

## CMM – običan mit, čarobna formula ili – nešto između

*Veliki outsourcing ugovori se ne zaslužuju inovativnim proizvodima, veličinom firme ili brojnim referencama. Poverenje klijenata koji žele da razvoj svog proizvoda prenesu offshore ili nearshore se stiče dokazano zrelim, implementiranim i merljivim procesima, koji garantuju kvalitet softvera, jeftiniju produkciju i isporuku na vreme. Upravo iz tog razloga, CMM je danas jedan od temelja poslovanja ogromnog broja preduzeća iz zemalja - outsourcing šampiona, kao što je Indija.*

CMM (*Capability Maturity Model*) je model za ocenjivanje zrelosti procesa razvoja softvera i identifikaciju ključnih aktivnosti koje je mogu povećati, kroz evolutivni pristup. Razvijen je na *Software Engineering Institute* (SEI) *Carnegie Melon Univerziteta* (<http://www.sei.cmu.edu>) u Pitsburgu, 1991. godine. Istraživanje koje je dovelo do razvoja CMM modela je finansirano od strane američke administracije i imalo je za cilj analizu rada izvođača koji su razvijali softver za potrebe Ministarstva odbrane (koje i danas finansira SEI, desetinama miliona dolara subvencija godišnje). Generalno, CMM predstavlja skup ciljeva koje jedna razvojna organizacija treba da postigne da bi poslovala brzo, efikasno i kvalitetno – model definiše šta je potrebno uraditi da bi se ostvarilo poboljšanje, a ne – kako. CMM nije čarobna formula za lak put do velikih *outsourcing* ugovora, već samo niz smernica za evoluciju zrelosti razvojne organizacije, kroz prepreke svih realnosti biznisa.

Sušтина CMM modela je opis evolucije razvojnih firmi od *ad-hoc* izvedenih projekata do zrelog – disciplinovanog okruženja u kojem se oni nesmetano i racionalno odvijaju. *Ad-hoc* razvojno okruženje podrazumeva improvizovano delovanje inženjera i menadžera u kome uspeh ili neuspeh projekta zavisi isključivo od njihove lične sposobnosti i upornosti. Očigledno, jedan od najvećih problema ovakvog stanja su nepredvidivi elementi razvoja – troškovi, bagovi, datumi, a razvijene komponente se najčešće ne mogu koristiti u druge svrhe (*reusability*). Uobičajeno, cenu haotičnog pristupa razvoju plaća klijent jer kupuje precenjeni softver, čija isporuka kasni, a čije je održavanje teško i skupo.

Zrelo razvojno okruženje sadrži sisteme kontrole zahtevanog kvaliteta i procese koji obezbeđuju efikasno upravljanje procesima i merenje njihovih parametara. Tačno predviđeni troškovi, rokovi i kvalitet predstavljaju realne parametre za donošenje odluka; kroz konzistentne i dokumentovane procese testiranja postiže se maksimalni kvalitet pre isporuke softvera, a kod se lako održava uz mnogo kraće cikluse reinžinjeringa; u svim projektima se ostvaruje konzistentan kvalitet a razvijene komponente se mogu više puta koristiti.

Generalizacijom CMM modela je nastalo više njegovih varijacija, namenjenih različitim disciplinama. Nakon razvoja prvobitnih platformi namenjenih razvoju softvera (SW-CMM) i njegovoj nabavci (*Software Aquisition CMM SA-CMM*), usledio je rad na prilagođavanju modela za *System Engineering* (SECM) upravljanje radnom snagom (*Workforce Management and Development*) i integrisani razvoj (IPD-CMM - *Integrated Product and Process Development*). Iako su ovi modeli uspešno primenjeni širom sveta, ispostavilo se da korišćenje dva ili više modela u jednoj organizaciji prouzrokuje određene probleme. S obzirom da implementacija više strukturno različitih poslovnih modela umanjuje sposobnost organizacije da se fokusira, 2000. godine SEI objavljuje CMMI (*CMM Integration*) smernice, koje imaju za cilj da integrišu SW-CMM, SECM i IPD-CMM.

