

Procesní standardy

Radek Mařík

CA CZ, s.r.o.

September 14, 2007



1 Vyspělost procesu

- Koncept
- CMMI model

2 Hodnocení softwarového procesu

- SEI a SPR hodnocení
- Malcolm Baldrige hodnocení
- ISO 9000, IEEE/ANSI

Vyspělost procesu, Standardy kvality ^[Kan95]

- který proces je použit,
- stupeň, jak se používá, se liší:
 - od organizace k organizaci,
 - od projektu k projektu,
- implementační procedury,
- metody a nástroje, které se používají,
- metriky a měření,
- co je dobrý stav vs. jak se do stavu dostat.

Metodologie používané k hodnocení vyspělosti procesu organizace nebo projektu:

- The SEI Capability Maturity Model Integrated (CMMI),
- The Software Productivity Research (SPR) process maturity assessment methods,
- The Malcom Baldrige discipline and assessment processes,
- ISO 9000 registrační proces.

Nezralé vs. Vyspělé softwarové organizace ^[PCCW93]

- Bez porozumění rozdílu mezi vyspělou a nezralou softwarovou organizací nelze vybrat přiměřené cíle k zlepšení softwarového procesu.
- Základním problémem je neschopnost řídit softwarový proces.
- V nedisciplinovaných organizacích závisí úspěch na heroických výkonech týmu.
- Zopakování výsledků závisí cele na přítomnosti těch samých individuálů na následujícím projektu.
- Nezralé organizace jsou reakční a manažéři se soustředí na řešení okamžitých krizí (hašení požáru).
- Kvality produktu je obtížně předpovídat.

Základní koncepty ^[PCCW93]

- **Proces** (IEEE-STD-610) je sekvence kroků vykonaných za daným účelem.
- **Softwarový proces** může být definován jako množina aktivit, metod, praktik a transformací, které lidé používají k vývoji a údržbě softwaru a asociovaných produktů (např. plánů projektů, návrhových dokumentů, kódu, testovacích případů a uživatelských manuálů).
- **Schopnost softwarového procesu** popisuje rozsah očekávaných výsledků, které mohou být dosaženy, pokud se dodržuje daný softwarový proces.
- **Výkonnost softwarového procesu** prezentuje dosažené aktuální výsledky, pokud se dodržuje daný softwarový proces.
- **Vyspělost softwarového procesu** je míra, se kterou je daný proces explicitně definován, organizován, měřen, řízen a jak je efektivní.
- **Vyspělost** znamená potenciál růstu schopnosti a indikuje jak bohatost softwarového procesu organizace tak konzistenci, se kterou je proces aplikován v projektech organizace.



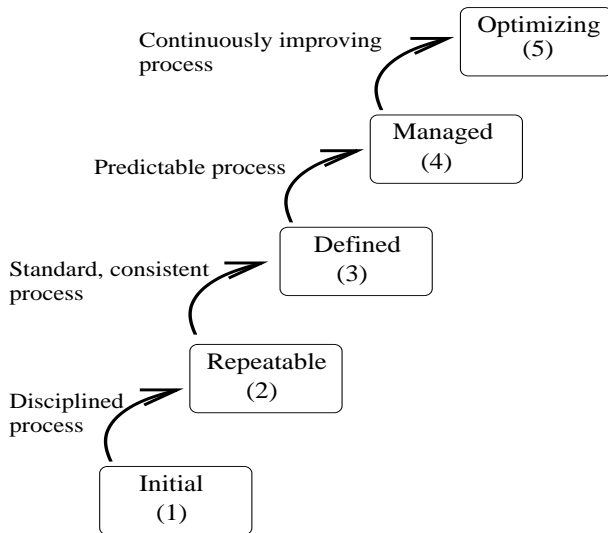
CMMI:

- organizační strategie zlepšení,
- jak navrhnout vývojovou cestu,
- průvodce k výběru strategie zlepšení procesu,
- poskytuje rámec organizace vývojových kroků do 5 úrovní výspělosti.

