

## 4IT421 Zlepšování procesů budování IS



Téma:

**CMMI-DEV v.1.3 PA Requirements Development**

Osnova:

1. Úvod .....	3
2. SEI .....	3
3. CMMI v.1.3 .....	4
4. CMMI-DEV v.1.3 PA Requirements Development .....	6
4.1. Requirements Development v souvislostech .....	6
4.2. Requirements Development .....	8
5. Konkrétní cíle a praktický přehled .....	11
5.1. Vývoj požadavků zákazníka .....	11
5.1.1. Elicit Needs .....	12
5.1.2. Transform Stakeholder Needs into Customer Requirements .....	13
5.2. Vývoj požadavků na produkt .....	13
5.2.1. Establish Product and Product Component Requirements .....	14
5.2.2. Allocate Product Component Requirements .....	15
5.2.3. Identify Interface Requirements .....	15
5.3. Analýza a ověření požadavku .....	15
5.3.1. Establish Operational Concepts and Scenarios .....	15
5.3.2. Establish a Definition of Required Functionality and Quality Attributes ..	16
5.3.3. Analyze Requirements .....	16
5.3.4. Analyze Requirements to Achieve Balance .....	17
5.3.5. Validate Requirements .....	17
6. Závěr .....	18
7. Zdroje .....	18

# 1.Úvod

Tato seminární práce se zabývá Integračním modelem zralosti a to jeho částí – procení oblastí - nazvanou Requirements Development. Autor si dal za cíl popsat nejen tuto jednu procesní oblast, ale popsat i základní souvislosti týkající se této procesní oblasti.

## 2.SEI

SEI neboli The Carnegie Mellon Software Engineering Institute je federální výzkumné a vývojové centrum v Pittsburghu. Další významné pobočky má SEI v Arlingtonu ve Virginii a ve Frankfurtu nad Mohanem v Německu. Tato společnost je z velké části financována Ministerstvem obrany USA. Dále spolupracují s akademickou obcí a jsou napojeni i na průmysl. Mezi hlavní oblasti výzkumu patří:

- Řízení procesů
- Rizika
- Zabezpečení
- Vývoj softwaru
- Návrh nových systémů

SEI definuje konkrétní podněty zaměřené na zlepšení schopnosti organizací v rámci softwarového inženýrství. SEI tak stojí například za CERT/CC (CERT Coordination Center), která se zabývá bezpečností, ATAM (Architecture Tradeoff Analysis Method ), který zvyšuje schopnost softwarových inženýrů analyzovat nebo předvídat v rámci strojírenského průmyslu, nebo stojí za publikování modelu CMMI v roce 2000.

### 3. CMMI v.1.3

CMMI v 1.3. byl vydán v listopad 2010. Jedná se o proces zlepšování modelu zralosti pro vývoj produktů a služeb. Obsahuje best practices, které se zaměřují na aktivity spojené s vývojem a údržbou, přičemž tyto aktivity pokrývají celý životní cyklus produktu od počátku až po jeho předání a údržbu.

CMMI poskytuje modely pro hodnocení úrovně procesního řízení v organizaci. Je nutné zvolit jednu z reprezentací, aby organizace dosahovala zlepšení procesů cestou, která je jí bližší. Jedná se o tyto dvě možné reprezentace:

#### 1) Stupňovitá

Umožňuje aplikovat systematické a strukturované postupy využití modelu pro zlepšení procesů pouze o jeden stupeň v daný okamžik. Díky těmto předepisovaným změnám jednotlivých procesních oblastí určuje i cestu zlepšení organizace ze základního levelu k optimalizovanému. Organizace označuje procesní oblasti podle jejich zralosti. Zlepšení je měřeno pomocí maturity levels.

#### 2) Kontinuální

Nabízí pro zdokonalení procesů maximální flexibilitu při použití modelu CMMI. Organizace se mohou rozhodnout o zlepšení jednoho procesu na základě problému, nebo mohou najednou pracovat s několika oblastmi, které úzce propojují obchodní cíle organizace. Organizace označuje procesní oblasti podle svých priorit. Zlepšení je měřeno pomocí capability levels. Celkem CMMI poskytuje 22 procesních oblastí a to nezávisle na reprezentaci. Tyto oblasti jsou rozděleny do několika kategorií:

- ▪ Proces management
- ▪ Projekt management
- ▪ Engineering
- ▪ Support

Každá procesní oblast má své specifické a generické cíle, které by měly být zajištěny specifickými a generickými praktikami.

