineering Institute, ISBN 0-201-54664-7.

Share Follow

edited Oct 16, 2008 at 13:58

answered Sep 15, 2008 at 18:25



**Ed Guinness**

35.1k ● 16 ● 113 ● 149



13

I found it to be bloated, documentation exercise that was used mainly as a contract-acquiring/maintaining vehicle. Once we had the contract, then it was an exercise in getting around the process.



As a developer, I got nothing out of it but lost MONTHS of my professional life fiddle-farting around with CMMI.



The same goes for 6 Sigma, which I branded "Common Sense in a Box". I didn't need to be trained how to figure out what the problem was to a process - it was generally quite evident.

For me, small teams and agile mechanisms work much better. Short cycles, lots of communication. That might not work in all environments, but it definitely works in mine.

Just my two cents.

Share Follow

answered Sep 15, 2008 at 18:31



[itsmatt](#)

31.4k ● 11 ● 102 ● 165



10

For a typical CMM level 1 programming shop, making the effort to get to level 2 is worthwhile; this means that you need to think about your processes and write them down. Naturally, this will meet resistance from cowboy programmers who feel limited by standards, documentation, and test cases.





The effort to get from level 2 ("there is a process") to level 3 ("everyone has the same process") normally gets bogged down in inter-departmental warfare, so it's probably not worth starting.

Share Follow

answered Sep 15, 2008 at 19:29



[Sten Vesterli](#)

3,033 ● 1 ● 23 ● 22

---

1 From experience I would agree. A pity then that CMM goes to level 5. – [Ed Guinness](#) Sep 26, 2008 at 11:10

---

I agree. The journey from level 2 to level 3 can be pursued but not in exuriating details. You might not get certified, but you will definately improve the processes. – [Vaibhav Garg](#) May 26, 2009 at 8:55

---

CMM level 5 is easier than level 3. With 3 people expect you to do the work, at 5 they expect you to just bribe the auditor. – [Jonathan Allen](#) Jan 27, 2015 at 0:36

---



If you see CMM run. And run fast.

10

Share Follow

answered Sep 15, 2008 at 18:27



[Jeroen Dirks](#)

7,847 ● 12 ● 56 ● 71





7



CMM and CMMI both offer some benefits if your organization takes the lessons it tries to teach at heart. The problem is that getting to the higher levels is very difficult and expensive, and the only time I have seen an organization go through the effort is because their customers won't let them bid on contracts until they are at a certain level.

This has the effect of the organization doing everything they can to "just get the number" without actually caring about it improving their process.

Share Follow

answered Sep 15, 2008 at 18:18



[Craig H](#)

7,979 ● 16 ● 51 ● 61



5



The higher end? No. CMM-5 shops do not impress me.

The lower end? Yes. CMM-1 organizations scare me.

CMM can help a new/novice team measure themselves and do the self improvement thing.

Share Follow

answered Sep 15, 2008 at 18:21



[Stu Thompson](#)

38.9k ● 19 ● 111 ● 156

- 1 The problem is that level 1 can not tell the chaotic, make-it-up-as-I-go mentality from the world class chessmaster; who has a strategy, and skill, he just develops it as he goes and doesn't write it all down. Level 1 doesn't mean ad-hoc; it

means management doesn't know what's going on. The assumption that level 1 is ad-hoc builds in invalid assumptions in the model from the start. – [Matthew Heusser](#) Feb 4, 2010 at 19:31

---

- 2 @Matthew Heusser: Level 1 is, **by definition**, *ad hoc*. Processes, if any, are undocumented and in a state of constant change. Additionally, you '*chess master*' analogy doesn't hold--software development teams are made out of many people working in concert, not just one lone wolf. Synergy is important. – [Stu Thompson](#) Feb 4, 2010 at 21:24
- 

Allright then. Consider the writer's room in Saturday Night Live. Unquestionably doing knowledge work as a team. Could be chaos, could be high-skill strategy that isn't documented. And isn't all good swdev in a state of constant change? – [Matthew Heusser](#) Feb 11, 2010 at 1:39 ✎

---

@Matthew Heusser: Thoughts on your SNL analogy: **1)** What happens in the writer's room is just part of the process that goes into making an SNL episode/season, which is a closer analogy to what we do in SW dev. **2)** Break down what they do there, and I'm sure you'll see a repeatable process. E.g.: a) what is worth mocking this week? b) brain storm ideas! c) divide into groups, work on the ideas, d) script dialog, e) push to production, legal, etc. by 4pm deadline. **Again**, CMM 1 is by definition *ad hoc*. Think on this: working in a reactionary manor (L1) vs. with intent (>L1) – [Stu Thompson](#) Feb 11, 2010 at 9:17

---



4



CMMI isn't really about improving your software, it is about documenting what you have done. You can almost estimate a company's CMMI level by the weight of the documentation it produces.