Úroveň výspělosti:

- je jasně definovaný stupeň dosažený výspělosti,
- každá úroveň výspělosti poskytuje vývojovou základnu pro kontinuální zlepšování procesu.
- každá úroveň je definována množinou cílů, které při splnění stabilizují jistou důležitou komponentu procesu.

5 úrovní vyspělosti softwarového procesu [PCCW93]



Charakteristiky 5 úrovní vyspělosti ^[PCCW93]

1 Počáteční:

- ad hoc, někdy až chaotický softwarový proces,
- některé procesy mohou být definovány,
- úspěch závisí na úsilí jednotlivců.

2 Opakovatelný:

- organizační procesy sledování nákladů, časového rozvrhu a funkčnosti,
- procesní disciplína k zopakování úspěchu na podobných aplikacích.

3 Definovaný:

- organizační a inženýrské aktivity dokumentovány, standardizovány a integrovány jako standard organizace,
- odsouhlasenou přizpůsobenou verzi standardu na všech projektech.

4 Řízený:

- sběr podrobných měrových ukazatelů kvality procesu i produktu,
- kvantitativní porozumění procesu a produktu.

5 Optimalizující:

- neustálé zlepšování procesu,
- kvantitativní zpětná vazba z procesu,
- zavádění inovačních myšlenek a technologií.

Charakterizace chování na úrovni vyspělosti 1 ^[PCCW93]

- Není vytvořeno stabilní prostředí pro vývoj a údržbu softwaru.
- Během krizí projekty typicky opouští plány a vrhají se na pouhé kódování a testování.
- Úspěch závisí cele na vyjimečném manažérovi a sehraném efektivním softwarovém týmu.
- Nelze se spoléhat na odhady časových rozvrhů, nákladů, funkčnosti a kvality produktu.

Charakterizace chování na úrovni vyspělosti 2 ^[PCCW93]

- Efektivní proces dovoluje zopakovat úspěšné praktiky minulých projektů.
- Jsou instalovány základní nástroje řízení softwaru.
- Požadavky na software a pracovní produkty jsou evidovány a je kontrolována jejich integrita.
- Jsou definovány standardy softwarového projektu.
- Plánování a trasování projektu je stabilní.
- Vytvářejí se reálné plány založené na zkušenostech z minulých projektů.

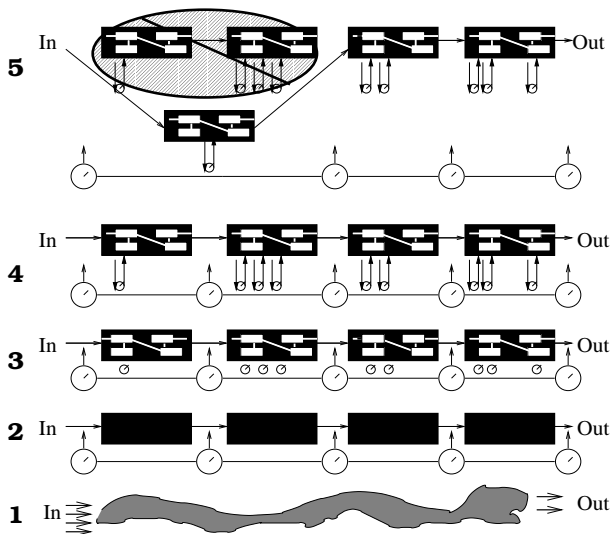
Charakterizace chování na úrovni vyspělosti 3 ^[PCCW93]

- Standardní proces vývoje a údržby softwaru je dokumentován, zahrnuje jak manažérské tak i inženýrské procesy, které spolu jsou koheretní.
- Standardní proces je základním procesem organizace.
- Existuje skupina odpovědná za softwarový proces.
- Projekty se odvíjejí od standardu s přihlédnutím na jejich unikátní charakteristiky.

