

Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação

**Uma Proposta**  
**Orientada a Perfis de Capacidade de Processo**  
**para Evolução da Melhoria de Processo de Software**

**Autor: Clênio Figueiredo Salviano**

**Orientador: Prof. Dr. Mario Jino**

Campinas – SP

Março de 2006

---

Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação

**Uma Proposta**  
**Orientada a Perfis de Capacidade de Processo**  
**para Evolução da Melhoria de Processo de Software**

**Autor: Clênio Figueiredo Salviano**

**Orientador: Prof. Dr. Mario Jino**

**Tese de Doutorado** apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Doutor em Engenharia Elétrica. Área de concentração: **Engenharia da Computação**

**Banca Examinadora**

Prof. Dr. Mario Lúcio Côrtes ..... IC/UNICAMP  
Prof. Dr. Mauro de Mesquita Spinola ..... POLI/USP  
Prof. Dr. Eleri Cardozo ..... DCA/FEEC/UNICAMP  
Prof. Dr. Ivan Luiz Marques Ricarte ..... DCA/FEEC/UNICAMP  
Prof. Dr. Maurício Ferreira Magalhães ..... DCA/FEEC/UNICAMP

Campinas – SP

Março de 2006

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Sa39p	<p>Salviano, Clênio Figueiredo Uma proposta orientada a perfis de capacidade de processo para evolução da melhoria de processo de software / Clênio Figueiredo Salviano. --Campinas, SP: [s.n.], 2006.</p>
	<p>Orientador: Mario Jino Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.</p>
	<p>1. Engenharia de software. 2. Indústria de software. 3. Engenharia de sistemas. I. Jino, Mario. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.</p>

Titulo em Inglês: A process capability profile driven proposal for software process improvement

Palavras-chave em Inglês: Software engineering, Software process improvement, Process capability model, Process capability profile, Process engineering, CMMI, ISO/IEC 15504

Área de concentração: Engenharia da Computação

Titulação: Doutor em Engenharia Elétrica

Banca examinadora: Mario Lúcio Côrtes, Mauro de Mesquita Spinola, Eleri Cardozo, Ivan Luiz Marques Ricarte e Mauricio Ferreira Magalhães

Data da defesa: 03/03/2006

## COMISSÃO JULGADORA - TESE DE DOUTORADO

**Candidato(a):** Clênio Figueiredo Salviano  
**Data da Defesa:** 3 de março de 2006  
**Título da Tese:** "Uma Proposta Orientada a Perfis de Capacidade de Processo para Evolução da Melhoria de Processo de Software."

### AVALIAÇÃO FINAL

Será considerada aprovada a tese que obtiver aprovação da maioria dos membros da Comissão Julgadora.

Votos Favoráveis: 06

Votos Contrários: 0

Resultado (Aprovada ou Reprovada): Aprovada

---

Prof. Dr. Mario Jino (Presidente): Mario Jino  
Prof. Dr. Mário Lucio Côrtes: Mário Lucio Côrtes  
Prof. Dr. Mauro de Mesquita Spínola: Mauro de Mesquita Spínola  
Prof. Dr. Eleri Cardozo: Eleri Cardozo  
Prof. Dr. Ivan Luiz Marques Ricarte: Ivan Luiz Marques Ricarte  
Prof. Dr. Maurício Ferreira Magalhães: Maurício Ferreira Magalhães

## Resumo

Melhoria de processo de software baseada nos níveis de maturidade fixos da arquitetura estagiada dos modelos SW-CMM e CMMI não atendem à diversidade de contextos e objetivos estratégicos das organizações intensivas em software. Esta pesquisa evolui a área de melhoria de processo com a utilização da flexibilidade da arquitetura contínua da ISO/IEC 15504 para um melhor alinhamento da melhoria com o contexto e objetivos estratégicos específicos de cada organização, principalmente as micro e pequenas. São identificadas três gerações de arquiteturas de modelos de capacidade de processo (estagiada fixa, contínua fechada e contínua aberta). É proposta uma engenharia de processo, de software e de qualquer outro trabalho humano intensivo em conhecimento, dirigida por perfis de capacidade de processo. É desenvolvida uma abordagem exemplo dessa engenharia com um conjunto de propriedades, um modelo unificado, um ciclo de melhoria e um conjunto de medições, bem como um método para o estabelecimento de perfis de capacidade de processo. As propostas são validadas com análises, aplicações práticas e uma agenda de pesquisa.

**Palavras-chave:** engenharia de software, melhoria de processo de software, modelo de capacidade de processo, perfil de capacidade de processo, engenharia de processo, CMMI, ISO/IEC 15504.

## Abstract

Software process improvement based on the staged architecture's fixed maturity levels of SW-CMM and CMMI models do not support the diversity of software intensive organizations context and strategic objectives. This research evolves the current process improvement area taking advantage of the flexibility of ISO/IEC 15504 continuous architecture towards a better alignment between an improvement with each organization's specific context and strategic objectives, especially of micro and small ones. Three generations of architectures for process capability models (fixed staged, closed continuous, open continuous) are identified. A process capability profile driven, software and any other knowledge intensive human work, process engineering is proposed. An exemplar approach for this engineering is developed, with a set of properties, a unified model, an improvement cycle and a measurement set, as well as a method for establishing process capability profile. These proposals are validated by analyses, practical applications and a research agenda.

**Keywords:** software engineering, software process improvement, process capability model, process capability profile, process engineering, CMMI, ISO/IEC 15504.

*À minha esposa Adriana e aos meus filhos Marcos e Daniel.*

## Agradecimentos

Nesta jornada, iniciada em 1992, no CTI e SMU e continuada no CTI/ITI/CenPRA, FEEC Unicamp e outros, contei com a ajuda de várias pessoas e entidades, as quais agradeço:

*Adriana, Marcos e Daniel, pelos ensinamentos, carinho, amor e paciência, meus pais, Cleôneo e Maria Olga (ambos em memória), irmãos, cunhados, sobrinhos e tios, colegas e amigos Ana Cristina Rouiller, Ana Maria Monte, Christianne von Wangenheim, Cristina Machado, Mário Nascimento e Romildo Monte, orientador Mario Jino, de 2004 a 2006, pela orientação, compreensão e paciência, ex-orientadores Manuel Mendes [2002-2004], Suku Nair [1994-1997] e Murat Tanik [1992-1994], membros da banca Eleri Cardozo, Ivan Ricarte, Mario Côrtes, Maurício Magalhães e Mauro Spinola, outros colegas do CenPRA, dos mais simples à diretoria, entre eles, Adriana Silveira, Alfredo Tsukumo, Angela Alves, Arthur Catto, Carlos Mammana, Carlos Passos, Cleonice Teracine, Fernando Cavazzoni, Jamiro Wanderley, Márcia Costa, Maria Alves (que um dia ao limpar minha sala, sempre muito bagunçada, perguntou porque eu não fazia um 5S para ajudar a mim e a ela), Miguel Argollo, Sônia Maintinguer, Vera Abreu, Vera Bianconi e Wagner De Martino, outros colegas de vários outros projetos, entidades e empreitadas relacionadas (15504MPE, ABNT, Ampla, CELEPAR, CPqD, ITS, MPS-BR, PBQP, Senior, Simpros, Softex Campinas, UNIVALI, ...), entre eles, Adilson Nicoletti, Afonso Soares, Alessandra Anacleto, Austregésilo Gonçalves, Berta Flecha, Danilo Scarlet, Diva Marinho, Edvar Pêra Jr., Eratóstenes Araújo, José Marcos Gonçalves, Kênia Pickler, Kival Weber, Marcello Thierry, Mario Côrtes, Marisa Marques, Odair Silva, Ozeas Santana, Raul Wuo e Renato Della Volpe; do grupo “WG10 SPICE” Alastair Walker, Alec Dorling, Terry Rout e Timo Varkoi; e do “grupo de lavras”, iniciado naquela noite, no boteco de Recife, Alexandre, Ana Liddy, André, Cleide, Giovani, Giuliano, Heron, Jean Lobo, Jones, Marcelo Oliveira, Marcelo Pessoa, Nilson, Renata, Renato e Teresa, alunos e orientados dos vários cursos de especialização em melhoria de processo, MCT, CenPRA e CNPq pelo apoio, oportunidade, compreensão e financiamento do trabalho, professores, funcionários e colegas da Unicamp e SMU, e outras crianças, incluindo algumas que já não são tão crianças mais, Daniel, Bruno, Indira, Marianna, Luana, Layla, Erik, Amanda, Mariah, Giovanna, Andressa, Mariana, Madalena, Joana e tantas outras, que sempre nos fazem lembrar, e reaprender, a infância, que no meu caso, “foi em Pedra Azul e em toda parte, onde eu tive o que sou”.*

*“the way you see the problem, is the problem”*

Stephen R. Covey [1989, p. 40]

*“... ensinar não é transferir conhecimento,  
mas criar as possibilidades para a sua produção ou a sua construção”*

Paulo Freire [1996, p. 25]

*“... o medo de amar é não arriscar,  
esperando que façam por nós o que é nosso dever, recusar o poder  
... o medo de amar é o medo de ter de a todo momento escolher,  
com acerto e precisão, a melhor direção”*

em “O medo de amar é o medo de ser livre”,  
letra de Fernando Brant para música de Beto Guedes

*“All models are wrong, but some are useful”*

George Box [1979]

*“1. No model can MAKE you improve,  
and no model can PREVENT you from improving.  
2. No model can MAKE you do stupid things,  
and not model can PREVENT you from doing stupid things.”*

Patrick O’Toole [2006]

**Lost**

*“Stand still. The trees ahead and bushes beside you  
Are not lost. Wherever you are is called Here,  
And you must treat it as a powerful stranger,  
Must ask permission to know it and be known.  
The forest breathes. Listen. It answers,  
I have made this place around you,  
If you leave it you may come back again, saying Here.  
No two trees are the same to Raven.  
No two branches are the same to Wren.  
If what a tree or a bush does is lost on you,  
You are surely lost. Stand still. The forest knows  
Where you are. You must let it find you.”*  
David Wagoner, in [Whyte 1996, pp 259-260]

**(Perdido)**

*“Pare. As árvores a frente e os arbustos atrás de você  
Não estão perdidos. Onde você está é chamado de Aqui,  
E você tem que trata-lo como um estranho poderoso,  
Tem que pedir permissão para conhece-lo e ser conhecido.  
A floresta respira. Escuta. Ela responde,  
Eu fiz este lugar em volta de você,  
Se você deixa-lo você talvez volte novamente, dizendo Aqui.  
Duas árvores nunca são as mesmas para o gavião.  
Dois galhos nunca são os mesmos para o canário.  
Se o que uma árvore ou um galho faz está perdido em você,  
Você está realmente perdido. Pare. A floresta sabe  
onde você está. Você tem que deixar que ela encontre você.”*  
Tradução livre por Clênio F. Salviano, 2005

# Sumário

<b>Lista de Figuras</b> .....	<b>xvi</b>
<b>Lista de Tabelas</b> .....	<b>xix</b>
<b>Siglas e Acrônimos</b> .....	<b>xxi</b>
<b>Principais trabalhos publicados pelo autor</b> .....	<b>xxvi</b>
<b>1 Introdução</b> .....	<b>1</b>
1.1 Apresentação .....	1
1.2 Problemas e oportunidades .....	3
1.3 Objetivos e propostas da pesquisa .....	4
1.4 Metodologia e processo da pesquisa.....	5
1.5 Estrutura da tese .....	8
<b>2 Melhoria de processo de software</b> .....	<b>9</b>
2.1 Engenharia de software e melhoria de processo de software.....	9
2.2 Vocabulário e conceitos básicos .....	15
2.2.1 Processo .....	16
2.2.2 Área de melhoria de processo de software .....	20
2.2.3 Contexto e objetivos estratégicos .....	23
2.2.4 Modelo e engenharia de modelo.....	24
2.2.5 Abordagem para melhoria de processo.....	31
2.2.6 Modelo de capacidade de processo.....	34
2.2.7 Capacidade de processo .....	35
2.2.8 Perfil de capacidade de processo .....	40
2.2.9 Medição .....	42
2.3 Modelos de capacidade de processo .....	47
2.3.1 Framework ISO/IEC 15504 e modelo ISO/IEC 15504-5.....	48
2.3.2 Modelo FAA iCMM .....	53
2.3.3 Framework SEI CMMI e modelo CMMI-SE/SW .....	55
2.3.4 MPS.BR e modelo MR-MPS .....	59
2.3.5 Outros modelos de capacidade de processo.....	61
2.3.6 Outros modelos de referência utilizados para melhoria de processo.....	63
2.4 Tendências de modelos e abordagens para melhoria de processo .....	68