Background: I have studied CMMI in my Software Engineering graduate program and have worked on a team that followed its guidelines.

Share Follow

answered Sep 15, 2008 at 18:20



**Jonathan Allen**

70.1k ● 77 ● 269 ● 455



4



My experience is that the CMM is so vague that its very easy to fulfill. Also, when they come to certify you, they look at the project that your organization chooses. Where I used to work, this was the project with no real deadline, plenty of money, and lots of time to spend on every nook and cranny of process. Many of the other projects continued with little to no code/design review sometimes without versioning software.

I think the emphasis on CMM certification is unfortunate. Companies know how to work the system, and do. Instead of focussing on real process improvement that meets their bottom line, they focus on getting a certification and working the system. I honestly think most organizations would rather spend time on the former instead of wasting so much time on the latter.

Really what matters is having conscientious people who want to make good development decisions and know that they will need help making those decisions. There is no substitute for quality programmers who know that programming is an ongoing group activity where they are just as likely to make a mistake as anyone else.



Share Follow

answered Sep 15, 2008 at 18:25



[Doug T.](#)

65.5k ● 28 ● 141 ● 205



4

I have been doing a lot of interviewing for small teams doing iterative development. Personally, if I see CMM on a resume it is a big red flag that signals interest in process over results.



Share Follow

answered Sep 15, 2008 at 18:39



[dp.](#)

41 ● 1



3

All formal methods exist to sell books/training courses/certification, and for no other reason. That's why there are so many formal methods. Once you realise this, you are free :-)



Share Follow

[edited May 24, 2010 at 8:47](#)



answered Sep 15, 2008 at 19:31



[Bob Moore](#)

6,894 ● 3 ● 32 ● 44



Yourdon still [believes](#). But he might also still believe the world is going to end with Y2K.

2



This is not something I would personally put a lot of faith in or want to be yoked with in the future. But often ours is not to reason why...



Share Follow



answered Sep 15, 2008 at 18:20



Flory

2,849 ● 20 ● 31

---

the pdf is not found at the given link – [Rafay](#) Oct 15, 2011 at 15:16

---



2

P.S. Though a bit off-topic, I would like to mention that faked CMMI certifications are very common as well as real certifications obtained through bribery.



Share Follow

answered Sep 15, 2008 at 18:22



Jonathan Allen

70.1k ● 77 ● 269 ● 455



2

CMM doesn't really speak to the quality of the software, but more towards the documentation and repeatability of the process. In other words, it is possible to have an orderly and repeatable development process, but still create crappy software. As long as the process is properly documented, it is possible to achieve CMM Level 5.



At the end of the day CMM is another tool that can be used or misused. If the end goal is to improve software quality, it is possible to use CMM to improve the development process and improve software quality. If achieving a certain CMM level is the goal, then most likely software quality will suffer.

Share Follow

answered Sep 15, 2008 at 18:28



user7607



2



The Model is losing it's credibility, first because the companies adopt the model not looking for a maturer software development model, but to be appraised to a CCMI level.

And the other problem, the one that I think leads to the lost credibility is that as a contractor, you have no guarantee that the project your CMMI appraisal supplier is selling you will be developed using the model practices. The CMMi label only states that the company have once developed projects that were evaluated as adherents to a specific CMMi Maturity level.

The problem is not just on CMMi but on the process developed by the companies. The CMMi does not describe the process itself, but just what the process should do. You have the same problem with PMBOK. Actually the problem is not just the PMBOK, but primarily the problem is the bad project managers that claim to follow the PMI statements.

Share Follow

edited Oct 16, 2008 at 13:55



Ed Guinness

35.1k ● 16 ● 113 ● 149

answered Sep 16, 2008 at 6:23



Pedro

2,310 ● 1 ● 18 ● 22



1



At school, I was taught: CMM is a good Idea, but lacking certification (anyone can say they are level 5 / level 4) it ends up being a marketing tool for offshore shops. So, yeah, the idea is sound, but how do you prove adherence?



Share Follow



answered Sep 15, 2008 at 18:18



Daren Thomas

70.2k ● 42 ● 155 ● 205



1



I used to. But now I find that CMM and CMMI don't really fit that well with agile approaches.