Level	Continuous Representation Capability Levels	Staged Representation Maturity Levels
Úroveň	Kontinuální prezentace – úrovně způsobilosti	Stupňovitá reprezentace – úrovně zralosti
0	Incomplete (neúplný)	
1	Performed (vykonávaný)	Initial (úvodní)
2	Řízený ( managed )	Managed (řízená)
3	Definovaný ( defined )	Defined (definovaná)
4		Quantitatively managed (kvantitativně řízená)
5		Optimizing (optimalizovaná)

Tabulka 1 : Porovnání úrovní způsobilosti a zralosti

## 4.CMMI-DEV v.1.3 PA Requirements Development

### 4.1. *Requirements Development v souvislostech*

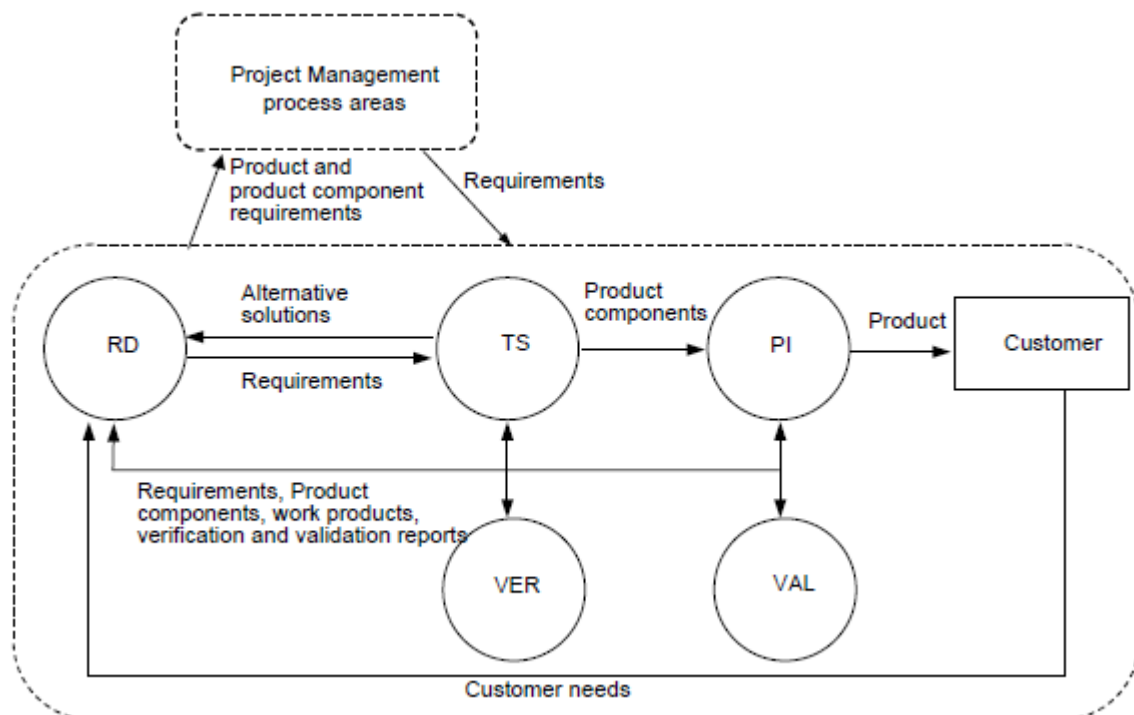
Requirements Development (dále jen RD) lze přeložit jako požadavky na rozvoj a je to jedna z dvaadvaceti procesních oblastí podle úrovně zralosti v CMMI 1.3. Patří do třetí úrovně zralosti a to do definovaných. Co je pro všechny tyto procesní oblasti charakteristické? Organizace již dosáhla všechny cíle úrovně zralosti 2 a je definován proces na úrovni organizace, který je popsán ve standardech, procedurách, nástrojích a metodách. Dále je i definováno i přizpůsobování procesů podle typu projektu. Následující obrázek zobrazuje sumář všech procesních oblastí a ukazuje na jaké úrovni zralosti se nachází. Šedé oblasti ve sloupcích úrovně způsobilosti (CL) představují cílový profil, který je ekvivalentní k úrovni zralosti (ML).