---

2.4.1	Relacionamentos e integração de modelos .....	68
2.4.2	Comparações e combinações das arquiteturas estagiada e contínua .....	71
2.4.3	Escolha de processos de modelos contínuos.....	83
2.5	Síntese do capítulo .....	87
<b>3</b>	<b>Gerações de arquiteturas de modelos de capacidade de processo.....</b>	<b>91</b>
3.1	Antecedentes para as gerações de arquiteturas .....	91
3.2	Pressupostos para as gerações de arquiteturas .....	94
3.3	Primeira geração: Arquitetura estagiada fixa.....	97
3.4	Segunda geração: Arquitetura contínua fechada.....	98
3.5	Terceira geração: Arquitetura contínua aberta.....	99
3.6	Síntese do capítulo .....	99
<b>4</b>	<b>Engenharia de processo dirigida por perfis de capacidade .....</b>	<b>100</b>
4.1	Fundamentos para uma engenharia de processo .....	100
4.2	Definição de engenharia de processo .....	101
4.3	Desdobramentos da engenharia de processo .....	104
4.4	Síntese do capítulo .....	111
<b>5</b>	<b>Abordagem PRO2PI para engenharia de processo .....</b>	<b>112</b>
5.1	Pressupostos, requisitos, apresentação e visão geral.....	112
5.2	PRO2PI-PROP: Propriedades de PRO2PI .....	121
5.3	PRO2PI-MODEL: Modelo de PRO2PI.....	125
5.4	PRO2PI-MEAS: Medições para PRO2PI .....	134
5.5	PRO2PI-CYCLE: Processo para ciclo de melhoria com PRO2PI .....	136
5.6	Síntese do capítulo .....	142
<b>6</b>	<b>Método PRO2PI-WORK para estabelecimento de PRO2PI.....</b>	<b>144</b>
6.1	Apresentação .....	144
6.2	PRO2PI-WORK descrito como um processo.....	146
6.3	Gerência de PRO2PI-WORK .....	150
6.4	PRO2PI-WORK para ensino de engenharia de processo.....	150
6.5	Síntese do capítulo .....	151
<b>7</b>	<b>Análises e aplicações .....</b>	<b>152</b>
7.1	Análise de modelos relevantes, gerações de arquitetura e PRO2PI.....	153
7.2	Aplicações de PRO2PI .....	162

---

7.2.1	Melhoria de processo de software na Empresa E1 [1999-2002] .....	162
7.2.2	Melhoria de processo de software na Empresa E2 [2002-2003] .....	166
7.2.3	Estabelecimento de perfil de capacidade de processo [2000-2005] .....	168
7.2.4	Projeto 15504MPE [2003-2004].....	177
7.2.5	Melhoria de processo em grupos de empresas [2004-2005].....	180
7.2.6	PRO2PI para domínios específicos [2004-2005].....	180
7.2.7	Cursos com PRO2PI-WORK [2004-2005] .....	182
7.3	Síntese das análises e aplicações.....	185
<b>8</b>	<b>Conclusão.....</b>	<b>187</b>
8.1	Síntese do trabalho .....	187
8.2	Principais contribuições do trabalho .....	188
8.3	Relacionamentos com outras propostas .....	190
8.4	Limitações .....	193
8.5	Trabalhos futuros .....	193
8.6	Considerações finais .....	196
	<b>Referências bibliográficas .....</b>	<b>198</b>
<b>A.1</b>	<b>Diagrama de Classe de PRO2PI-MODEL .....</b>	<b>219</b>
A1.1	Referência para prática.....	219
A1.2	Elemento de referência para prática.....	220
A1.3	Grupo de elementos de referência para prática .....	221
A1.4	Sistema de grupo de elementos de referência para prática .....	222
<b>A.2</b>	<b>Medição de complexidade de uma melhoria.....</b>	<b>224</b>
<b>A.3</b>	<b>Fases e atividades de PRO2PI-WORK .....</b>	<b>228</b>
A3.1	PRO2PI-WORK Fase 1: Preparação do trabalho.....	232
A3.2	PRO2PI-WORK Fase 2: Escolha de PRO2PI.....	233
A3.3	PRO2PI-WORK Fase 3: Orientações para PRO2PI .....	243
A3.4	PRO2PI-WORK Fase 4: Conclusão do trabalho.....	245

---

## Lista de Figuras

FIGURA 1 – OBJETIVOS E PROPOSTAS DA PESQUISA .....	4
FIGURA 2 – METODOLOGIA E PROCESSO DESSA PESQUISA .....	6
FIGURA 3 – “NASCIMENTO” DA ENGENHARIA DE SOFTWARE .....	10
FIGURA 4 - CONCEITOS BÁSICOS DA MELHORIA DE PROCESSO DE SOFTWARE .....	16
FIGURA 5 – EXEMPLO DE REPRESENTAÇÃO DE PROCESSO EM ETVX .....	17
FIGURA 6 – ESSÊNCIA DA REPRESENTAÇÃO DE PROCESSO DO SPEM EM UML .....	18
FIGURA 7 – DUAS PROPOSTAS SEMELHANTES DE ARQUITETURA DE PROCESSO.....	19
FIGURA 8 - MODELO DE ARQUITETURA PARA O SISTEMA DE EPS [KINNULA 1999] .....	21
FIGURA 9 – MODELOS COMO REPRESENTAÇÃO DE SISTEMAS DO MUNDO .....	25
FIGURA 10 – CONCEITOS E RELAÇÕES CENTRAIS DA OOSE E MDE .....	27
FIGURA 11 - METAMODELOS COMO ONTOLOGIAS SIMPLES [BÉZIVIN2003A] .....	28
FIGURA 12 – METAMODELAGEM DE QUATRO CAMADAS E EXEMPLO PARA PROCESSO.....	29
FIGURA 13 – ETIMOLOGIA DE “MED” A “MODEL” E “MED” .....	30
FIGURA 14 – QUATRO ABORDAGENS SIMILARES PARA CICLOS DE MELHORIA DE PROCESSO .....	32
FIGURA 15 – PASSOS PARA UM CICLO DE MELHORIA DE PROCESSO [ISO/IEC 15504-4] .....	33
FIGURA 16 – PASSOS PARA DETERMINAÇÃO DA CAPACIDADE [ISO/IEC 15504-4] .....	34
FIGURA 17 – QUATRO TIPOS DE MODELOS E O PROCESSO DE UMA ORGANIZAÇÃO .....	35
FIGURA 18 – EXEMPLO DE UM PCP NO MODELO ISO/IEC 15504-5 .....	41
FIGURA 19 –OUTRO EXEMPLO DE PCP, NO MODELO CMMI-SE/SW .....	42
FIGURA 20 – VISÃO GERAL DO MODELOS DE MEDIÇÃO DA ISO/IEC 15939 .....	43
FIGURA 21 – MODELO DE INFORMAÇÕES DE MEDIÇÃO [ISO/IEC 15939 2002 P.20] .....	44
FIGURA 22 – EXEMPLO DE MEDIÇÃO EM PSM [AGUIAR 2002, SLIDE 55] .....	45
FIGURA 23 - MODELO DO PROCESSO DE MEDIÇÃO DE SOFTWARE [ISO/IEC 15939 2002, P.9].....	46
FIGURA 24 - HISTÓRICO E VERSÕES DOS MODELOS DE CAPACIDADE.....	47
FIGURA 25 – MODELO CONCEITUAL DA ISO/IEC 15504 .....	48
FIGURA 26 – EXEMPLOS DE MODELOS E MÉTODOS NA VISÃO DA ISO/IEC JTC1 SC7 .....	51
FIGURA 27 – ESTRUTURA DETALHADA DO MODELO ISO/IEC 15504-5.....	52
FIGURA 28 – ESTRUTURA E ELEMENTOS DO MODELO ISO/IEC 15504-5 [2006].....	53
FIGURA 29 – ESTRUTURA MAIS DETALHADA DO MODELO ICMM V2.0 .....	54
FIGURA 30 – ESTRUTURA SIMPLIFICADA E ELEMENTOS DO MODELO ICMM V2.0.....	55
FIGURA 31 – ESTRUTURA MAIS DETALHADA DOS MODELOS DO CMMI.....	57

---

FIGURA 32 – ESTRUTURA SIMPLIFICADA E ELEMENTOS DO MODELO CMMI-SE/SW v1.1 .....	58
FIGURA 33 – ESTRUTURA MAIS DETALHADA DOS ELEMENTOS DO MODELO MR-MPS v1.0 .....	60
FIGURA 34 – ESTRUTURA SIMPLIFICADA E ELEMENTOS DO MODELO MR-MPS v1.0.....	61
FIGURA 35 – MODELOS DE EXCELÊNCIA DA ISO 9001, EFQM E PNQ .....	64
FIGURA 36 - ESTRUTURA SIMPLIFICADA E ELEMENTOS DO MODELO PMBOK THIRD EDITION.....	66
FIGURA 37 – FRAMEWORK QUAGMIRE DE MODELOS ©2001 SPC.....	69
FIGURA 38 – PROPOSTA ALTERNATIVA PARA NÍVEIS DE MATURIDADE DO CMMI-SE/SW .....	77
FIGURA 39 – RELACIONAMENTO ENTRE OS NÍVEIS DE MATURIDADE DO CMMI E MPS-BR.....	77
FIGURA 40 - PERFIS DE PROCESSO NOS NÍVEL 1 A 4 DE MATURIDADE DO CMMI-SE/SW.....	78
FIGURA 41 – PERFIL DE PROCESSO NO NÍVEL 5 DE MATURIDADE DO CMMI-SE/SW .....	79
FIGURA 42 – UTILIZAÇÃO PROPOSTA POR MYERS E MODELO PROPOSTO POR ROUT.....	80
FIGURA 43 – EXEMPLO DE ESCOLHA DE PROCESSOS E NÍVEIS DE CAPACIDADE COM A 15504-4.....	83
FIGURA 44 – METODOLOGIA OWPL.....	86
FIGURA 45 – QUINZE MODELOS, SEUS TIPOS E RELACIONAMENTOS COM O PROCESSO .....	90
FIGURA 46 – POSIÇÕES SOBRE MODELO E UTILIZAÇÃO DAS ARQUITETURAS ESTAGIADA E CONTÍNUA.....	90
FIGURA 47 – ABORDAGEM AMP1 PARA MELHORIA DE PROCESSO.....	92
FIGURA 48 – ABORDAGEM AMP1 E MÉTODO MEP1 .....	93
FIGURA 49 – FASES, ATIVIDADES E RESULTADOS DO MÉTODO MEP1 .....	93
FIGURA 50 – NÍVEIS DE CAPACIDADE [ISO/IEC 15504-2 2003] .....	96
FIGURA 51 – PAR CONSISTENTE PERFIL DE CAPACIDADE DE PROCESSO E PROCESSO .....	101
FIGURA 52 - DIAGRAMA DE VENN DA PCDE E OUTRAS ENGENHARIAS.....	104
FIGURA 53 – ÁREAS E CONTEXTO DA ENGENHARIA PCDE .....	105
FIGURA 54 – (ÁREA DE) PROCESSO DE ESTABELECIMENTO DE PCP.....	105
FIGURA 55 – PERFIL NO RELACIONAMENTO DE MODELOS COM O PROCESSO .....	108
FIGURA 56 – CONCEITOS BÁSICOS PARA ENGENHARIA PCDE.....	109
FIGURA 57 – PCDE NA ETIMOLOGIA PROPOSTA POR FAVRE [2005].....	110
FIGURA 58 – DIAGRAMA DA UTILIZAÇÃO CORRENTE DA MELHORIA DE PROCESSO.....	114
FIGURA 59 – DIAGRAMA DA DEFINIÇÃO DE PRO2PI.....	115
FIGURA 60 - DIAGRAMA DA UTILIZAÇÃO (E DEFINIÇÃO) DE PRO2PI.....	116
FIGURA 61 - DIAGRAMA DA AVALIAÇÃO DE PROCESSO (E DEFINIÇÃO E UTILIZAÇÃO DE PRO2PI).....	117
FIGURA 62 – DIAGRAMA DA DEFINIÇÃO DE MODELOS MAIS ESPECÍFICOS .....	117
FIGURA 63 – DIAGRAMA DA ABORDAGEM PRO2PI PARA MODELOS E MELHORIA DE PROCESSO.....	119
FIGURA 64 – DIAGRAMA, ANÁLOGO AO DE PRO2PI, DO DESENVOLVIMENTO DE SOFTWARE .....	119
FIGURA 65 – PROPRIEDADES DE PRO2PI.....	121
FIGURA 66 – PRO2PI-MODEL E OUTROS MODELOS DE PROCESSO OU SOFTWARE.....	127