Porozumění úrovním vyspělosti ^[PCCW93]

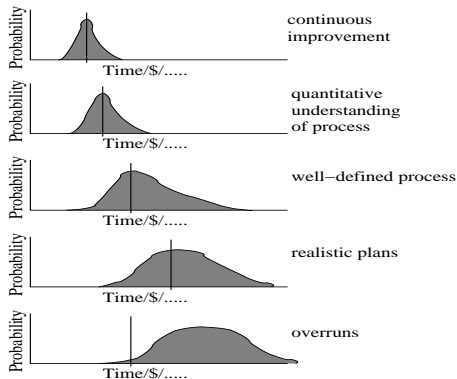
- Základní atributy - dostatečná úroveň abstrakce
- Interpretace praktik by měla být rozumná.
- CMMI nepředpisuje.
- CMMI nepředepisuje organizaci, jak se zlepšit.
- 1-2: několik let, ostatní pak dva roky.
- Úspěch na úrovni 1 závisí cele na heroických výkonech lidí.
- Úrovně Opakující a Definovaný posunují pozornost na organizační a manažerská témata.
- Úroveň 2: disciplinovaný softwarový proces, management se orientuje, dokumentované procedury.
- Úroveň 3: integrace na úrovni organizace.
- Úroveň 4 a 5: relativně neznámá teritoria, založeno na konceptech statistického řízení procesu.
- Úroveň 4: řízení procesu, obecně stabilní prostředí.
- Úroveň 5: neustálé zlepšování procesu.

Viditelnost do softwarového procesu [PCCW93]



Schopnost procesu [PCCW93]

- Čím je vyspělost vyšší, tím se zmenšuje rozdíl mezi cílenými a aktuálními výsledky.
- Variability aktuálních výsledků se zmenšuje okolo cílů.



Přeskakování úrovní [PCCW93]

- Organize může s výhodou použít procesy vyšších úrovní.
- Tyto procesy se nemohou plně rozvinout, pokud nemají řádný základ.
- Přeskočení úrovně často nepřinese hledané výhody, neboť každá úroveň tvoří základy pro následující úroveň.
- Organizace na úrovni 1 nemají úspěch se zavedením definovaného procesu, neboť manažéři jsou přetíženi rozvrhováním a tlaky cen. Toto je **základní** důvod, proč je nutné nejprve řešit manažérské problémy před technickými.
- Technici se často domnívají, že jsou schopni implementovat inženýrské procesy bez ohledu na manažérské procesy, ale končí pod tlakem plánování a nákladů.

Operační definice CMMI ^[PCCW93]

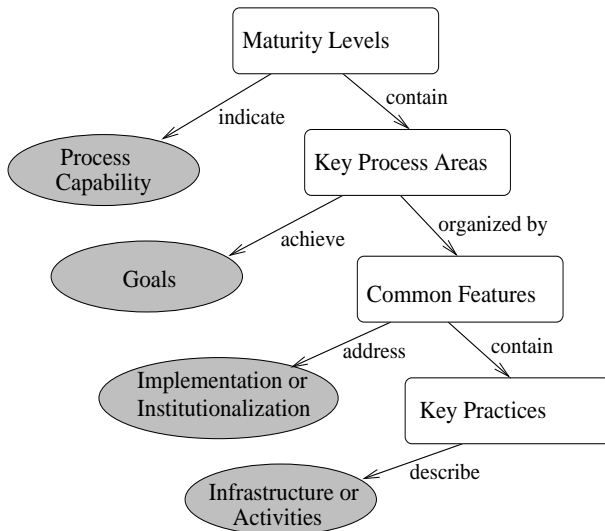
Použití CMMI:

- **Hodnotící týmy** - identifikace silných a slabých stránek organizace.
- **Výběrové týmy** - identifikace risků při výběru z řady kontraktorů.
- **Manažéři a technici** - porozumění aktivitám, které potřebují k plánování a implementaci procesu a programu jeho vylepšení.
- **Procesní skupina** - průvodce, který pomáhá definovat a zlepšit proces.