Ve třetí úrovni zralosti, kam patří i RD se vyskytují všechny 4 kategorie. RD patří do kategorie Engineering spolu s PI ( Product Integration – integrace produktu), TS ( Technical solution – technické řešení), VAL ( Validation – validace) a VER (Verification – ověření). Tato kategorie pokrývá činnosti vývoje a údržby v různých inženýrských disciplínách a týká se vývoje produktu nebo služby.

Name	Abbr.	ML	CL1	CL2	CL3
Configuration Management	CM	2	Target Profile 2		
Measurement and Analysis	MA	2			
Project Monitoring and Control	PMC	2			
Project Planning	PP	2			
Process and Product Quality Assurance	PPQA	2			
Requirements Management	REQM	2			
Supplier Agreement Management	SAM	2			
Decision Analysis and Resolution	DAR	3	Target Profile 3		
Integrated Project Management	IPM	3			
Organizational Process Definition	OPD	3			
Organizational Process Focus	OPF	3			
Organizational Training	OT	3			
Product Integration	PI	3			
Requirements Development	RD	3			
Risk Management	RSKM	3			
Technical Solution	TS	3			
Validation	VAL	3			
Verification	VER	3			
Organizational Process Performance	OPP	4	Target Profile 4		
Quantitative Project Management	QPM	4			
Causal Analysis and Resolution	CAR	5	Target Profile 5		
Organizational Performance Management	OPM	5			

Obrázek 1: Sumář procesních oblastí

Další obrázek poskytuje pohled z ptačí perspektivy z interakcí mezi 6 inženýrskými procesními oblastmi.



Obrázek 2: Engineering Process Areas

Obrázek 2 znázorňuje procesní oblast RD, která identifikuje potřeby zákazníků a překládá tyto požadavky do Product Requirementu (požadavek na výrobek). Celá sada požadavků na daný produkt je analyzována, aby výsledným řešením byla efektivní a koncepční produkce. Tato sada požadavků je pak určena k tomu, aby vytvořila dílčí vzorové požadavky na produkt. Ostatní požadavky, které pomáhají definovat produkt, jsou odvozeny a následně přiděleny ke komponentám produktu. Tento soubor požadavků na produkty a jejich komponenty jasně popisuje provedení produktu, požadavky na kvalitu, konstrukční prvky, způsob ověřování požadavků atd., z hlediska vnímání a potřeb vývojáře. Procesní oblast RD dodává požadavky na procesní oblast technického řešení (TS), kde jsou požadavky převedeny do architektury produktu, návrhy komponent a provedení komponenty produktu (např. kódování). Požadavky jsou také dodávány do procesní oblasti PI, kde jsou komponenty produktu kombinovány a jejich rozhraní je ověřeno, aby bylo zajištěno, že interface splňuje požadavky RD. Procesní oblast technického řešení se opírá o specifické postupy v oblasti ověřování (VER) procesů a návrhů a vzájemného hodnocení během návrhu před konečnou verzí. Procesní oblast validace (VAL) postupně ověřuje produkty z hlediska pro potřeb zákazníka. Validace může být provedena v reálném provozu i v simulovaném prostředí. Důležitým elementem celé validace je koordinace požadavků zákazníka.

V oblasti PM lze v některých projektech využít i rekurzi, protože některé projekty jsou vnořené v rámci jiných větších projektů.

## ***4.2. Requirements Development***

Co je tedy účelem RD? Účelem RD je tedy získat informace, analyzovat a zřídit požadavky zákazníků, produktů nebo komponent produktů.

Tato procesní oblast tedy popisuje tři typy požadavků:



- Požadavky zákazníků
- Požadavky na výrobky
- Požadavky na komponenty výrobků

Všechny tyto požadavky reagují na příslušné zúčastněné subjekty, včetně potřeb týkajících se různých fází životního cyklu (např. testovací kritéria) a vlastnosti produktu (např. citlivost, bezpečnost, spolehlivost nebo udržitelnost). Požadavky také řeší omezení způsobené výběrem designu. Všechny vývojové projekty mají nějaké požadavky a tyto požadavky jsou základem pro design. Vývoj požadavků zahrnuje následující činnosti:

- Zjištění informací, analýza, validace a komunikace se zákazníky a zjištění jejich potřeb, očekávání a omezení a nutí k získání prioritních požadavků, které představují pochopení toho, co uspokojí zainteresované strany
- Shromáždění a koordinace potřeb zúčastněných
- Životní cyklus vývoje požadavků na produkt
- Zavedení počátečního požadavku (v souladu s požadavky zákazníka) produktu a komponenty produktu
- Vytvoření správnost a kvality požadavku zákazníka

Tato procesní oblast řeší všechny požadavky zákazníků, protože zákazník může poskytnout konkrétní design požadavky. Proto jsou tyto požadavky důležitější než ty na produktové úrovni. Požadavky zákazníků jsou upřesněny v požadavcích produktu nebo komponenty produktu. Některé požadavky zákazníka, produktu nebo komponenty produktu jsou odvozené z vybraných designových řešení. Pod pojmem „produkt“ a „komponenta produktu“ je myšleno také služba a servisní systém a jejich komponenty. Požadavky jsou identifikovány a roztrženy během fází životního cyklu produktu.

Procesní oblast RD obsahuje 3 specifické cíle:

- Develop Customer Requirements – vývoj požadavků zákazníka
- Develop Product Requirements – vývoj požadavků produktu
- Analyze and Validate Requirements – analýza a ověření požadavku

Analýza slouží k pochopení, definování a vybrání požadavků na všech úrovních od konkurenčních alternativ. Tyto analýzy jsou následující:

- Analýza potřeb a požadavků pro každou fázi životního cyklu, včetně potřeb všech zúčastněných, provozního prostředí a faktorů, které odrážejí celkové zákaznicko očekávání jako například spolehlivost, bezpečnost nebo dostupnost
- Vývoj provozního konceptu
- Definice požadované funkcionality a kvality

Tyto analýzy objevují postupně a rekurzivně podrobnější detaily vrstev produktové architektury, dokud nejsou všechny informace podrobné a dostatečné. V důsledku analýzy požadavků a provozní koncepce (včetně funkčnosti, podpory nebo údržby) výroba nebo výrobní koncept produkuje více odvozených požadavků včetně těchto následujících:

- Omezení různých typů
- Technologická omezení
- Náklady a nákladové položky
- Časové omezení a harmonogramy

- Rizika
- Jedinečné obchodní důvody, předpisy nebo zákony zavedené projektantem

Hierarchie logických jednotek (např. funkce a podfunkce, objektů tříd a podtříd atd.) je založená na iteraci skrze vývoj provozní koncepce. Požadavky jsou propracované, získané a přidělené k těmto logickým jednotkám. Požadavky a logické jednotky jsou přiděleny k produktu, komponentě produktu, k člověku nebo k procesu. Účast všech zúčastněných stran na vývoji požadavků dává tak velký přehled nad samotným vývojem. Tato činnost se neustále ujišťuje, že požadavky jsou správně nadefinované.

## **5. Konkrétní cíle a praktický přehled**

### ***5.1. Vývoj požadavků zákazníka***

*Potřeby zainteresovaných subjektů, očekávání, omezení nebo podoba rozhraní je shromažďována a je sumarizována v rámci požadavků zákazníka.*

Potřeby všech zúčastněných stran (např. zákazníci, koncový uživatelé, dodavatelé, výrobci, testeři, zaměstnanci logistické podpory) jsou základem pro stanovení požadavků. Potřeby zúčastněných, očekávání, omezení, rozhraní a koncepce jsou analyzovány, harmonizovány, roztrženy a rozpracovány do srozumitelného balíčku zákaznických požadavků. Často se stává, že potřeby zúčastněných, očekávání, omezení, nebo rozhraní jsou špatně identifikovatelné, definované nebo až zcela konfliktní. Po celou dobu životního cyklu by tak mělo být vše definované, srozumitelné a dobře pochopitelné. Pro usnadnění požadované

interakce lze uživatele nebo zákazníka zastoupit kvůli pomoci s řešením střetů. V rámci vytváření sady požadavků by se mělo dbát na environmentální, právní nebo jiná omezení, která musí být zohledněna.

#### **5.1.1. *Elicit Needs***

*Získání potřeb zúčastněný, očekávání, rozhraní nebo omezení ve všech fázích životního cyklu.*

Jedná se o sběr dalších informací nebo identifikací, které nejsou vysloveně určeny přímo zákazníkem. Další požadavky by měli respektovat životní cyklus vývoje produktu.