---

FIGURA 67 - PRO2PI-MODEL E OUTROS METAMODELOS E MODELOS DE PROCESSO E SOFTWARE.....	128
FIGURA 68 – DIAGRAMA SIMPLIFICADO DE PRO2PI-MODEL .....	130
FIGURA 69 - DIAGRAMA DE CLASSES DE PRO2PI-MODEL EM UML .....	132
FIGURA 70 – PRINCIPAIS ELEMENTOS DE PRO2PI-MODEL E SEUS RELACIONAMENTOS .....	133
FIGURA 71 – ELEMENTOS DA ESTRUTURA DO CMMI E 15504-5 EM PRO2PI-MODEL .....	133
FIGURA 72 - PRODUTO DE INFORMAÇÃO VIABILIDADE DE PRO2PI.....	135
FIGURA 73 - PRO2PI-CYCLE: PROCESSO PARA CICLO DE MELHORIA COM PRO2PI.....	137
FIGURA 74 – FASES DE PRO2PI-CYCLE E DE IDEAL, CICLO 15504 E AMP1 .....	138
FIGURA 75 – PRO2PI-CYCLE COMO UM PROCESSO NA NOTAÇÃO ETVX.....	139
FIGURA 76 - FASES DO MÉTODO PRO2PI-WORK E RELAÇÃO COM PRO2PI-CYCLE .....	145
FIGURA 77 – PRO2PI-WORK COMO UM PROCESSO EM ETVX .....	147
FIGURA 78 – FASES E ATIVIDADES DO MÉTODO PRO2PI-WORK.....	148
FIGURA 79 – LOCAIS, TIPOS E TEMPOS TÍPICOS DE UMA REALIZAÇÃO DO MÉTODO.....	149
FIGURA 80 – PRO2PI-WORK PARA CURSO.....	151
FIGURA 81 – POSICIONAMENTO DOS MODELOS RELEVANTES NAS GERAÇÕES DE MODELOS.....	155
FIGURA 82 – RESULTADOS DA AVALIAÇÃO NA EMPRESA E1 EM 1999 E 2002 .....	164
FIGURA 83 – PROCESSO DA FÁBRICA DE SOFTWARE E PROCESSOS ESCOLHIDOS .....	168
FIGURA 84 – OBJETIVOS E ESTRATÉGIA DE NEGÓCIOS EM BALANCED SCORECARD.....	169
FIGURA 85 – PROCESSO PARA GESTÃO E ACOMPANHAMENTO DE CONTRATO .....	170
FIGURA 86 – PROCESSO DE CONTRATAÇÃO DA ORGANIZAÇÃO .....	172
FIGURA 87 – DUAS FASES DO PROJETO DE MELHORIA.....	175
FIGURA 88 – MÉTODO MARES, COM DETALHE DA CONTEXTUALIZAÇÃO [ANACLETO 2005].....	178
FIGURA 89 – QUADRO DE RELEVÂNCIA DOS PROCESSOS EM TRÊS EMPRESAS .....	181
FIGURA 90 – SÍNTESE DOS OBJETIVOS, PROPOSTAS E VALIDAÇÕES DA PESQUISA .....	189
FIGURA 91 – VERSÕES DE PRO2PI E RELACIONAMENTOS COM OUTRAS PROPOSTAS .....	190
FIGURA 92 – EXEMPLO DE RELEVÂNCIA DE PROCESSOS DO CMMI-SE/SW .....	238
FIGURA 93 – EXEMPLO DE RELEVÂNCIA DE PROCESSOS DA ISO/IEC 15504-5 .....	239
FIGURA 94 – MODELO DE MEDIÇÃO PARA RELEVÂNCIA DE UMA ÁREA DE PROCESSO.....	239
FIGURA 95 – DOIS EXEMPLOS DE PERFIS DE CAPACIDADE DE PROCESSO.....	242
FIGURA 96 – MÉTODO PRO2PI-WORK, PRO2PI E SEUS PRINCIPAIS RELACIONAMENTOS .....	246

## Lista de Tabelas

TABELA 1 - SIGLAS ORIGINAIS EM PORTUGUÊS UTILIZADAS NO TEXTO .....	XXI
TABELA 2 - SIGLAS ORIGINAIS EM INGLÊS UTILIZADAS NO TEXTO .....	XXI
TABELA 3 - SIGLAS DAS ÁREAS DE PROCESSO DO CMMI-SE/SW VERSÃO 1.1 .....	XXIV
TABELA 4 - SIGLAS DOS GRUPOS DE PROCESSO DA ISO/IEC 15504-5 .....	XXIV
TABELA 5 - SIGLAS DEFINIDAS NO TEXTO.....	XXV
TABELA 6 – PERGUNTAS CHAVE E ELEMENTOS QUE AS RESPONDEM .....	18
TABELA 7 – PROCESSO DE ENGENHARIA DE DOMÍNIO DA ISO/IEC 15504-5 [2006] .....	37
TABELA 8 – NÍVEL 2 DE CAPACIDADE, SEM SEUS ATRIBUTOS, DA ISO/IEC 15504-5 [2006] .....	38
TABELA 9 - ATRIBUTO PA 2.1, SEM OS INDICADORES, DA ISO/IEC 15504-5 [2006].....	38
TABELA 10 – ATRIBUTO PA 2.2, COM OS INDICADORES, DA ISO/IEC 15504-5 [2005] .....	39
TABELA 11 – REFERÊNCIAS PARA COMPARAÇÕES DIRETAS ENTRE MODELOS .....	68
TABELA 12 – REPRESENTAÇÕES CONTÍNUA E ESTAGIADA DO CMMI .....	72
TABELA 13 – COMPARAÇÕES ENTRE AS REPRESENTAÇÕES ESTAGIADA E CONTÍNUA [KASSE 2004] .....	73
TABELA 14 - ARQUITETURA CONTÍNUA DA 15504 E ESTAGIADA DO SW-CMM.....	75
TABELA 15 – VANTAGENS E DESVANTAGENS SEGUNDO OLSON [2003] .....	82
TABELA 16 – GERAÇÕES DE ARQUITETURA DE MODELOS DE CAPACIDADE DE PROCESSO .....	97
TABELA 17 – (ÁREA DE) PROCESSO DE ESTABELECIMENTO DE PCP COMO 15504-5.....	106
TABELA 18 – NOMES DAS CLASSES DE PRO2PI-MODEL EM INGLÊS E PORTUGUÊS.....	132
TABELA 19 - ARTEFATOS DE PRO2PI-WORK .....	149
TABELA 20 - OBJETIVO DA ANÁLISE DAS GERAÇÕES DE ARQUITETURA.....	152
TABELA 21 – OBJETIVO DA ANÁLISE DAS UTILIZAÇÕES DE PRO2PI .....	153
TABELA 22 – OBJETIVO DA ANÁLISE DAS UTILIZAÇÕES DE PRO2PI-WORK .....	153
TABELA 23 - MODELOS E CONCEITOS DA ENGENHARIA DE PROCESSO.....	156
TABELA 24 – MODELOS E CLASSES DE PRO2PI-MODEL .....	161
TABELA 25 – PCPS DA MELHORIA DE PROCESSO .....	166
TABELA 26 – PCPS DO QUINTO PROJETO .....	172
TABELA 27 – PCPS DO NONO PROJETO .....	176
TABELA 28 – SÍNTESE DOS PCPS ESTABELECIDOS EM NOVE PROJETOS DE MELHORIA .....	176
TABELA 29 – EXPERIÊNCIA COM PRO2PI-WORK E TRABALHOS GERADOS .....	183
TABELA 30 – RESUMO DOS TRABALHOS DE CURSOS PRO2PI-WORK.....	185
TABELA 31 – SÍNTESE DAS UTILIZAÇÕES DE PRO2PI.....	186

TABELA 32 – VALORES DAS PONTUAÇÕES DE ATRIBUTO DE PROCESSO.....	226
TABELA 33 – EXEMPLO DE VALORES DE REDUÇÃO PARA A COMPLEXIDADE DE UM PRO2PI .....	227
TABELA 34 – EXEMPLO DE COMPLEXIDADE PARA UM PRO2PI.....	227
TABELA 35 – OUTRO EXEMPLO DE COMPLEXIDADE PARA OUTRO PRO2PI .....	227
TABELA 36 - ARTEFATOS DE PRO2PI-WORK.....	229
TABELA 37 - PRODUTOS DE TRABALHO E ATIVIDADES: ENTRADA, ATUALIZADO OU SAÍDA. ....	230
TABELA 38 - QUESTÕES EXEMPLO PARA FATORES ORGANIZACIONAIS .....	235
TABELA 39 - TRECHO DO QUESTIONÁRIO PARA RELEVÂNCIA DE ÁREA DE PROCESSO.....	237
TABELA 40 - VALORES PARA COMBINAÇÕES DA IMPORTÂNCIA E RISCO .....	240
TABELA 41 - EXEMPLOS DE RESPOSTAS PARA QUESTÃO 3.10 E MAPEAMENTO PARA 15504-5.....	241

## Siglas e Acrônimos

Tabela 1 - Siglas originais em Português utilizadas no texto

<b>Sigla</b>	<b>Nome original em Português</b>	<b>Tradução para Inglês</b>
15504MPE	ISO/IEC 15504 para Micro e Pequenas Empresas	ISO/IEC 15504 for Micro and Small Organizations
CenPRA	Centro de Pesquisa Renato Archer	'Renato Archer' Research Center
Coop-MPS	Método para Projetos Cooperativos de Melhoria de Processo de Software	Method for Cooperative Projects of Software Process Improvement
FPNQ	Fundação Prêmio Nacional da Qualidade	"National Prize for Quality" Foundation
MAE	Matriz de Análise Estratégica	Strategic Analysis Matrix
MA-MPS	Método de Avaliação para Melhoria de Processo de Software	Assessment Method for Software Process Improvement
MARES	Método de Avaliação Rápida para Empresas de Software	Rapid Assessment Method for Software Organizations
ProGer	Modelo de Processo para Gerenciamento de Projetos de Software	Process Model for Software Project Management
MPS-BR	Melhoria do Processo de Software Brasileiro	Brazilian Software Process Improvement
MR-MPS	Modelo de Referência para Melhoria de Processo de Software	Reference Model for Software Process Improvement
PNQ	Prêmio Nacional da Qualidade	National Prize for Quality
Softex	Excelência de Software (Brasileiro)	(Brazilian) Software Excellence

Tabela 2 - Siglas originais em Inglês utilizadas no texto

<b>Sigla</b>	<b>Nome original em Inglês</b>	<b>Tradução para Português</b>
ACM	Association for Computing Machinery	Associação para "Máquinas de Computação"
AMRDEC-SED	Aviation and Missele Research, Development and Engineering Center – Software Engineering Directorate	Diretório de Engenharia de Software do Centro de Pesquisa, Desenvolvimento e Engenharia de Aviação e Mísseis
BNF	Backus Naur Form	Forma de Backus e Naur
BSC	Balanced Scorecard	Marcador Balanceado
CBSE	Component Based Software Engineering	Engenharia de Software Baseada em Componentes
CD	Committee Draft (ISO/IEC term)	Versão Preliminar do Comitê
CMM or SW-CMM	Capability Maturity Model for Software	Modelo de Maturidade da Capacidade para Software
CMMI	Capability Maturity Model	Modelo Integrado de Maturidade da

<b>Sigla</b>	<b>Nome original em Inglês</b>	<b>Tradução para Português</b>
	Integration	Capacidade
CMMI-SE/SW	Capability Maturity Model Integration for System Engineering and Software Engineering	Modelo Integrado de Maturidade da Capacidade para Engenharia de Sistemas e Engenharia de Software
COBIT	Control Objectives for Information and Related Technology	Objetivos de Controle para Tecnologias de Informação e Relacionadas
DMAIC	Define, Measure, Analyze, Implement, Control	Definir, Medir, Analisar, Implementar, Controlar
DSL	Design Space Language	Linguagem para Espaço de Projeto
eSCM-SP	eSourcing Capability Model for Service Providers	Modelo de Capacidade de <i>eSourcing</i> para Provedores de Serviços
ESI	Embedded Systems Institute	Instituto de Sistemas Embutidos
ESI	European Software Institute	Instituto Europeu de Software
FAA	Federal Aviation Administration	Administração Federal de Aviação
FCD	Final Committee Draft (ISO/IEC term)	Versão Final do Comitê
FDIS	Final Draft for International Standard (ISO/IEC term)	Versão Final para Norma Internacional
GP	Generic Practice	Prática Genérica
GQM	Goal Question Metric (Measure)	Meta Questão Medição
iCMM	Integrated Capability Maturity Model	Modelo Integrado de Maturidade da Capacidade
IDEAL	Initiating, Diagnosing, Establishing, Acting, Learning	Inicia, Faz diagnóstico, Estabelece, Atua, Aprende
IEC	International Electrotechnical Commission	Comissão Internacional de Eletrotécnica
IEEE	Institute of Electrical and Electronics Engineers	Instituto dos Engenheiros Elétricos e Eletrônicos
IMPACT	Initiate, Measure, Plan, Act, Confirm and Transfer	Iniciar, Medir, Planejar, Atuar, Confirmar e Transferir
IPD-CMM	Integrated Product Development Capability Maturity Model	Modelo de Maturidade da Capacidade para Desenvolvimento Integrado de Produtos
IPPD	Integrated Product and Process Development	Desenvolvimento de Produtos e Processos Integrados
IS	International Standard (ISO/IEC term)	Norma Internacional
ISO	International Organization for Standardization	Organização Internacional para Normalização
ITsqc	Information Technology Service Qualification Center	Centro de Qualificação de Serviços em Tecnologia da Informação
KPA	Key Process Area	Área chave de processo
MDA	Model-Driven Architecture	Arquitetura Dirigida por Modelos
MDD	Model-Driven Development	Desenvolvimento Dirigida por Modelos
MDE	Model-Driven Engineering	Engenharia Dirigida por Modelos
NATO	North Atlantic Treat Organization	Organização do Tratado do Atlântico Norte (OTAN)