### **Pet nivoa zrelosti**

CMM model opisuje 5 nivoa zrelosti procesa razvoja: 1) Inicijalni; 2) Ponovljivi; 3) Definisani; 4) Upravljeni; 5) Optimizovani. Inicijalni nivo zrelosti se karakteriše *ad-hoc* razvojem koji u nekim slučajevima može voditi ka haosu i CMM model se njime ne bavi.

Svaki nivo predstavlja jedno stanje evolucije odnosno sazrevanja cele organizacije – CMM ne može biti individualna inicijativa ili stanje zrelosti samo jednog ili više odeljenja firme u kojoj se vrši razvoj. Svaki

nivo se karakteriše **ključnim oblastima procesa** (KPA – *Key Process Area*) u okviru kojih se ostvaruju određeni **ciljevi**. Svaku od KPA čini grupa **ključnih aktivnosti** (*Key Practices*) – one predstavljaju infrastrukturu, aktivnosti, akcije ili uslove koji moraju biti ostvareni ili izvedeni da bi ciljevi, definisani ključnom oblasti procesa, mogli da budu ispunjeni. CMM, dalje, definiše grupe ključnih aktivnosti u okviru kojih se vrši postizanje ciljeva. Grupe ključnih aktivnosti su:

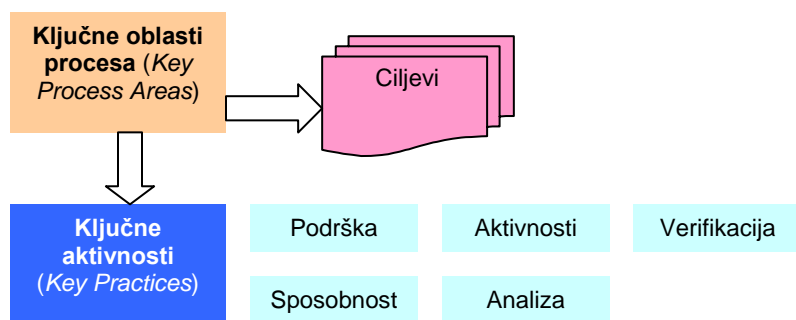
**Ostvarivanje široke korporativne podrške** (*Commitment to perform*). U okviru ove grupe treba da budu predstavljene aktivnosti organizacije koji predstavljaju deo podrške ostvarenju ciljeva definisanih KPA.

**Stvaranje potrebnih uslova za implementaciju** (*Ability to perform*). Obuka zaposlenih, nabavka potrebnih resursa.

**Izvršene aktivnosti** (*Activities performed*). Ključne aktivnosti direktno vezane za procese iz KPA.

**Analiza izvršenih aktivnosti** (*Measurement and analysis*). Merenja i analize uspeha izvršenih aktivnosti implementacije.

**Verifikacija implementiranih procesa** (*Verifying implementation*). Nezavisna verifikacija implementacije procesa.



## Nivo 2 – Ponovljivi procesi

Cilj drugog nivoa CMM modela je osiguranje uspeha svakog razvojnog projekta – **ponavljanje** prethodnih, uspešno izvedenih projekata i izbegavanje problema koji su ranije doveli do neuspeha. Najveći broj CMM sertifikovanih organizacija danas ima implementiran nivo ponovljivosti procesa.

Ključne oblasti procesa drugog CMM nivoa su:

**Upravljanje zahtevima** (*Requirements Management*). Proces definisanja zahteva i njihovih izmena u toku projekta ili nakon njega mora biti definisan. Kroz učešće klijenta, mora se obezbediti da njegova očekivanja budu ispunjena.

**Planiranje projekta** (*Software Project Planning*). Projekat se mora odvijati prema definisanom kalendaru u kome su za određene datume vezani tačno definisani ciljevi.

**Praćenje projekta** (*Software Project Tracking and Oversight*). Progres procesa mora biti stalno praćen, da bi se, u slučaju devijacija razvoj mogao prilagoditi novim okolnostima.