Oh sure you can squeeze things to get that square peg into the round hole, but when push comes to shove, you are still basing your approach on an ability to predict everything that is needed, and anticipating everything that will be encountered, when building a software system.



And we all know, how well that approach works in real life! (-:

cheers,

Rob

Share Follow

answered Sep 15, 2008 at 18:22



[Rob Wells](#)

37k ● 13 ● 84 ● 147



1



Agile is the next CMM and both are fragile. The field of process and quality consulting is a good business in any industry and like the engineering folks everyone needs new buzzwords to keep the money flowing.

CMM when it first came out of the SEI was a good concept based on solid academic work but it was soon picked up by the process consultants and is a worthless certification now, which is used by most CIOs to cover their ass (Nobody got fired for picking a CMM Level 5 company)

Agile is going to go down that route soon and then we can be sure to see the next silver bullet in the horizon soon :)

Share Follow

answered Sep 15, 2008 at 18:25



[Sijin](#)

4,550 ● 23 ● 22



When I worked on commercial flight software, we used CMM and as our processes improved our ability to

1

accurately predict completion times improved. But this was a cumbersome process, other approaches should work just as well.



Share Follow



answered Sep 15, 2008 at 18:28



Tom Lendway



1

Smaller projects are less dependant on process for success. The key metric is the Hero to Bystander Ratio. Any project with an HTBR of less than 0.2 is in serious trouble.



Share Follow



answered Sep 15, 2008 at 19:21



Jess Kiddern



1

There are quite a few good ideas that can readily be adapted and adopted by any organisation for their own good, but getting a badge is a pain due to the requirement for all kinds of redundant documentation.



The problem is that CMMi is not a process but just a guide for whatever process you might choose to have and that in itself invites half-baked ideas flowing around.



Another point is that migration is a real pain when you are starting, but its the same as any other teething trouble, I guess.



The main issue with understanding the value of CMMi is understanding CMMi itself.

1



CMMi is a documented approach to Continuous Improvement for Software Production.



Understanding Continuous Improvement with SPC is difficult enough in manufacturing but add the intangible Software product and the difficulty is exponential.

I would recommend to anyone, or organization, new to CMMi: to document their current process then look at what outcomes (cost/benefit) can be measured independently of the process. In this way if any process, procedure or standard was changed would it yield a 'better' result. The prerequisite to this exercise is a documented, stable repeatable process since it is impossible to measure the benefit of any change within an ad-hoc environment as you are not comparing 'like for like'.

By focusing on the above concepts initially, the organization will begin to understand and embrace the essential value of the CMMi.





1



Legend has it that the US Department of Defense, which did a lot of contracting, found that many of its projects faced time and cost overruns, and even when they were delivered, the projects were not exactly what was ordered.

So they wanted a way to be sure that that a contractor would be able to deliver on time, within budget and close to what was required. Thus the capability maturity model was born.

The thesis is that if things are written down, then they survive attrition. But saying that write down everything would not be enough, it must be checked that they are written down correctly. Among other things.

Throughout all this, it never crossed their minds to consider the cost of doing all this. Because from the point of view of the DoD, if it gave out a project for \$ 1 million to get something in a year, and ended up paying \$ 10 million over 10 years and not getting what they wanted, and now if they instead had to pay \$ 5 million for that same thing to get what they actually wanted in two years, they are still saving \$ 5 million, and not to mention that they are actually getting something.

So if you are contractor to US DoD or something like that, go ahead and get CMM, because it would be a requirement. But if you are competing with the 1000s of software development shops on elance, to get projects



with limited budgets, limited time and so on... CMM is not a good choice.

That said, feel free to read the CMMI Dev pdf (v 1.3 at time of writing). It makes a lot of good points. It deconstructs the organisation very nicely. And if you see any points which make you go 'aha! i have this problem', then by all means use that wisdom to resolve your problem. In our case, one small change we made was to ensure that we make a list of all the people who are allowed to give us requirements. If there was more than one person who was allowed to give us requirements, then any requirement coming from one source was circulated to the others, and they had to say 'okay' before we added it to the backlog. This small change made a big difference in how much we worked and reworked.

In short look at the process areas and compare them to your pain areas, and take the suggestions given by CMM. The way you implement it is your own. And you can always implement it in a way that does not take too much time or cost too much money. But I guess the same applies even to the relevant ISO/IEC standards.

Share Follow

answered Feb 27, 2015 at 13:05



Kinjal Dixit

7,925 ● 2 ● 62 ● 68