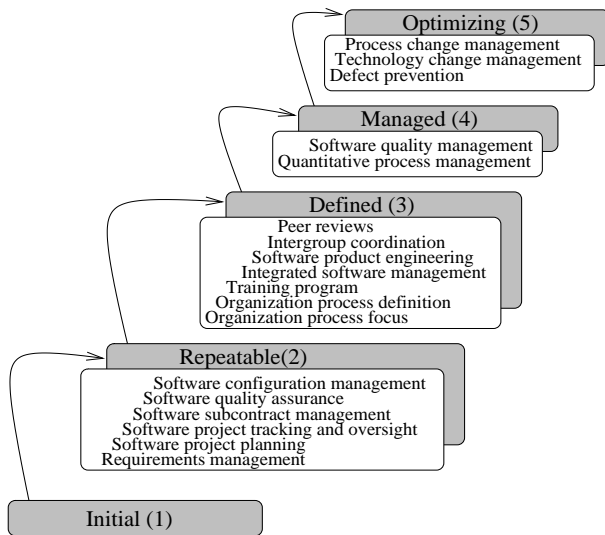
Dekompozice úrovní vyspělosti:

- abstraktní přehledy,
- **Klíčové procesní oblasti** - organizovány do 5 sekcí zvané obecné vlastnosti.
- **Obecné vlastnosti** - specifikují klíčové praktiky, které jako celek splňují cíle dané klíčové procesní oblasti.

Struktura CMMI [PCCW93]



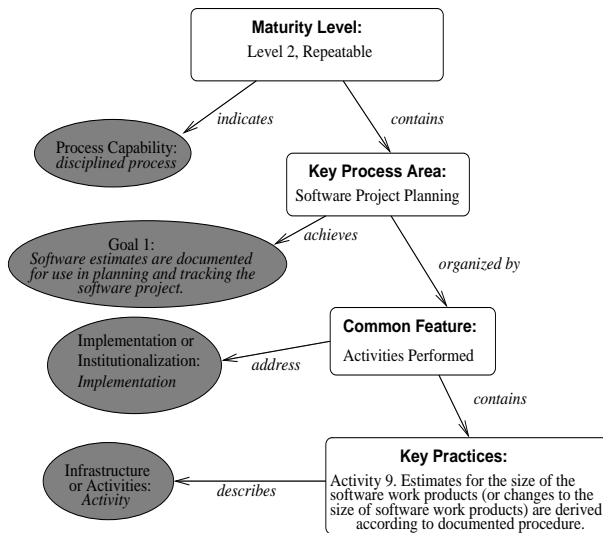
Klíčové procesní oblasti [PCCW93]



Klíčové procesní oblasti, obecné vlastnosti, klíčové praktiky ^[PCCW93]

- **Každá klíčová procesní oblast** - identifikuje sadu aktivit, které ,pokud jsou všechny prováděny, splňují množinu cílů důležitých k dosažení procesní schopnosti.
- Všechny cíle klíčové procesní oblastí musí být dosaženy k tomu, aby daná oblast byla splněna.
- “Klíčová” znamená, že existují oblasti, které nejsou klíčové k dosažení úrovně vyspělosti.
- **Obecné vlastnosti** - atributy, které indikují, zda implementace a ustanovení klíčové oblasti je efektivní, opakovatelné a trvajících.
- **Klíčové praktiky** - předepisují infrastrukturu a aktivity, které nejvíce přispívají k efektivní implementaci klíčové oblasti (co, ne jak).
- Každá klíčová praktika je jednoduchá věta, která je často doplněna podrobným popisem s příklady.

Příklad klíčové praktiky [PCCW93]



Klíčové procesní oblasti úrovně 2: Opakovatelné ^[PCCW93]

Řízení požadavků

- 1 Systémové požadavky na systém jsou řízeny tak, že vytvářejí evidovanou základnu určenou pro softwarové inženýrství a potřeby manažérů.
- 2 Plány na vývoj softwaru, produkty a aktivity jsou udržovány konsistentní ss systémovými požadavky na software.

Plánování softwarového projektu

- 1 Softwarové odhady jsou dokumentovány za účelem použití při plánování a traosvání softwaru.
- 2 Softwarové projektové aktivity a povinnosti jsou plánovány a dokumentovány.
- 3 Skupiny a jednotliví lidé souhlasí s tím, co mají udělat v rámci daného projektu.