<b>Sigla</b>	<b>Nome original em Inglês</b>	<b>Tradução para Português</b>
OMG	Object Management Group	Grupo de Gerência de Objetos
OOSE	Object Oriented Software Engineering	Engenharia de Software Orientada a Objetos
OPM3	Organizational Project Management Maturity Model	Modelo de Maturidade da Gerência Organizacional por Projeto
OWPL	Walloon Observatory for Software Practice (original em francês: Observatoire Wallon des Pratiques Logicielles)	Observatório Wallon para Práticas de Software (Wallon: uma das línguas faladas na Bélgica)
PA	Process Area	Área de Processo
PA	Process Attribute (ISO/IEC 15504 term)	Atributo de Processo (termo da ISO/IEC 15504)
PAM	Process Assessment Model (ISO/IEC 15504 term)	Modelo para Avaliação de Processo (termo da ISO/IEC 15504)
PCL	Process Capability Level	Nível de Capacidade de Processo
PCP	Process Capability Profile	Perfil de Capacidade de Processo
PDCA	Plan Do Check Act	Planeja, Executa, Verifica, Atua
PMBOK	(Guide to) Project Management Body of Knowledge	(Guia para) Corpo de Conhecimento de Gerência de Projeto
PMI	Project Management Institute	Instituto de Gerência de Projeto
PRM	Process Reference Model (ISO/IEC 15504 term)	Modelo de Referência de Processo (termo da ISO/IEC 15504)
RAPID	Rapid Assessment for Process Improvement for Software Development (Method)	(Método) Avaliação Rápida para Melhoria de Processo para Desenvolvimento de Software
SDCA	Standard Do Check Act	Padroniza, Executa, Verifica, Atua
SE-CMM: EIA 731	System Engineering Capability Maturity Model	Modelo de Maturidade da Capacidade para Engenharia de Sistemas
SE	System Engineering	Engenharia de Sistemas
SEI	Software Engineering Institute	Instituto de Engenharia de Software
SEP	Software Engineering Process	Processo da Engenharia de Software
SM <sup>MM</sup> or SM <sup>CMM</sup>	Software Maintenance (Capability) Maturity Model	Modelo de Maturidade (da Capacidade) de Manutenção de Software
SFO	Strategy-Focused Organization	Organização focada em estratégia
SPC	Software Quality Consortium	Consórcio de Qualidade de Software
SPE	Software Process Engineering	Engenharia de Processo de Software (EPS)
SPEM	Software Process Engineering Metamodel	Metamodelo de Engenharia de Processo de Software
SPI	Software Process Improvement	Melhoria de Processo de Software (MPS)
SPICE	Software Process Improvement and Capability dEtermination	Melhoria de Processo de Software e Determinação da Capacidade
SS	Supplier Sourcing	Aquisição (fonte de fornecimento)
SW or SE	Software Engineering	Engenharia de Software
SWEBOK	(Guide to) Software Engineering Body of Knowledge	(Guia para) Corpo de Conhecimento de Engenharia de Software
SWOT	Strengths, Weakness, Opportunities,	Pontos fortes, fraquezas, oportunidades e

<b>Sigla</b>	<b>Nome original em Inglês</b>	<b>Tradução para Português</b>
	Threats	ameaças
TR	Technical Report (ISO/IEC term)	Relatório Técnico
U-CMMI	Unified Capability Maturity Model Integration	Modelo Integrado de Maturidade da Capacidade Unificado

Tabela 3 - Siglas das áreas de processo do CMMI-SE/SW versão 1.1

<b>Sigla</b>	<b>Nome Original em Inglês</b>	<b>Tradução para Português</b>
CAR	Causal Analysis and Resolution	Análise de Causa e Resolução
CM	Configuration Management	Gestão de Configuração
DAR	Decision Analysis and Resolution	Análise de Decisão e Resolução
IPM	Integrated Project Management	Gestão Integrada de Projeto
MA	Measurement and Analysis	Medição e Análise
OID	Organizational Innovation and Deployment	Inovação e Melhoria Organizacional
OPD	Organizational Process Definition	Definição do Processo Organizacional
OPF	Organizational Process Focus	Foco no Processo Organizacional
OPP	Organizational Process Performance	Desempenho do Processo Organizacional
OT	Organizational Training	Treinamento Organizacional
PI	Product Integration	Integração de Produto
PMC	Project Monitoring and Control	Acompanhamento e Controle de Projeto
PP	Project Planning	Planejamento de Projeto
PPQA	Process and Product Quality Assurance	Garantia da Qualidade de Processo e Produto
QPM	Quantitative Project Management	Gestão Quantitativa de Projeto
RD	Requirements Development	Desenvolvimento de Requisitos
REQM	Requirements Management	Gestão de Requisitos
RSKM	Risk Management	Gestão de Riscos
SAM	Supplier Agreement Management	Gestão de Acordos com Fornecedores
TS	Technical Solution	Solução Técnica
VAL	Validation	Validação
VER	Verification	Verificação

Tabela 4 - Siglas dos grupos de processo da ISO/IEC 15504-5

<b>Sigla</b>	<b>Nome original em Inglês</b>	<b>Nome traduzido para Português</b>
ACQ	Acquisition	Aquisição
SPL	Supplier	Fornecimento
ENG	Engineering	Engenharia
OPE	Operation	Operação
MAN	Management	Gerência
PIM	Process Improvement	Melhoria de processo
RIN	Resource and Infrastructure	Recursos e Infra-estrutura
REU	Reuse	Reuso
SUP	Support	Apoio

Tabela 5 - Siglas definidas no texto

<b>Sigla</b>	<b>Nome em Inglês</b>	<b>Nome em Português</b>
AMP1	Process Improvement Approach (Number) One	Abordagem para Melhoria de Processo (Número) Um
MEP1	Process Selection Method (Number) One	Método para Escolha de Processos (Número) Um
DM-IEPI	Derived Measured for Investment Effectiveness for Process Improvement	Medida Derivada para Eficiência de Investimento para Melhoria de Processo
DM-PIC	Derived Measure for Process Improvement Complexity	Medida Derivada para Complexidade de Melhoria de Processo
PACP	Process Area Capability Profile	Perfil de Capacidade de Área de Processo
PCM	Process Capability Model	Modelo de Capacidade de Processo
PCM-AG	Process Capability Model Architecture Generations	Gerações de Arquitetura de Modelos de Capacidade de Processo
PCME	Process Capability Model Engineering	Engenharia de Modelos de Capacidade de Processo
PCDE	{(Process Capability Profile)-Driven [Software and any other Knowledge Intensive Human Work] (Process Engineering)} or a short version {(Process Capability Profile)-Driven (Process Engineering)}	{(Engenharia de processo) (de Software e de qualquer outro Trabalho Humano Intensivo em Conhecimento) Dirigida por (Perfis de Capacidade de Processo)} ou de forma mais curta {(Engenharia de processo) Dirigida por (Perfis de Capacidade de Processo)}
PRO2PI	Process Capability Profile to Process Improvement	Perfil de Capacidade de Processo para Melhoria de Processo
PRO2PI-CYCLE	PRO2PI Improvement Cycle Process	Processo para Ciclo de Melhoria com PRO2PI
PRO2PI-MEAS	PRO2PI Measurements	Medições de PRO2PI
PRO2PI-MODEL	PRO2PI Model	Modelo de PRO2PI
PRO2PI-PROP	PRO2PI Properties	Propriedades de PRO2PI
PRO2PI-WORK	PRO2PI Establishment Workshop Method	Método para Oficina de Estabelecimento de PRO2PI
U-IEPI	Unit for Investment Effectiveness for Process Improvement	Unidade para Eficiência de Investimento para Melhoria de Processo
U-PIC	Unit for Process Improvement Complexity	Unidade para Complexidade de Melhoria de Processo

## Principais trabalhos publicados pelo autor

Dezoito trabalhos, publicados pelo autor (a maioria em co-autoria) e relacionados a esta tese estão identificados a seguir, classificados em três categorias: núcleo, experiência e desdobramento. Os quatro trabalhos classificados como núcleo são aqueles nos quais propostas da tese foram apresentadas. Os seis trabalhos considerados como desdobramento são aqueles nos quais outras propostas, direta ou indiretamente influenciadas pelas propostas desta tese, estão apresentadas. Os oito trabalhos considerados como experiência são aqueles nos quais experiências com as propostas da tese ou de desdobramentos da tese estão relatadas.

### **Categoria núcleo:**

- 1.1: Clênio F. Salviano, Eduardo P. de Souza, Ana C. B. Dominoni, Adilson S. Nicoletti, "Experiência de Avaliação de Processos e Planejamento da Melhoria Utilizando a Futura Norma ISO/IEC 15504 (SPICE)", Anais do WQS 1999 Workshop de Qualidade de Software do Simpósio Brasileiro de Engenharia de Software (SBES), p. 1-17, Florianópolis, SC, Outubro 1999.
- 1.2: Clênio F. Salviano, Um Método para Escolha dos Processos para uma Melhoria Alinhada aos Objetivos de Negócio, Anais do WQS 2001 Workshop de Qualidade de Software do Simpósio Brasileiro de Engenharia de Software (SBES), p. 39-50, Rio de Janeiro, RJ, Outubro 2001.
- 1.3: Clênio F. Salviano, Mario Jino e Manuel Mendes, Towards an ISO/IEC 15504-Based Process Capability Profile Methodology for Process Improvement (PRO2PI), in Proceedings of SPICE 2004 The Fourth International SPICE Conference, Lisbon, Portugal, p. 77-84, April 28-29, 2004.
- 1.4: Clênio F. Salviano and Mario Jino, Using Continuous Modes as "Dynamic and Specific Staged Models", slides from presentation at Fourth Annual CMMI Technology Conference and Users Group, Denver, USA, November 2004.

### **Categoria experiência:**

- 2.1: Clênio F. Salviano e Eduardo P. de Souza, "SPICE Trials and Dissemination in Brazil: 1996-1999", Proceedings of SPICE 2000 The First International SPICE Conference, Limerick, Ireland, June 10-11, 2000, p. 245-254.
- 2.2: Adilson S. Nicoletti e Clênio F. Salviano, An Experience using ISO/IEC TR 15504 and ISO 9000:2000 for Software Process Improvement, Proceedings of SPICE 2003 The Third International SPICE Conference, Nordwick, Netherlands, pp. 141-142, March 2003.
- 2.3: Odair J. da Silva, Carlos A. Borges, Clênio F. Salviano, Ana L. Sampaio, Adalberto N. Crespo e Ana Cristina Rouiller, An ISO/IEC 15504-Based Software Process Improvement Project in a Small Brazilian Software Organization, in Proceedings of SPICE 2003 The Third International SPICE Conference, 2003.

- 
- 2.4: Alessandra Anacleto, Christiane G. von Wangenheim, Clênio F. Salviano e Rafael Savi, Experiences from Applying 15504 to Small Software Companies in Brazil, in Proceedings of SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, p. 33-37, April 28-29, 2004.
  - 2.5: Flávia de Petri, Juliana Rodrigues, Luiz Mapelli, Vera Lúcia Oliveira e Clênio F. Salviano, Uma experiência de capacitação e início de melhoria de processo de software com o método PRO2PI-WORK, em Anais do SIMPROS 2005: Sétimo Simpósio Internacional de Melhoria de Processo de Software, São Paulo, SP, 21-23/11/2005, 12 páginas, 2005.
  - 2.6: Christiane G. von Wangenheim, Kênia K. Pickler, Marcello Thiry, Alessandra C. Zoucas, Clênio F. Salviano, "Aplicando avaliações de contextualização em processos de software alinhados ao CMMI-SE/SW", em Anais do SIMPROS 2005: Sétimo Simpósio Internacional de Melhoria de Processo de Software, São Paulo, SP, 21-23/11/2005, 13 páginas, 2005.
  - 2.7: Christiane G. von Wangenheim, Timo Varkoi e Clênio F. Salviano, Performing ISO/IEC 15504 Conformant Software Process Assessments in Small Software Companies, experience report at EUROSPI 2005.
  - 2.8: Christiane G. von Wangenheim, Alessandra Anacleto e Clênio F. Salviano. "Helping Small Companies Assess Software Processes", IEEE Software, vol. 23, no. 1, pp. 91-98, January/February 2006.