**Upravljanje eksternim procesima razvoja** (*Software Subcontract Management*). Podizvođači projekta se moraju birati primenom sistema kriterijuma, a relacije sa njima moraju biti upravljive. Analiza procesa sertifikacije pokazuje da je ovaj KPA prosečno na najnižem nivou implementacije u odnosu na ostale, karakteristične za nivo 2.

**Upravljanje kvalitetom** (*Software Quality Assurance*). Procesi validacije i *auditinga* obezbeđuju visoki kvalitet isporučenog softvera.

**Upravljanje životnim vekom proizvoda** (*Software Configuration Management*). Deo razvoja mora biti i proces za efikasno upravljanje i održavanje proizvoda kroz njegov životni vek.

### **Nivo 3 – Definisani procesi**

Ostvarenje CMM nivoa 3 podrazumeva da su procesi razvoja softvera i njegovog upravljanja su dokumentovani, standardizovani i integrisani u sistem procesa cele organizacije. U svim projektima se koristi razvijena (prilagođena specifičnim zahtevima projekta) verzija standardnih procesa za razvoj i održavanje softvera. Studija, urađena još 1994 (*Larry Putman*), pokazala je da se vreme razvoja softvera u organizaciji koja pređe put od nivoa 1 do nivoa 3, u proseku smanjuje za 70%.

Zrelost nivoa 3 podrazumeva ostvarene ciljeve definisane za nivo 2, ali i realizovane sledeće ključne oblasti procesa:

**Fokus na značaj procesa u organizaciji** (*Organization Process Focus*). U organizaciji se mora ostvariti svest o značaju procesa i njihovoj odgovornoj primeni za uspeh projekata.

**Definisanje procesa razvoja i upravljanja razvojem** (*Organization Process Definition*). Uspostavljanje i održavanje standardnih procesa razvoja i upravljanja razvojem.

**Program obuke** (*Training Program*). Primena procesa kojima se razvijaju veštine i znanja svih aktera u razvoju i njegovom upravljanju.

**Integrisano upravljanje razvojem** (*Integrated Software Management*). Sve aktivnosti proistekle iz ključnih oblasti karakterističnih za nivo 2 se moraju integrisati u definisane standarde procesa razvoja i upravljanja razvojem. Dokaz kompleksnosti ove KPA je i dokazana činjenica da je njene aktivnosti i karakteristične procese mnogo teže implementirati, u odnosu na ostale ključne oblasti procesa.

**Primena standarda za kodiranje** (*Software Product Engineering*). Uspostavljanje prakse razvoja koja će obezbediti konzistentan – jednoobrazan kod, uvođenjem standarda za kodiranje.

**Koordinacija timova i radnih grupa** (*Inter-Group Coordination*). Uspostavljanjem procesa strukturirane komunikacije obezbediće se laka integracija različitih modula, razvijenim u odvojenim grupama.

**Diskusija o projektu** (*Peer Reviews*). Periodični sastanci svih aktera projekta imaju za cilj rano uočavanje mogućih defekata i problema u razvoju, kao i poboljšanje komunikacije i razumevanja širom organizacije.

### **Nivo 4 – Upravljeni procesi**

Nivo 4 karakteriše sposobnost merenja širokog spektra parametara toka procesa i kvaliteta nastalih rezultata. Sami proizvodi, kao i proces njihovog razvoja moraju biti kvantitativno opisani, pri čemu svaku izmerenu veličinu mora da karakteriše potpuno razumevanje i upravljivost.

Ključne oblasti procesa u nivou 4 su:

**Kvantitativno upravljanje procesima** (*Quantitative Process Management*). Kvantitativno upravljanje procesima podrazumeva niz sredstava za merenje i kontrolu izmerenih veličina procesa razvoja. Izmerene veličine, kao pokazatelj kvaliteta se koriste za ranu identifikaciju problema u nesmetanom odvijanju razvoja.

**Upravljanje kvalitetom softvera** (*Software Quality Management*). Upravljanje kvalitetom softvera se vrši na osnovu izmerenih, egzaktnih veličina koje se referenciraju na kvantitativno opisane ciljeve – definisane elemente kvaliteta softvera.