Hodnocení SPR a SEI ^[Kan95]

- vyvinuto Jonesem v Software Productivity Research, Inc. (SPR),
- publikováno v 1986,
- velký stupeň podobnosti, ale i podstatné odlišnosti mezi metodami SEI a SPR,
- process ohodnocení

SEI:

- zaměřeno na strukturu softwarové organizace a softwarové procesy,
- 85 otázek, binární škála (ano/ne),

SPR:

- pokrývá jak strategické problémy korporace, tak i projektovou taktiku,
- okolo 400 otázek, pětibodová stupnice dle Likerta:
 - 1 výborný,
 - 2 dobrý,
 - 3 průměrný,
 - 4 dostatečný,
 - 5 nedostatečný

Okruhy SPR otázek ^[Kan95]

- měření kvality a produktivity,
- zkušenosti programátorů s odstraňováním defektů před testováním,
- zkušenosti programátorů s odstraňováním defektů při testování,
- cíle kvality a spolehlivosti na projektech,
- odstraňování defektů před testováním na projektové úrovni,
- odstraňování defektů při testování na projektové úrovni,
- odstraňování defektů po uvedení na trh.

Malcolm Baldrige hodnocení ^[Kan95]

- The Malcolm Baldrige National Quality Award (MBNQA),
- jedna z nejprestižnějších ocenění kvality ve Spojených státech,
- 1988, the U.S. Department of Commerce,
- udělováno ročně americkým společností, které vynikají v řízení kvality a stupni jejího dosažení,
- mnohem širší rozsah než SEI a SPR hodnocení (výroba, servis, drobné podnikání)
- přezkušovacího kritéria - sedm kategorií obsahující 28 položek:
 - schopnosti vedení,
 - informace a analýza,
 - strategické plánování kvality,
 - používání lidských zdrojů,
 - zabezpečení kvality produktů a servisu
 - výsledky kvality,
 - spokojenost zákazníků.
- 1992, The European Quality Award

ISO 9000 [Kan95]

- sada standardů a průvodců k řízení zajištění kvality, system
- International Organization for Standardization
- řada firem neprojde auditem napoprvé, od 60% do 70%.
- velmi silné požadavky na správu dokumentů,
 - musí být adekvátní účelu,
 - identifikace vlastníka,
 - řádně schválen před zveřejněním,
 - řízená distribuce,
 - identifikace verze,
 - očíslované stránky,
 - indikace celkového počtu stran,
 - zastaralé dokumenty jsou ihned likvidovány,
- ISO 9000 audit se zaměřuje na systém řízení kvality a řízení procesu.
- *“Řekni, co děláš, dělej, co říkáš, a dokaž to.”*

IEEE/ANSI Standardy softwarového procesu ^[Kit95]

- IEEE Standard Glossary of Software Engineering Terminology, IEEE Std 610.12-1990
- IEEE Standard Dictionary of Measures to Produce Reliable Software, IEEE Std 982.1-1988
- IEEE Guide for the Use of IEEE Standard Dictionary of Measures to Produce Reliable Software, IEEE Std 982.2-1988
- IEEE Standard for Developing Software Life Cycle Processes, IEEE Std 1074-1991
- IEEE Standard for Software Test Documentation, (Reaff.1991), IEEE Std 829-1983
- IEEE Standard for Software Verification and Validation Plans, (Reaff. 1992, IEEE Std 1012-1986
- IEEE Standard for Software Review and Audits, IEEE Std 1028-1988
- IEEE Standard for Software Quality Assurance Plans, IEEE Std 730-1989
- IEEE Standard for Software Unit Testing, IEEE Std 1008-1987

Literatura I



Stephen H. Kan.

Metrics and Models in Software Quality Engineering.
Addison-Wesley, 1995.



Edward Kit.

Software Testing in the Real World.
Addison-Wesley, 1995.



Mark C. Paulk, Bill Curtis, Mary Beth Chrissis, and Charles V. Weber.

Capability maturity model for software, version 1.1, cmu/sei-93-tr-024, esc-tr-93-117.
Technical report, Software Engineering Institute, February 1993.