#### **Categoria desdobramento:**

- 3.1: Alessandra Anacleto, Christiane G. von Wangenheim, Clênio F. Salviano e Rafael Savi, A Method for Process Assessment in Small Software Companies, in Proceedings of SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, p. 69-76, April 28-29, 2004.
- 3.2: Jan Eduardo Lobo e Clênio F. Salviano, Experiência de Avaliação de Processos e Desenvolvimento de uma Ferramenta para Apoio Baseada na ISO/IEC TR 15504, Revista ProQualiti – Qualidade na Produção de Software, Núcleo de Estudos Avançados em Engenharia e Qualidade de Software, UFLA, Volume 1, Número 1, Maio 2005, ISSN 1807-5061.
- 3.3: Christiane G. von Wangenheim e Clênio F. Salviano, Consolidação de uma Metodologia para Avaliação de Processos de Software de MPEs Baseada na Norma ISO/IEC 15504 (SPICE) (Projeto PBQP 2.32 2004), artigo sobre projeto premiado do PBQP-Software (Programa Brasileiro de Qualidade e Produtividade de Software) ciclo 2004, publicado nos Anais do IV Simpósio Brasileiro de Qualidade de Software, SBQS, p. 21-26, Porto Alegre, RS, 6 a 10 de junho de 2005.
- 3.4: Kival C. Weber, Eratóstenes Araújo, Cristina A. F. Machado, Danilo Scalet, Clênio F. Salviano e Ana Regina C. da Rocha, Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software - versão 1.0 (MR-MPS e MA-MPS), Anais do IV Simpósio Brasileiro de Qualidade de Software, SBQS, p. 347-360, Porto Alegre, RS, 6 a 10 de junho de 2005.
- 3.5: Kival C. Weber, Eratóstenes E. R. Araújo, Ana Regina C. da Rocha, Cristina A. F. Machado, Danilo Scalet, Clênio F. Salviano, Brazilian Software Process Reference Model and Assessment Method, in Proceedings of The 20th International Symposium on Computer and Information Sciences (ISCIS'05), October 26-28<sup>th</sup>, Instambul, Turkey, LNCS 3733, pp. 402-411, 2005.

- 3.6: Clênio F. Salviano e Alfredo N. Tsukumo, Coop-MPS: Um Método para Projetos Cooperativos de Melhoria de Processo de Software e sua aplicação com o Modelo MPS.BR, ProQualiti – Qualidade na Produção de Software, Núcleo de Estudos Avançados em Engenharia e Qualidade de Software, UFLA, Volume 1, Número 2, Novembro 2005, ISSN 1807-5061.

# Capítulo 1

## 1 Introdução

Os objetivos deste capítulo são apresentar a área da pesquisa, identificar problemas e oportunidades, apresentar os objetivos, metodologia e processo da pesquisa, e apresentar a estrutura da tese.

### 1.1 Apresentação

O termo ‘engenharia de software’ foi definido na conferência NATO 68, “como uma provocação, para implicar na necessidade da construção de software ser baseada nos moldes dos fundamentos teóricos e disciplinas práticas, que são tradicionais nos tipos de engenharia já estabelecidos” [Naur e Randell 1969, p. 8]. Desde então a comunidade tem procurado entender a natureza das atividades de software e sua relação com as outras disciplinas da engenharia, e desenvolver métodos, técnicas, modelos e ferramentas para a engenharia de software. Tem havido um amadurecimento e avanço na área, porém software é ainda mais “produzido” e não exatamente “engenhariado” [Pollice 2005]. Segundo Herbsleb et al. [1997] “enquanto Fred Brooks nos avisava que não haveria uma única solução ‘bala de prata’ (*silver bullet*) para as dificuldades essenciais do desenvolvimento de software [Brooks 1986], Watts Humphrey e outros no *Software Engineering Institute* (SEI) estavam ocupados colocando idéias que viriam a ser o modelo SW-CMM” [Hersberg et al. 1997, p. 31].

A versão 1.1 do modelo SW-CMM<sup>1</sup> (*Capability Maturity Model for Software*) [Humphrey 1988, Humphrey 1989, SEI 1993, Paulk et al. 1994], lançada em 1993, estabeleceu a área de melhoria de processo de software com o objetivo de orientar a melhoria do processo de projetos de desenvolvimento de software por encomenda, como meio para a melhoria das organizações intensivas em software, tendo como referência níveis de maturidade. O modelo SW-CMM define a arquitetura

---

<sup>1</sup> O SW-CMM foi lançado com o nome de CMM. Como vários outros modelos foram lançados em seguida, cada um para uma disciplina diferente, foram utilizados prefixos para indicar a disciplina, como, por exemplo, SE-CMM (*System Engineering CMM*), SA-CMM (*Software Aquisition CMM*) e P-CMM (*People CMM*). Por isto, CMM passou a ser referenciado também como SW-CMM (*Software CMM*). Neste tese as duas identificações são utilizadas com o mesmo significado: CMM e SW-CMM.

estagiada com cinco níveis de maturidade, como estágios sequenciais e cumulativos, que atuam como referência para evolução do processo de imaturo, imprevisível e com baixa qualidade, para um processo maduro, sistemático e com qualidade.

Como uma alternativa à rigidez da arquitetura estagiada, foi proposta a arquitetura contínua pela ISO/IEC 15504, que também é conhecida como modelo SPICE (*Software Process Improvement and Capability dEtermination*). A ISO/IEC 15504 propõe modelos com duas dimensões: um conjunto de áreas de processo e uma seqüência de níveis de capacidade<sup>2</sup> de processo. Com a arquitetura contínua, cada organização pode escolher um perfil de capacidade de processo, composto por áreas de processo selecionadas, cada uma em um determinado nível de capacidade, para orientar a melhoria. A ISO/IEC 15504 já teve três versões [Dorling 1995, ISO/IEC TR 15504 1998, ISO/IEC 15504 2003] com o aumento da flexibilidade a cada versão. A ISO/IEC 15504 [2003] define um *framework* para modelos de capacidade de processo e processos para avaliação de processo e o modelo de capacidade de processo ISO/IEC 15504-5 para a engenharia de software.

O SW-CMM foi evoluído para um *framework* de modelos integrados (CMMI: *Capability Maturity Model Integration*) alinhado com a ISO/IEC 15504. Os modelos do CMMI são disponibilizados tanto na representação estagiada, como o SW-CMM, como na representação contínua, como a ISO/IEC 15504-5. Outros modelos de capacidade de processo relevantes são o FAA iCMM [Ibrahim et al. 2001], ITsqc eSCM-SP [Hyder et al. 2004a], PMI OPM3 [PMI 2003] e MPS.BR MR-MPS [Weber et al. 2005a].

Uma parte da comunidade tem defendido a arquitetura estagiada, como, por exemplo, Curtis [1998], Chrissis et al. [2003] e Weber et al. [2005b]; outra tem defendido a arquitetura contínua, como, por exemplo, Rout et al. [2000], Ibrahim [2000] e Shaeffer [2004]; e outra tem buscado diferentes formas de composição entre as duas arquiteturas, como, por exemplo, Ahern et al. [2001], Olson [2003] e Kasse [2004].

O cenário atual é composto pelo estabelecimento com sucesso da melhoria baseada nos modelos estagiados (SW-CMM e CMMI) e o reconhecimento das limitações naturais dessa abordagem, como, por exemplo, Palmer [1995], Conradi e Fuggetta [2002] e Rifkin [2002]. A arquitetura contínua ainda

---

<sup>2</sup> O termo capacidade, no contexto de capacidade de processo, é utilizado nessa tese como tradução do termo original em inglês *capability*, porque embora o termo *capability* signifique uma especialização do termo *capacity* (capacidade em português), não existe um termo específico em português para *capability*. Alguns autores utilizem o termo *capabilidade*, porem ele não é reconhecido pelos principais dicionários.

---

não está estabelecida no estado da prática, sendo uma potencialidade para evoluir as abordagens de melhoria de processo.

Apesar de ser uma força no estado da prática do setor de software, a área de melhoria de processo ainda não é um tópico popular de pesquisas mais rigorosas, principalmente nas universidades [Card 2004]. Como a área tem sido impulsionada pela prática, os diferentes modelos e abordagens são considerados concorrentes, embora compartilhem os mesmos princípios básicos. A insuficiência de pesquisas nessa área, tem limitado a identificação desses princípios básicos, que é papel da ciência.

## 1.2 Problemas e oportunidades

Esta pesquisa identifica que o não entendimento dos princípios básicos e a falta de apoio metodológico para utilização da flexibilidade oferecida pelos modelos com arquitetura contínua são os principais impedimentos técnicos para a evolução da melhoria de processo com o uso mais disseminado dos modelos contínuos. Identifica também que as principais limitações da atual melhoria de processo de software podem ser superadas com uma combinação adequada das arquiteturas contínua e estagiada.

Do ponto de vista do setor de software existe a demanda por uma melhor efetividade dos programas de melhoria de processo e de um melhor alinhamento ao contexto e objetivos estratégicos de cada organização, principalmente as micro e pequenas. Como os modelos com arquitetura estagiada definem níveis de maturidade fixos, para projetos de desenvolvimento de software sob encomenda, esses níveis podem não ser viáveis, relevantes e suficientes, ou mesmo não aplicáveis, a algumas organizações como, por exemplo, micro e pequenas organizações orientadas à evolução e comercialização de produtos. Também tem aumentado a quantidade de organizações que utilizam elementos de mais de um modelo como referência para a melhoria como, por exemplo, os modelos do CMMI, ISO 9001 [2000] e PMBOK [PMI 2004].

A melhoria de processo de software pode ser vista como uma abordagem para estabelecer os processos para uma engenharia de software e, a partir desses processos, viabilizar o estabelecimento de uma engenharia de software mais eficiente e eficaz, ou mesmo um novo paradigma para produção de software. Enquanto parte da comunidade mantém a busca, iniciada em 1968, pela engenharia de software como, por exemplo, Shaw [1990], Humphrey [1999], Capps [2003], Abran et al. [2004] e Jackson [2004]; outra parte questiona o uso da engenharia para software como, por exemplo, Armour [2003] e Cockburn [2004]; e outra parte foca mais na comunidade da prática para entender melhor esta situação como, por exemplo, Coplien [1995], Jones [2003] e Glass [2003]. A melhoria de processo de

software, que estuda a prática da engenharia de software sob uma referência conceitual baseada no autores da qualidade na manufatura, principalmente Shewhart, Deming, Juran e Crosby [apud Humphrey 1989, Chrissis et al. 2003], estabelecida com os modelos estagiados e com a potencialidade dos modelos contínuos, pode estabelecer conexões entre essas três partes da comunidade.

### 1.3 Objetivos e propostas da pesquisa

Esta pesquisa foi iniciada com o objetivo de desenvolver apoio metodológico para a escolha de processos do modelo ISO/IEC TR 15504-5 para orientar a melhoria de uma organização intensiva em software, alinhada ao contexto e objetivos estratégicos dessa organização. Com o desenvolvimento e utilização desse apoio metodológico e com a evolução da área esse objetivo foi ampliado. O objetivo principal da pesquisa passou a ser explorar as possibilidades de flexibilidade oferecidas pela arquitetura contínua preservando as vantagens da arquitetura estagiada. Esse objetivo principal foi desdobrado em cinco objetivos relacionados [Figura 1].

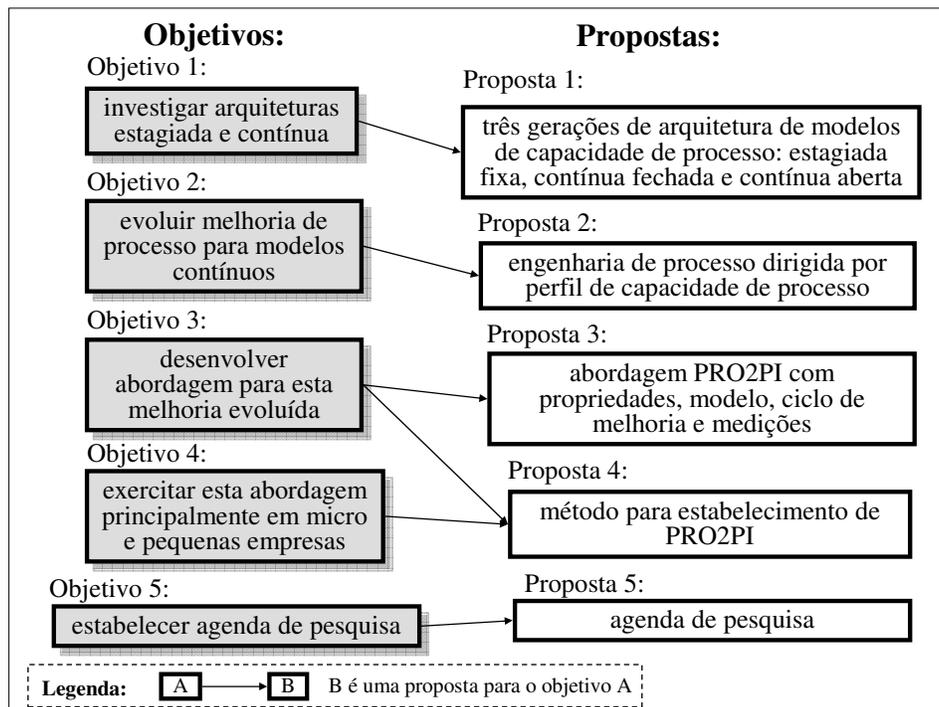


Figura 1 – Objetivos e propostas da pesquisa

O objetivo 1 é entender melhor as semelhanças, diferenças e relacionamentos das arquiteturas estagiada e contínua de modelos de capacidade de processo e identificar os princípios básicos unificadores das arquiteturas. O objetivo 2 é evoluir a abordagem corrente de melhoria de processo de

---

software, com o desenvolvimento de apoio metodológico para a utilização da flexibilidade dos modelos contínuos para a melhoria de processo orientada ao contexto e objetivos estratégicos de uma organização. O objetivo 3 é desenvolver uma abordagem para essa evolução da melhoria de processo. O objetivo 4 é experimentar essa nova abordagem principalmente para início de um programa de melhoria em micro e pequenas organizações intensivas em software. O objetivo 5 é estabelecer uma agenda para continuação das pesquisas.