## Nivo 5 – Optimizovani procesi

Iako se nivo optimizacije procesa veoma teško dostiže, većina organizacija sa implementiranim nivoom 4 dostiže i nivo 5 (u proseku, za 13 meseci od početka implementacije). Interesantno je da je većina organizacija sa ostvarenim nivoom 5, smeštena u meki *outsourcinga* – Indiji. Prema zvaničnim podacima SEI, u svetu postoji samo 207 firmi sa sertifikovanim nivoom 5.



Ključne oblasti procesa, karakteristične za nivo 5 su:

**Prevenција defekata i bagova** (*Defect Prevention*). Identifikacija uzroka defekata, problema i bagova u procesu razvoja i samom softveru i rad na prevenciji njihovog ponovnog pojavljivanja.

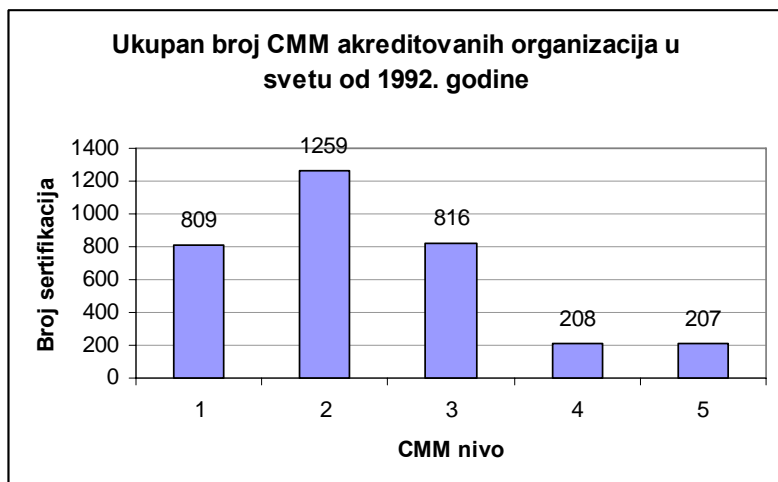
**Postepeno uvođenje novih tehnologija u organizaciji** (*Technology Change Management*).

**Poboljšanje postojećih procesa** (*Process Change Management*). Uspostavljanje procesa kontinualnog poboljšanja ključnih procesa, karakterističnih za sistem kvaliteta, produktivnost i efikasnost.

### **Proces verifikacije**

Prema podacima sa zvaničnog sajta, SEI partneri su, od 1992. godine do danas izvršili više od 3000 procena, na osnovu kojih su dobijeni podaci prikazani u dijagramu. Očigledno je da najveći broj organizacija ima implementiran nivo 2, odnosno ponovljive procese. Na osnovu analize procesa sertifikacije

u 2004. godini, utvrđeno je da je po prvi put od 1996. godine zabeležen pad broja izvršenih procena u odnosu na proteklu godinu.



Sertifikaciju može da izvrši samo akreditovani SEI partner – organizacija čija je stručna kvalifikacija priznata i verifikovana. U našem okruženju ne postoje SEI partneri – najbliža adresa na koju se neka od naših firmi može obratiti za pomoć u implementaciji CMM metodologije i sertifikaciju je Nemačka, u kojoj postoji 10 SEI partnera.

Da bi procena bila objektivna i referentna, sertifikaciona politika SEI je korišćenje određene, tačno definisane metodologije za evaluaciju. Svaka metodologija je izvedena iz zvaničnih smernica SEI za procenu – CAF (*CMM Appraisal Framework*). Da bi se objektivnost sačuvala, deo sertifikata predstavlja i informacija o tome koja je metodologija korišćena u proceni. Najnoviji metod za procenu implementiranih procesa razvoja softvera je CBA IPI (*CMM-Based Appraisal for Internal Process Improvement*), razvijen i testiran 1994. godine. SCE (*Software Capability Evaluation*) je metod za procenu implementiranih procesa u sektoru nabavke softvera, mada se može koristiti i za procenu drugih procesa. Pored ove, dve najreferentnije metode koje mere učinak CMM implementacije, koriste se još *Interim Profile* metod, koji nije izveden iz CAF smernica, a u novije vreme i SCAMPI (*Standard CMMI Assesment Method for Process Improvement*), razvijen 2000. godine, paralelno sa CMMI, zasnovan na istim principima kao CBA IPI.