- Příklad metod k získávání informací:
- Dotazníky, rozhovory, scénáře získané od koncových uživatelů
- Brainstorming
- Průzkumy trhu
- Beta testování
- Extrakce ze zdrojů (dokumenty, standardy, specifikace)
- Technologické ukázky
- Pozorování stávajících produktů
- Uživatelské zkušenosti
- Reverzní inženýrství
- Atd.

Příklady zdrojů požadavků, které nemohou být identifikovány zákazníkem:

- Obchodní podmínky
- Standardy
- Regulační stanovy
- Technologie

➤ Atd.

Hlavní myšlenkou tedy je využití metod k získání informací o potřebách zúčastněných, o omezeních, očekáváních nebo externích rozhraních.

#### **5.1.2. Transform Stakeholder Needs into Customer Requirements**

*Transformace potřeb zúčastněných, očekávání, omezení a rozhraní do prioritních požadavků zákazníka.*

Různé vstupy všech zúčastněných by měly být konsolidovány, chybějící informace by měly být doplněny a konfliktní místa opraveny podle požadavků zákazníka. Požadavky mají také svou prioritu. V některých situacích může zákazník poskytnout sadu požadavků, které vychází z předchozích projektů nebo činností. V těchto případech tak mohou být některé požadavky v rozporu a musí tak být transformovány do správné sady požadavků po vhodném vyřešení daného konfliktu.

Důležitou částí tedy je zřídit a udržovat priority zákaznických požadavků a udržovat kvalitu všech vlastností. Dále pak definovat omezení pro ověřování a schvalování.

### **5.2. Vývoj požadavků na produkt**

*Požadavky zákazníků jsou zpracovány a ovlivňují vývoj produktu nebo komponenty produktu.*

Požadavky zákazníků jsou analyzovány v souvislosti s vývojem operačního konceptu s cílem získat podrobnější a přesnější sady požadavků na produkt nebo komponentu produktu. Požadavky na produkt a komponenty produktu reagují na potřeby ve všech fázích životního cyklu. Všechny požadavky jsou znovu

přezkoumány a je upřednostněn koncept produktu. Požadavky jsou přidělovány k produktu a komponentě produktu včetně lidí, procesů nebo objektů. V případě iteračního nebo přírůstkového vývoje je požadavkům přidělena jistá míra priority. Návaznost na funkční požadavky, objekty, testy atd. je dokumentována. Požadavky a funkce jsou základem pro spojení technického řešení. Dodatečná rozhraní jsou definovány a zavedeny.

#### **5.2.1.            *Establish Product and Product Component Requirements***

*Vytváří a udržuje požadavky na produkt a komponenty produktu, které jsou založeny na požadavcích zákazníka.*

Požadavky na funkce a vlastnosti zákazníkem mohou být vyjádřeny v podmínkách a mohou být popsány netechnickým popisem. Tyto požadavky na produkt jsou vyjádřením technických požadavků, které mohou být použity rozhodnutí designu. Příkladem by mohlo být, když zákazník řekne, že chce pevné nehlukné dveře do bytu, znamená to určení požadavků na velikost dveří, hmotnost dveří, tlumení zvuku nebo rezonanční frekvence atd.

Základem tedy je vývoj technických požadavků nezbytných pro produkt nebo komponenty produktu. Odvozené požadavky pak vyplývají z návrhů na design. Dále rozvíjí požadavky na zachycení kritických míst.

Příklady kvality vlastností:

- Reagovat do 1 s
- Systém je k dispozici min 99% času

Dále pak udržuje vztahy mezi požadavky z hlediska průběhu řízení změn.

### **5.2.2.        *Allocate Product Component Requirements***

*Přidělování jednotlivých požadavků ke komponentám.*

Tato část odkazuje na procesní oblast technického řešení. Architektura přiděluje základní požadavky na komponenty.

### **5.2.3.        *Identify Interface Requirements***

Identifikace rozhraní požadavku

Tato část je opět popsány více v procesní oblasti technického řešení. Identifikuje vnitřní i vnější rozhraní produkt a dále identifikuje vývoj požadavků na rozhraní.