O objetivo 1 é tratado com a identificação de três gerações de arquiteturas de modelos de capacidade de processo denominadas, respectivamente, de estagiada fechada, contínua fechada e contínua aberta, como substituição da tradicional classificação entre arquiteturas estagiada e contínua. O objetivo 2 é tratado com uma proposta de uma engenharia de processo dirigida por perfis de capacidade de processo como uma evolução da abordagem corrente de melhoria de processo de software. O objetivo 3 é tratado com uma proposta de uma abordagem para melhoria de processo dirigida por perfis de capacidade de processo (PRO2PI – *Process Capability Profile to Process Improvement*<sup>3</sup>) por meio de seus quatro elementos: as propriedades de um perfil, um modelo para integração de elementos de várias fontes, um ciclo para melhoria com a abordagem e um conjunto de medições para o estabelecimento de perfis. O objetivo 4 é tratado com a proposta de um método para oficinas de estabelecimento de PRO2PI como início do ciclo de melhoria em uma micro ou pequena organização e como um processo de treinamento em melhoria de processo. O objetivo 5 é tratado com o início de uma série de pesquisas como continuação e desdobramento das aplicações realizadas e a proposta de uma agenda de pesquisa.

## 1.4 Metodologia e processo da pesquisa

O processo dessa pesquisa foi orientado por uma metodologia baseada na abordagem ‘indústria-como-laboratório’ [Potts 1993] com em três fases (desenvolvimento, caracterização e formalização) cada uma com ciclos de exploração, aplicação e consolidação [Figura 2].

---

<sup>3</sup> Na sigla PRO2PI o 2 representa a repetição de PRO (*PROcess Capability PROfile*) e também o termo *to (para) como corruptela do inglês ‘two’*, e o PI representa *Process Improvement*.

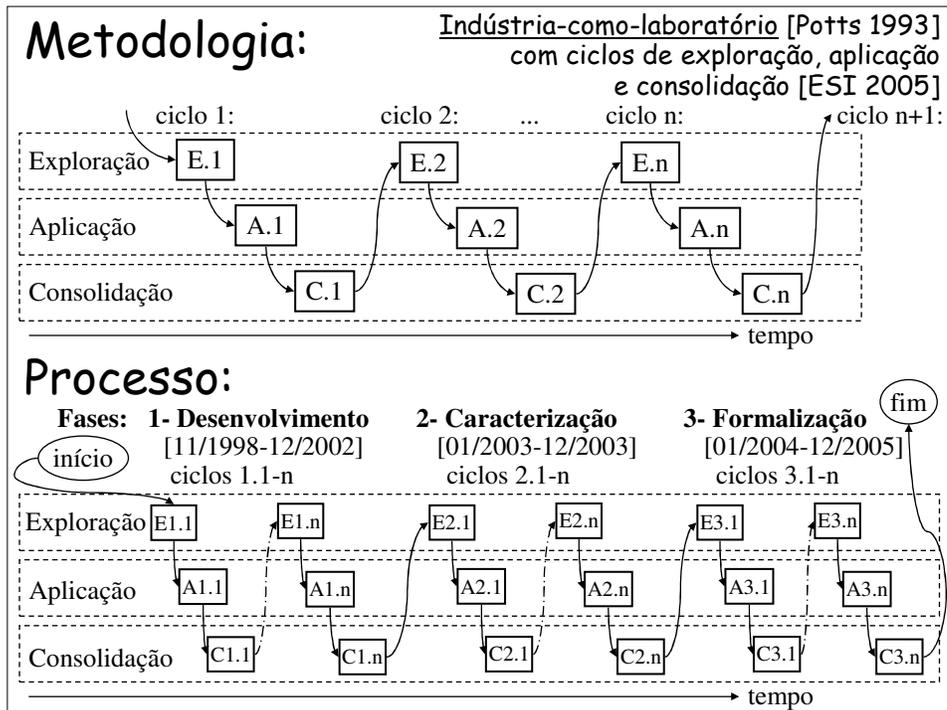


Figura 2 – Metodologia e processo dessa pesquisa

Para pesquisas em engenharia de software, Potts defende a abordagem ‘indústria-como-laboratório’ (*industry-as-laboratory*) como uma abordagem complementar á tradicional ‘pesquisa-depois-transfere’ (*research-then-transfer*). Na abordagem indústria-como-laboratório, as idéias para pesquisa são baseadas em problemas práticos e são refinadas em estudos de casos contínuos e incrementais. Com isto, segundo Potts, “os domínios de pesquisa e aplicação tendem a convergirem, e a transferência de tecnologia e o processo de avaliação são realizados nos estágios iniciais do programa de pesquisa” [Potts 1993, p. 22]. Essa abordagem indústria-como-laboratório tem sido utilizada, por exemplo, em projetos do *Embedded Systems Institute* (ESI), para confrontar pesquisa acadêmica com requisitos e restrições da indústria, por meio de ciclos de exploração, aplicação e consolidação [ESI 2005].

O processo desta pesquisa foi realizado em três fases: desenvolvimento, caracterização e formalização. A primeira fase, chamada de desenvolvimento, foi realizada de novembro de 1998 a outubro de 2002 e focou no desenvolvimento e aplicação de uma Abordagem para Melhoria de

---

Processo (AMP1<sup>4</sup>), baseada no ciclo de melhoria da ISO/IEC TR 15504-7 [1998] e com a utilização do modelo contínuo da ISO/IEC TR 15504-5 [1998]. Para essa abordagem foi desenvolvido também um Método para Escolha de Processos (MEP1<sup>5</sup>) para orientar uma melhoria alinhada ao contexto e objetivos estratégicos da organização. O principal marco dessa fase foi o seu início, com o desenvolvimento da estrutura da abordagem e apresentação para empresas de Blumenau, onde uma das empresas se interessou pelo trabalho. AMP1 e MEP1 foram então especificados e aplicados no período de março a junho de 1999 [Salviano et al. 1999]. Depois houve a continuação do trabalho [Nicoletti e Salviano 2003], o detalhamento do método MEP1 [Salviano 2001] e novas aplicações entre 1999 a 2003 [Salviano e Souza 2000, Silva et al. 2003].

A segunda fase, denominada de caracterização, foi realizada de setembro de 2003 a outubro de 2004. O principal marco dessa fase foi o desenvolvimento do artigo para a conferência SPICE 2004, cuja submissão foi em novembro de 2003, a revisão e conclusão da versão final em dezembro de 2003 e apresentação e publicação em abril de 2004 [Salviano et al. 2004]. Este artigo contém a definição das gerações de arquitetura de modelos e a primeira versão de PRO2PI. Houve uma descrição dessas propostas também para a comunidade do CMMI [Salviano e Jino 2004].

A terceira fase, denominada de formalização, foi realizada de novembro de 2004 a outubro de 2005. Nessa fase, PRO2PI foi consolidado, como um exemplo de abordagem para uma engenharia de processo dirigida por perfis de capacidade de processo, e composta por quatro elementos: um conjunto de propriedades para caracterizar um perfil, um modelo para integrar as diferentes fontes para um perfil, um ciclo de melhoria orientado a um perfil e um conjunto de medições. Para apoiar a utilização de PRO2PI foi também desenvolvido e utilizado um método para o estabelecimento de PRO2PI.

Essa terceira fase incluiu também o desenvolvimento de outros projetos de pesquisa com participação, direta ou indireta, das propostas de PRO2PI, entre eles, Lobo e Salviano [2005], Weber et al. [2005a] e von Wangenheim e Salviano [2005], e o início de outros projetos, com a participação do autor, entre eles, Cavalcante e Costa [2005], Miranda [2005], Ferreira [2005] e de Petri et al. [2005], e mesmo sem a participação do autor, entre eles, Faria [2005] e Bernardo [2005]. Um balanço

---

<sup>4</sup> Esta identificação da abordagem como AMP1 foi definida neste trabalho para facilitar a referência à abordagem. A abordagem foi referenciada nas publicações originais como abordagem CTI, abordagem ITI e abordagem CenPRA, dependendo do nome corrente da instituição na qual ele foi desenvolvido.

<sup>5</sup> Esta identificação MEP1 foi definida neste trabalho para facilitar a referência ao método, que não tinha sido nomeado nas publicações originais.

das experiências com micro e pequenas empresas foi realizado [von Wangenheim et al. 2006a] juntamente com uma comparação com experiências semelhantes na Finlândia [von Wangenheim et al. 2006b].

## **1.5 Estrutura da tese**

Esta tese está organizada em oito capítulos e três apêndices, dos quais esse é o primeiro capítulo. No Capítulo 2 é apresentado o estado da prática e estado da arte da área de melhoria de processo de software, incluindo modelos e abordagens para melhoria de processo de software.

Os capítulos 3 a 6 descrevem as principais contribuições da tese. No Capítulo 3 são descritos os resultados iniciais da pesquisa e uma identificação de três gerações de arquiteturas de modelos de capacidade de processo. No Capítulo 4 é descrita uma proposta de uma engenharia de processo dirigida por perfis de capacidade de processo. No Capítulo 5 é descrita uma proposta de uma abordagem para melhoria de processo dirigida por perfis de capacidade de processo (PRO2PI) que inclui um conjunto de propriedades, um modelo, um ciclo de melhoria e medições. No Capítulo 6 é descrita a proposta de um método para oficinas de estabelecimento de PRO2PI.

No Capítulo 7 são descritas as experiências de utilização das várias versões da metodologia e uma análise em relação à versão atual. No Capítulo 8 é apresentada a conclusão do trabalho com resumo das contribuições, relacionamentos com outros trabalhos, limitações, agenda para trabalhos futuros de pesquisa e considerações finais.

Nos três apêndices são descritos respectivamente o diagrama de classes do modelo de PRO2PI, uma medição de complexidade de um ciclo de melhoria e as fases e atividades do método para oficinas de estabelecimento de PRO2PI.

# Capítulo 2

## 2 Melhoria de processo de software

O objetivo deste capítulo é apresentar uma visão do estado da arte e estado da prática da área de melhoria de processo de software para subsidiar o entendimento das propostas descritas nesta tese. Para tanto, este capítulo apresenta:

- considerações sobre a engenharia de software e a melhoria de processo de software como um detalhamento e desdobramento da apresentação e da metodologia de pesquisa, descritas de forma resumida no capítulo anterior;
- um vocabulário e conceitos básicos da área de melhoria de processo de software;
- os modelos e frameworks de modelos mais relevantes utilizados como referência em programas de melhoria de processo de software;
- as abordagens mais relevantes utilizadas para esses programas de melhoria de processo de software;
- outros modelos e abordagens relevantes para as propostas desta pesquisa; e
- tendências e propostas mais relevantes para esta pesquisa, particularmente aquelas sobre os relacionamentos e integração dos modelos mais relevantes, comparações e combinações das arquiteturas estagiada e contínua, e abordagens para escolha de processos de modelos com arquitetura contínua.

### 2.1 Engenharia de software e melhoria de processo de software

O termo Engenharia de Software (*Software Engineering*) foi definido pela primeira vez na conferência NATO 68 [Naur e Randell 1969]. Em 1967 um grupo de estudos em ciência da computação foi estabelecido pelo comitê científico da NATO (*North Atlantic Treat Organization*) para discutir os problemas do software (Figura 3).

Naquela reunião foi então proposta a realização de uma conferência de trabalho em engenharia de software. “O termo *engenharia de software* foi deliberadamente escolhido como uma provocação, para implicar na necessidade da construção de software ser baseada nos moldes dos fundamentos

teóricos e disciplinas práticas, que são tradicionais nos tipos de engenharia já estabelecidos” [Naur e Randell 1969, p. 8].