Analiza izvršenih procena pokazuje da je najteže implementirati nivo ponovljivosti (nivo 2) i upravljivosti (nivo 4). Za implementaciju CMM nivoa 2 primenom neke od metodologije, u proseku je potrebno 20 meseci, dok je za postizanje nivoa 4 potrebno čak 25 meseci.

U velikim preduzećima čija osnovna delatnost nije razvoj softvera, uvedeni ISO 9001 sistem kvaliteta može da ograniči primenu CMM-a u njihovim razvojnim odeljenjima. Uporedna analiza odgovarajućih ISO 9001 smernica, relevantnih za razvoj softvera i CMM modela, izvršena u SEI, pokazuje da primena ISO sistema kvaliteta može da pomogne ostvarenju većine ciljeva CMM nivoa 2 i samo nekih sa nivoa 3. Jedan on neverovatnih zaključaka ove analize je da organizacija sa CMM nivoom 1 može da dobije ISO 9001 sertifikat ! Ukoliko se prisetite da razvoj na nivou 1 SEI opisuje atributima kao što je – “haotičan“, rizikujete da ostanete potpuno zbunjeni. Sa druge strane, ciljevi implementirani CMM nivoom 3 su u kontradiktornosti sa ISO smernicama – organizacije sa ovim nivoom mogu imati mnogo problema u dobijanju ISO 9001 sertifikata !

### **„Daj taj CMM, pa da radimo...”**

Iako će vam sertifikat pomoći da dobijete posao, CMM nije čarobna formula koja će, sama po sebi, povećati produktivnost razvoja i poboljšati kvalitet proizvedenog softvera. Osnovni moto alibija CMM gurua za

eventualni neuspeh CMM-compliant projekta je – CMM predstavlja skup ciljeva, a ne metoda za njihovo ostvarenje. Upravo, iz tog razloga, uvođenje i primena CMM modela razvoja podrazumeva mnogo više kreativnosti i fleksibilnosti od suve discipline, kako se to u javnosti pogrešno shvata. Pored nesporazuma koje može izazvati ovakvo nerazumevanje CMM metoda, on sadrži još veoma ozbiljnih mana, čije pravovremeno previđanje može izazvati velike probleme u produkciji.

Autori CMM modela tvrde da se uzroci većine grešaka u produkciji ne nalaze u tehničkim barijerama koje se savlađuju u razvoju, već upravo u – upravljanju projektom. Iako se sa ovim većina slaže, CMM jednostavno ide predaleko u ovom pravcu. Programeri mogu pisati veoma loš kod, neoptimizovan i nefaktorisan, a da i dalje, bez ikakvih problema, organizaciji u kojoj rade bude verifikovan visok CMM nivo. Kvalitet koda je jedan od osnovnih ciljeva razvoja softvera, ali CMM ne poseduje mehanizme koji će pomoći njegovo osiguranje.

Dalje, CMM ne možete primeniti da biste izvadili vruće kestenje iz vatre. Ukoliko imate problem sa projektom na kome trenutno radite, nemojte razmatrati CMM kao moguće rešenje. CMM se može posmatrati kao niz preventivnih mera za izbegavanje problema u produkciji, ali ne nudi rešenja za poboljšanje tekućih projekata ili recikliranje starih.

Konačno, CMM se zasniva na ponovnom korišćenju postojećih rezultata završenih ili projekata u toku, u iste ili slične svrhe u narednim. Naravno, ovo ima smisla samo ukoliko su projekti sličnog karaktera, sa sličnim proizvodima. Ukoliko otvarate projekat sa tehnologijama koje su vam do sada bile nepoznate, rigidna primena CMM-a, ne samo da ne mora da vas ubrza, već može čak i povećati rizik od problema u izvođenju projekta.