

Capability Maturity Model Integration (CMMI)

Kurzfassung zum Tutorial

Ralf Kneuper

DB Systems, Kleyerstr. 27, 60326 Frankfurt

<http://www.kneuper.de>; {ralf.kneuper@gmx.de}

Abstract: Das im Januar 2002 erschienene CMMI ist Nachfolger dieses CMM und soll es bis Ende 2003 ablösen. CMMI wurde ebenfalls am Software Engineering Institute (SEI) der Carnegie-Mellon Universität in Pittsburgh im Auftrag des amerikanischen Verteidigungsministeriums entwickelt. In dieses neue CMMI sind viele Erfahrungen und Verbesserungsvorschläge zum CMM eingeflossen. Die Anforderungen sind einheitlicher strukturiert und der Anwendungsbereich ist über die Softwareentwicklung hinaus erweitert worden. Außerdem wurde zusätzlich zur stufenförmigen Einteilung mit fünf Reifegraden eine kontinuierliche Darstellung eingeführt.

1 Warum CMM?

Bei der Entwicklung von Software bzw. von Systemen aus Hard- und Software hat fast jede Organisation Schwierigkeiten, mit der Zeit, dem Budget und der zugesagten Qualität fertig zu werden. Die Ursachen dafür sind vielfältig. Derartige Probleme bei der Vergabe hoch komplexer Software haben das US-Verteidigungsministerium um 1986 dazu veranlasst, einen Lösungsansatz zu suchen, um bei der Vergabe einigermaßen sicher zu sein, dass der Auftragnehmer auch wie versprochen liefern kann. 1991 entstand das *Capability Maturity Model* CMM. Es besteht aus fünf Stufen, die den Reifegrad (*maturity level*) einer Organisation und ihrer Prozesse beschreiben (s. Tabelle). Die Stufen liefern Kriterien für die Beurteilung der Kompetenz einer Organisation, einen Auftrag auch erfolgreich durchzuführen.

2 Warum CMMI?

Ziel des 1997 gestarteten CMMI-Projektes war es, einen einheitlichen Rahmen für eine Reihe verwandter Reifegradmodelle zu schaffen und die Modelle so zu integrieren (daher das „I“ in CMMI), dass sie gemeinsam anwendbar sind. Außerdem sollten Erfahrungen und Verbesserungsideen zu den Vorgängermodellen eingearbeitet werden. Neben der neuen Unterscheidung in eine stufenförmige und eine kontinuierliche Darstellung gibt es auch je nach Anwendungsbereich Varianten, u.a. Softwareentwicklung, Systementwicklung oder Kauf von Software. Im Tutorial ging es vor allem um eine dieser Varianten, nämlich das *CMMI für die Softwareentwicklung* in seiner stufenförmigen Ausprägung.

Die Integration wird u.a. dadurch erreicht, dass gemeinsame Inhalte der verschiedenen Varianten des CMMI nur einmal beschrieben und dann an vielen Stellen genutzt werden. Dies gilt einerseits auf der Ebene der *Prozessgebiete*, andererseits auf der Ebene darunter durch Definition von generischen Zielen und Praktiken. Prozessgebiete fassen jeweils alle Anforderungen zu einem Thema zusammen, z.B. zur Projektplanung, organisationsweitem Training oder Ursachenanalyse und Problemlösung. Die folgende Tabelle gibt einen Überblick:

	Reifegrad	Prozessgebiet (Process Area, PA)	Abk.
5	Optimierend	<i>Organisational Innovation and Deployment</i> <i>Causal Analysis and Resolution</i>	OID CAR
4	Quantitativ gemanagt	<i>Organisational Process Performance</i> <i>Quantitative Project Management</i>	OPP QPM
3	Definiert	<i>Requirements Development</i> <i>Technical Solution</i> <i>Product Integration</i> <i>Verification</i> <i>Validation</i> <i>Organisational Process Focus</i> <i>Organisational Process Definition</i> <i>Organisational Training</i> <i>Integrated Project Management</i> <i>Risk Management</i> <i>Decision Analysis and Resolution</i>	RD TS PI VER VAL OPF OPD OT IPM RSKM DAR
2	Gemanagt	<i>Requirements Management</i> <i>Project Planning</i> <i>Project Monitoring and Control</i> <i>Supplier Agreement Management</i> <i>Measurement and Analysis</i> <i>Process and Product Quality Assurance</i> <i>Configuration Management</i>	RM PP PMC SAM MA PPQA CM
1	Initial		

Jedes Prozessgebiet umfasst eine Reihe von Zielen, die dabei erreicht werden sollen:

- *Spezifische Ziele*. Sie gelten nur für das jeweilige Prozessgebiet.
- *Generische Ziele*. Sie beschreiben die Institutionalisierung des Prozessgebietes, also all das, was zu tun ist, damit die spezifischen Ziele regelmäßig, dauerhaft und effizient umgesetzt werden. Ziele werden als *generisch* bezeichnet, weil sie übergreifend für die verschiedenen Prozessgebiete gelten. Die verschiedenen generischen Ziele beschreiben, wie intensiv das jeweilige Prozessgebiet institutionalisiert wird und haben damit eine enge Beziehung zum Reife- bzw. Fähigkeitsgrad.

Literatur

- [Kn02] Kneuper, R.: CMMI - Verbesserung von Softwareprozessen mit Capability Maturity Model Integration. dpunkt, Heidelberg, November 2002.