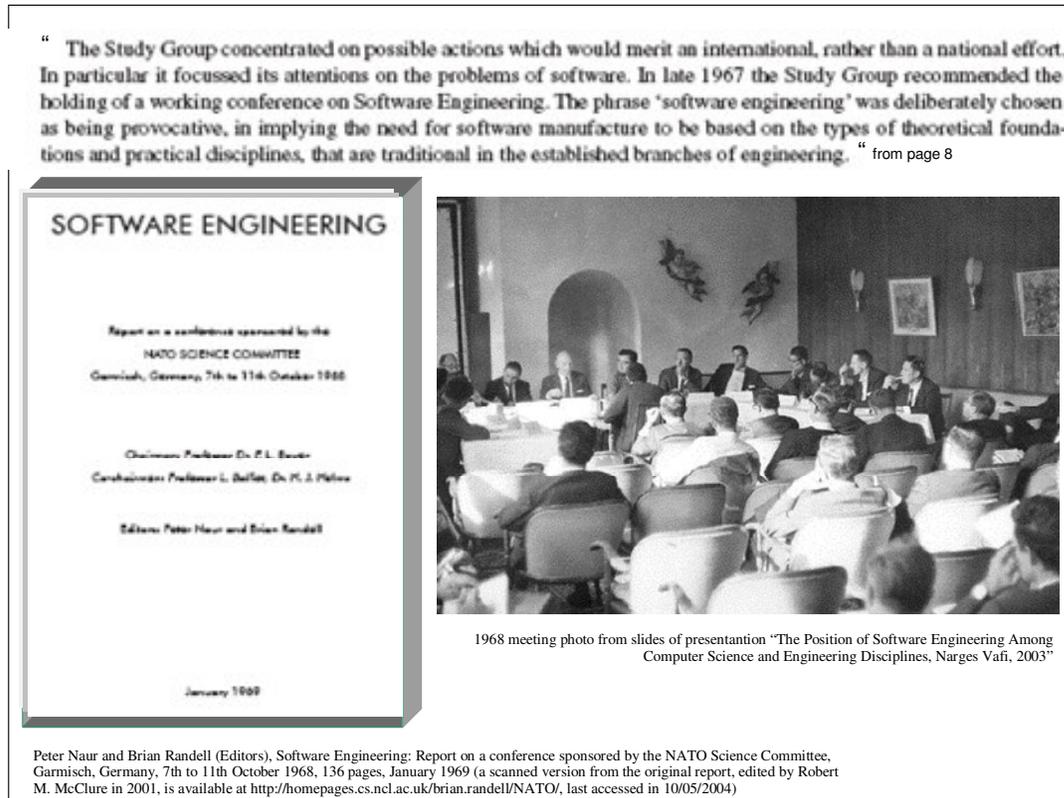


Figura 3 – “Nascimento” da Engenharia de Software

Desde então a comunidade tem procurado entender a natureza das atividades de software e sua relação com a engenharia. Várias técnicas, modelos e conceitos têm sido desenvolvidas mas um problema ainda permanece: a distância entre teoria e prática. A criação do SEI (*Software Engineering Institute*) em 1980 com a meta de “prover liderança no avanço do estado da prática de engenharia de software para melhorar a qualidade de sistemas que dependem de software [Paulk et al. 1994] e um balanço mais recente realizado na edição de novembro de 2003 da revista *IEEE Software* (IEEE: *The Institute of Electrical and Electronics Engineers*), sobre o estado da prática [Glass 2003], evidenciam essa distância entre teoria e prática. Glass critica aqueles que vêem o estado da prática como largamente caótico e sem sucesso, e reconhece que muitos pesquisadores acadêmicos não sabem o que ocorre na prática da engenharia de software e que, o que ele considera como pior ainda, acham que sabem.

Gary Pollice [2005] comenta que “nossa disciplina [engenharia de software ou desenvolvimento de software] amadureceu e avançou na última metade de século, mas alguma coisa ainda falta. Muito,

---

senão a maioria, do software que criamos é ainda produzido, não exatamente engenhariado” [Pollice 2005, p.1]. Ainda segundo Pollice, não existe consenso se seus alunos de “engenharia de software” estão aprendendo engenharia de software. Os alunos estão aprendendo a serem membros produtivos de uma equipe de desenvolvimento de software, com algum conhecimento de processo, como fazer o processo funcionar para a equipe, como escrever software que satisfaz seus requisitos e como testa-lo, e como lidar com as mudanças constantes que são parte do mundo real. Por tudo isto, o que temos hoje é desenvolvimento de software, e não, pelo menos ainda, engenharia de software.

Mesmo a utilização de engenharia para software tem sido contestada. Cockburn propõe o fim da engenharia de software e o início de jogos de cooperação econômica como um modelo mais apropriado para software [Cockburn 2004]. Armour defende que a visão de software como um produto tem sido provavelmente um erro e que software deve considerado como uma mídia na qual armazenamos conhecimento [Armour 2003].

Brooks foi um dos primeiros a reconhecer, em 1975 no livro “*The mythical man-month*”, a importância da gerência de projetos na engenharia de software [Brooks 1975] e também identificou a inerente complexidade das atividades de software e conseqüentemente da impossibilidade de soluções mágicas (“*no silver bullet*”) para a engenharia de software [Brooks 1986]. Na edição comemorativa de vinte anos da publicação de “*The mythical man-month*”, Brooks, e a comunidade de software, confirmou a validade das suas principais constatações, com uma análise da situação corrente e a repercussão das duas publicações [Brooks 1995].

Mary Shaw estudou diversas definições de engenharia, suas relações com a engenharia de software, e conclui que essas definições compartilham um conjunto de cláusulas [Shaw 1990]. Essas cláusulas foram sintetizadas e comentadas em relação à engenharia de software. Segundo Shaw, engenharia envolve a:

*“criação de soluções eficientes* (engenharia não é apenas sobre solução de problemas, mas sim sobre solução de problemas com um uso econômico de todos os recursos, inclusive recursos monetários, financeiros e temporais) *para problemas práticos* (engenharia trata de problemas práticos, cujas soluções interessam a pessoas fora do domínio da engenharia, sendo que essas pessoas são os clientes da engenharia) *por meio da aplicação de conhecimento científico* (engenharia resolve problemas de uma maneira particular: pela aplicação de ciência, matemática, análise e projeto) *construindo coisas* (engenharia enfatiza soluções que são usualmente objetos concretos, apesar que a engenharia de software, por exemplo, tem alterado isto ao

ênfatisar soluções abstratas que são os sistemas de software) *a serviço da condição humana* (engenharia não serve apenas para o cliente imediato, mas também desenvolve tecnologia e conhecimento para a sociedade de modo geral”.

Para atingir esse objetivo, uma engenharia utiliza conhecimentos científicos sobre domínios tecnológicos que estão codificados de uma forma que seja diretamente útil para um engenheiro. Deste modo esse conhecimento codificado provê respostas para questões que ocorrem com frequência na prática. Ou seja, esse conhecimento deve ser reutilizado para a geração de soluções.

O guia para o corpo de conhecimento da engenharia de software SWEBOK (*Guide to the Software Engineering Body of Knowledge*), é uma caracterização, validada por consenso, da disciplina de engenharia de software e um guia para o corpo do conhecimento que apoia essa disciplina [Abran et al. 2004]. O SWEBOK adotou a definição IEEE 610.12, que define engenharia de software como: “(i) a aplicação de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção de software; ou seja, a aplicação de engenharia a software, (ii) o estudo das abordagens como em (i)”.

A utilização mais sistemática de processos, foi iniciada nos anos 1930. Nessa época, Walter Shewhart iniciou um trabalho em melhoria de processo com os princípios do controle estatístico. Esses princípios foram refinados por W. Edwards Deming e Joseph Juran, sempre focados na indústria de manufatura. Nos anos 1970, Philip Crosby notou que as organizações de manufatura com as quais ele trabalhou poderiam ser estudadas de acordo com a qualidade de sua produção e definiu cinco estágios sequenciais e cumulativos do processo de produção, baseados principalmente nas atitudes gerenciais encontradas em cada estágio. Esses estágios indicam a qualidade do processo de produção. Nos anos 1980, Ron Radice adaptou esses princípios de controle estatísticos e os estágios de qualidade ao desenvolvimento de software. Em seguida, Watts Humphrey estendeu esse trabalho e definiu cinco níveis de maturidade [parágrafo adaptado de Chrissis et al. 2003].

Alfonso Fuggetta ao concluir uma breve visão geral da história e resultados da pesquisa em processo de software reconheceu que “a utilização de processos para tratar a inerente complexidade de software ganhou força a partir dos anos 1980” e enfatizou que “a visão do desenvolvimento de software como um processo tem ajudado significativamente a identificação das diferentes dimensões do desenvolvimento de software e os problemas que devem ser tratados para estabelecer práticas efetivas” [Fuggetta 2000]. Ainda segundo Fuggetta, “nós temos de prestar atenção para a complexa interpelação de numerosos fatores organizacionais, culturais, tecnológicos e econômicos [do desenvolvimento de software]”.

Na área de pesquisa em processos de software, um marco importante é o reconhecimento que processos de software são software também [Osterweil 1987, Osterweil 1997].

A melhoria de processo de software tem mostrado na prática ser uma abordagem viável, eficaz e eficiente para a necessária melhoria das organizações intensivas em software. A comunidade tem relatado vários casos de sucesso, como, por exemplo, Herbsleb et al. [1994], DACS [1999] e Card [2002].

As abordagens para melhoria de processo, como, por exemplo, a abordagem IDEAL [McFeeley 1996], a abordagem para Melhoria de Processo da ISO/IEC 15504 [ISO/IEC TR 15504-7 1998, ISO/IEC 15504-4 2004], a abordagem baseada em problemas e metas [Porter e Sakry 2002] e as orientações para a melhoria [O'Toole 2000], utilizam como referência um modelo de processo que sistematiza e representa as melhores práticas, definem uma medição para avaliação da capacidade dos processos e provê um roteiro racional para a melhoria dos processos. Exemplos de modelos mais utilizados são o SW-CMM versão 1.1 (*Capability Maturity Model for Software*) [Paulk et al. 1994], a Norma ISO/IEC 12207 [1998], a Norma ISO/IEC 15504 (também conhecida como SPICE: *Software Process Improvement and Capability Determination*) [ISO/IEC 15504 1998], a evolução do SW-CMM lançada em 2000 o CMMI (*Capability Maturity Model Integration*) [Chrissis et al. 2003], e a aplicação para software da ISO 9000, principalmente a versão 2000 com o par coerente 9001 e 9004 [ISO 9001 2000, ISO 9004 2000]. Essas metodologias e modelos estão bem difundidas na comunidade e descritas em várias publicações [por exemplo, Rocha et al. 2001], inclusive com considerações sobre o relacionamento entre os modelos [Sheard 2001].

Uma forte tendência atual é a ampliação da abrangência da melhoria de processo de software para a melhoria de processo de sistemas. Os dois principais modelos de capacidade de processo de software (SW-CMM e ISO/IEC TR 15504) foram evoluídos e suas novas versões (CMMI e ISO/IEC 15504) tratam de processos de sistema, incluindo processos de software.

Melhoria de processo de software e sistema é parte da disciplina de engenharia de processo de sistema e software, que, por sua vez, é ao mesmo tempo uma disciplina autônoma e parte da disciplina de engenharia de software e sistemas. Existe uma tendência atual de generalizar as atividades de software para sistemas, cujos componentes podem ser implementados por diferentes engenharias, como, por exemplo, elétrica, eletrônica, e software. Com isto podemos estender (i) processos da engenharia de software para processos da engenharia de sistemas; e (ii) engenharia de processo de software para engenharia de processo de sistemas

Os modelos da ISO/IEC 15504 e CMMI são atualmente os principais modelos de referência para avaliação e melhoria dos processos de software e sistema. Outras normas, como, por exemplo, a Norma NBR ISO/IEC 12207 tratam de outros aspectos da engenharia de processo de software. A Norma ISO/IEC 12207 - Processos de Ciclo de Vida de Software [NBR 1998, ISO 2002, Machado 2003] tem como objetivo principal o estabelecimento de uma estrutura comum para os processos de ciclo de vida de software como forma de ajudar as organizações a compreender todos os componentes presentes na aquisição e fornecimento de software e, assim, conseguir firmar contratos e executar projetos de forma mais eficaz.