## **5.3.   *Analýza a ověření požadavku***

*Požadavky jsou analyzovány a ověřovány.*

Analýzy jsou prováděny tak, aby zjistily, jaký dopad bude mít dané prostředí na schopnost uspokojit potřeby zainteresovaných stran včetně očekávání, omezení a rozhraní. Analýza by měla brát v úvahu vše, včetně proveditelnosti, nákladových omezení nebo třeba potenciální velikost trhu. Požadavky jsou, aby se zvýšila pravděpodobnost, že výsledný produkt bude fungovat správně.

### **5.3.1.        *Establish Operational Concepts and Scenarios***

*Zřízení a udržení provozní koncepce.*

*Scénářem je myšlen typický sled událostí, které mohou nastat při vývoji nebo*

*při používání.*

### **5.3.2.            *Establish a Definition of Required Functionality and Quality Attributes***

*Zřídit a udržet definici požadované funkcionality.*

Jeden z přístupů k definování požadované funkcionality je analyzovat scénáře funkční analýzy. Tato analýza tak zahrnuje akce, sekvence, vstupy, výstupy nebo jiné informace, které bude produkt využívat. Výsledným popisem je nějaké seskupení funkcí s jejich požadavky. Tyto přístupy se vyvinuly v posledních letech prostřednictvím nových metod a nástrojů.

### **5.3.3.            *Analyze Requirements***

Analyzuje požadavky, které jsou nezbytné.

S ohledem na provozní koncepce a scénáře jsou požadavky analyzovány na jedné úrovni hierarchie, zda jsou dostačující k dosažení vyšších cílů. Analyzované požadavky pak poskytují základ pro podrobnější a přesnější požadavky na nižších úrovních hierarchie.

- Analyzuje potřeby zúčastněných stran, omezení, očekávání
- Analyzuje požadavky na určení, zda splňují cíle vyšší úrovně požadavků
- Analyzuje požadavky na zajištění, že jsou úplné, proveditelné, realizovatelný a ověřitelné
- Identifikujte klíčové podmínky, které mají silný vliv na náklady, výkon, nebo rizika
- Atd.



#### **5.3.4.            *Analyze Requirements to Achieve Balance***

*Analyzuje požadavky na udržení rovnováhy mezi omezeními a potřebami*

Zainterесované strany mohou řešit:

- Náklady
- Harmonogram
- Funkčnost
- Priority
- Rizika
- Atd

#### **5.3.5.            *Validate Requirements***

*Ověření požadavků, jestli výsledný produkt bude fungovat tak jak bylo zamýšleno v prostředí koncového zákazníka.*

Ověřování se provádí již v raných fázích vývoje. Tato aktivita bývá často integrována s činností řízení rizik. Zralé společnosti budou ověřovat požadavky více sofistikovaným způsobem za použití více technik.

Příklad metod používaných pro ověřování požadavků

- Analýza
- Prototyp
- Simulace
- Demonstrace (předvádění)

## 6. Závěr

Cílem práce bylo popsat detailněji procesní oblast - Requirements Development. Snažil jsem se popsat i souvislosti týkající této oblasti. Na počátku práce jsem uvedl jen základní informace o vývojovém centru SEI a základní informace o CMII. Do tohoto modelu jsem se pak pokusil zařadit RD a ten pak detailněji rozebrat. Jako určitý problém a omezení této práce bych viděl v tom, že velká většina informací o této problematice je v anglickém jazyce a jedná se většinou o odbornou angličtinu.

## 7. Zdroje

BRUCKNER, Tomáš et al. *Tvorba informačních systémů: principy, metodiky, architektury*. 1. vyd. Praha: Grada Publishing, 2012. 357 s. Management v informační společnosti. ISBN 978-80-247-4153-6.

BUCHALCEVOVÁ, Alena. *Metodiky budování informačních systémů*. Vyd. 1. Praha: Oeconomica, 2009. 205 s. Vysokoškolská učebnice. ISBN 978-80-245-1540-3.

HEMALA, David. *Zlepšování výkonnosti procesů organizace na bázi potenciálu zlepšení*. Praha, 2012. Vedoucí práce Dušan Chlapek.

TOMÁŠKOVÁ, Barbora. *Implementace metodiky řízení projektů v podmínkách IT společnosti*. Praha, 2010. Vedoucí práce Dušan Chlapek.

CMMI® for Development, Version 1.3, CMMI Product Team, listopad 2010.  
Dostupné z www: <http://www.sei.cmu.edu/reports/10tr033.pdf>