A utilização de modelos para a melhoria não é suficiente para garantir o sucesso. Para a obtenção da melhoria, alguns fatores têm sido reconhecidos como fundamentais. Existe uma estatística informal que circula pela comunidade que, de cada três iniciativas de melhoria de processo, apenas uma tem sucesso. Portanto é fundamental reconhecer as razões desses resultados. Na nossa opinião, baseada em compilações de fontes variadas [por exemplo, Paulk 1994, O'Tolle 2000] e observações próprias, os pontos relacionados a seguir, sem ordem de importância, são os principais fatores para o sucesso dos esforços para a melhoria:

- entendimento das características, limitações e implicações da utilização da abordagem baseada em melhoria de processo;
- escolha, entendimento, utilização e interpretação de um bom modelo de processo ou de um conjunto de modelos;
- alinhamento dos esforços de melhoria com o contexto e objetivos estratégicos da organização;
- estabelecimento de metas relevantes, viáveis e mensuráveis;
- considerações aos aspectos gerenciais, técnicos e humanos da melhoria;
- comprometimento da alta gerência, condução por empreendedores e participação de todos;
- condução dos esforços de melhoria como um programa baseado em uma abordagem; e
- conhecimento do processo atual.

A área de melhoria e avaliação de processo de software está tendo um grande destaque atualmente, em grande parte, pelo sucesso e reconhecimento do conjunto de ações realizadas em função do modelo SW-CMM (*Capability Maturity Model for Software*). A partir do lançamento e divulgação da versão 1.1 em 1993, o tema da melhoria de processo ganhou força na comunidade. Essa força foi consequência dos resultados práticos obtidos pelas organizações que realizaram programas de melhoria com o SW-CMM como modelo de referência. O SW-CMM organizou grande parte do

---

conhecimento da engenharia software em um modelo, baseado em um conjunto de premissas. Essas premissas incluem:

- As organizações de software estão sobrecarregadas de trabalho, reagindo a crises constantes, tentando fazer hoje o que deveria já estar pronto, trabalhando de forma reativa, conhecida como “apagando incêndios”;
- Os maiores problemas nas organizações de software são de natureza gerencial e não de natureza técnica;
- Sem uma disciplina de gerenciamento bem estabelecida, as atividades do processo de engenharia são as primeiras a serem sacrificadas;
- Os princípios da qualidade, definidos por autores como Juran e Deming, que revolucionaram a indústria de manufatura, também podem ser aplicados com sucesso na produção de software;
- Boas práticas existem em algumas organizações de software, mas estas boas práticas são pouco disseminadas; e
- A qualidade de um software é altamente influenciada pela qualidade do processo utilizado no desenvolvimento e manutenção.

Estas premissas apontavam para soluções que em um primeiro momento focassem na utilização de princípios básicos de gerência de projeto para “arrumar a casa”, gerar resultados imediatos e preparar a organização para as próximas etapas da melhoria. Sem uma gerência de projetos bem estabelecida, o risco de qualquer outra iniciativa não produzir os resultados esperados é muito grande.

## 2.2 Vocabulário e conceitos básicos

Essa tese identifica vinte e oito conceitos básicos e os principais relacionamentos entre esses conceitos como uma proposta para caracterizar a visão corrente da área de melhoria de processo de software [Figura 4].

Os vinte e oito conceitos e seus relacionamentos identificados na Figura 4 estão definidos e comentados a seguir, agrupados em subseções. A primeira vez que cada conceito é referenciado no início de sua definição, o conceito é escrito em **negrito**.



Essa pesquisa estabelece as seguintes definições para processo e processo de software. Processo é o que as pessoas fazem para um determinado propósito, utilizando suas habilidades e conhecimento, com o apoio de artefatos, ferramentas e outros recursos. Processo de software é o que as pessoas fazem para um determinado propósito, utilizando suas habilidades e conhecimento, com o apoio de artefatos, ferramentas e outros recursos, para produzir software e seus produtos associados.

**Representação de processo**, que também é conhecido por documentação de processo, modelo de processo e desenho de processo, é uma representação das principais características e elementos de um processo, em uma determinada notação. Uma das notações utilizadas na comunidade é a notação ETVX (*Entry criteria, Task, Verification and eXit criteria*; Critério de entrada, tarefa, verificação e critério de saída) [Radice e Phillips 1988]. Em ETVX, um processo é representado pelo propósito, entrada, critérios de entrada, atores, atividades, verificação, saída e critério de saída, segundo uma representação gráfica semelhante à ilustrada na Figura 5.

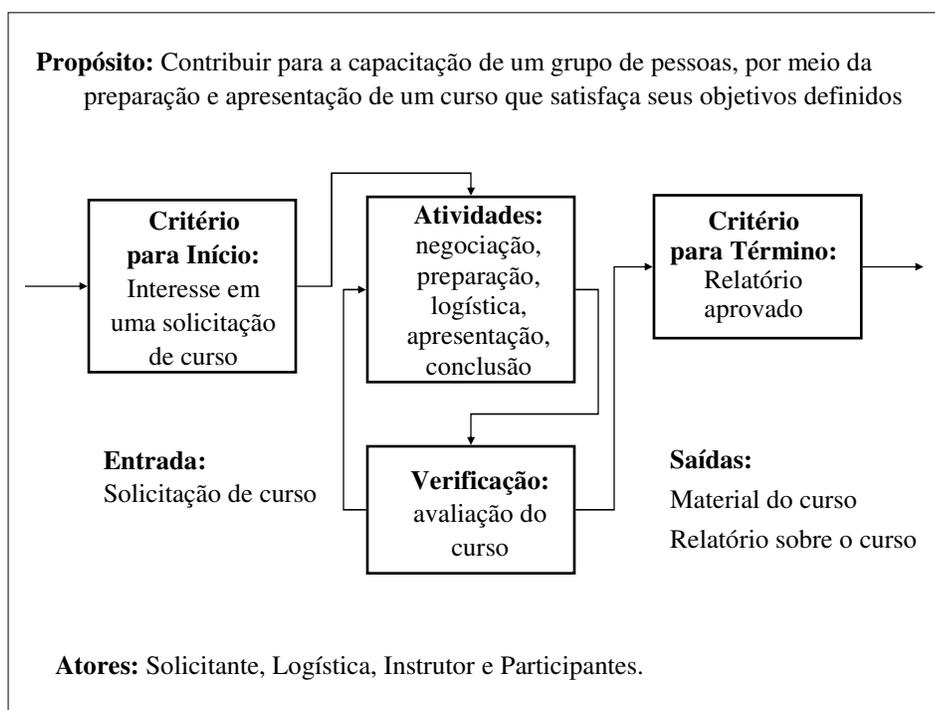


Figura 5 – Exemplo de representação de processo em ETVX

A Figura 5 ilustra como exemplo a representação de um processo para negociação, preparação, logística, apresentação e conclusão de cursos.

Segundo Olson et al. [1994] uma representação de um processo deve responder a oito questões chave, que estão relacionadas na Tabela 6 juntamente com o elemento da representação que tipicamente contém a resposta.

Tabela 6 – Perguntas chave e elementos que as respondem

Pergunta chave:	Elemento da representação de processo:
Por quê a atividade é executada?	1. Propósito
Quem realiza a atividade?	2. Atores, papéis
Quais artefatos são utilizados?	3. Entradas
Quais artefatos são produzidos	4. Saídas
Quando a atividade é iniciada?	5. Critérios de entrada
Quando a atividade é terminada?	6. Critérios de saída
Como a atividade é implementada?	7. Procedimento, método
Qual a seqüência das atividades?	8. Fluxo
Onde a atividade é realizada?	9. Contexto

O SPEM (*Software Process Engineering Metamodel*, metamodelo de engenharia de processo de software) [OMG 2002b] é um metamodelo de notações para representação de processo. A Figura 6 ilustra, como um diagrama UML (*Unified Modeling Language*, Linguagem de Modelagem Unificada) [OMG 2003a], a essência da representação de processo do SPEM.

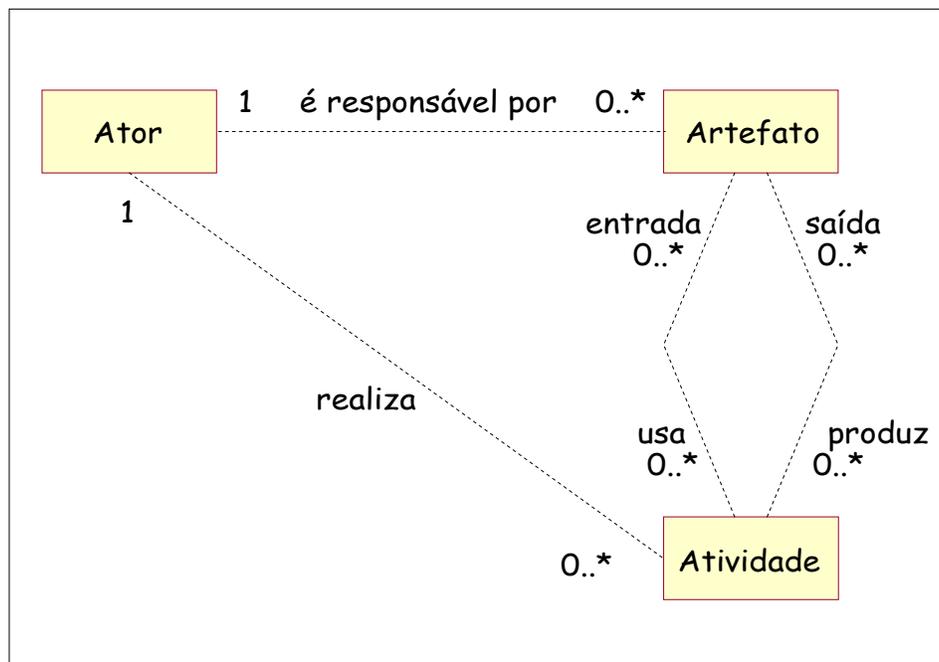


Figura 6 – Essência da representação de processo do SPEM em UML

Conforme ilustrado na Figura 6 a essência de uma representação de processo é definida no SPEM por meio de três entidades (ator, artefato e atividade) e seus relacionamentos. Um ator é responsável por zero ou mais artefatos e realiza zero ou mais atividades. Cada atividade usa zero ou mais artefatos como entrada e produz zero ou mais artefatos como saída. Cada artefato é usado como entrada por zero ou mais atividades e produzido como saída por zero ou mais atividades.

Processos são organizados em arquiteturas. **Arquitetura de processo** é uma estrutura para os tipos de elementos de processo e uma caracterização desses tipos de elementos. Uma arquitetura de processo bastante utilizada na área é a descrita por Olson et al. [1994] e outra arquitetura semelhante é descrita por O’Toole [2000] [Figura 7].

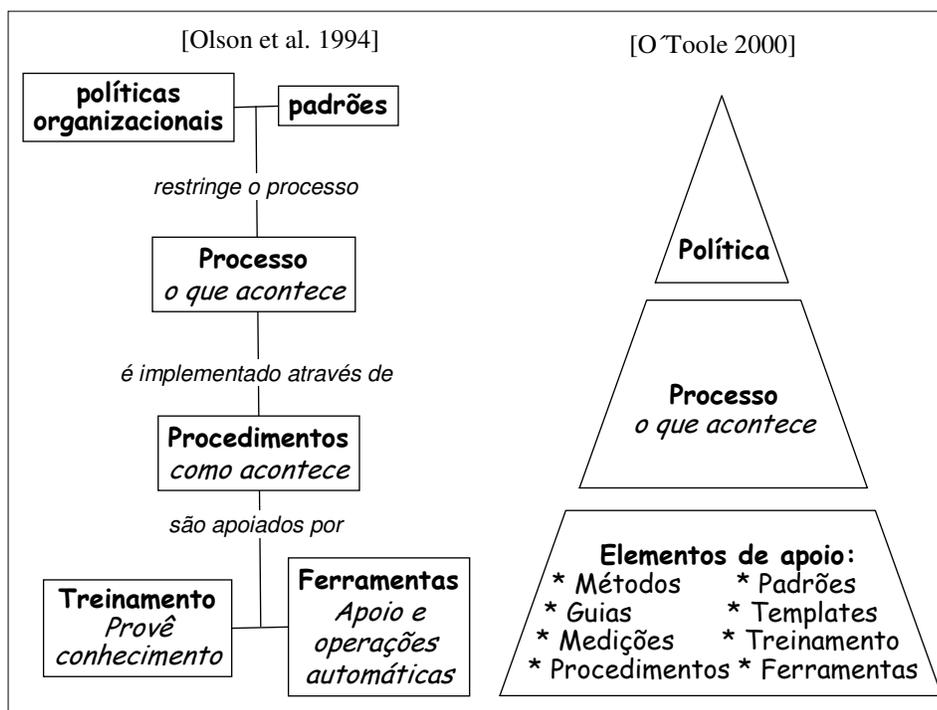


Figura 7 – Duas propostas semelhantes de arquitetura de processo

As duas propostas de arquitetura de processo ressaltam a caracterização de políticas, processos e procedimentos, como três níveis hierárquicos. Política descreve uma orientação geral para a organização. Processo descreve o que é esperado que seja feito, enquanto procedimento descreve como é esperado que seja feito. Segundo O’Toole, “esta distinção é muitas vezes perdida porque a representação de processos e procedimentos tipicamente incluem vários elementos iguais: propósito, papéis, entradas, critério de saída, atividades/etapas, saída, critério de conclusão, etc. A diferença real está nos graus de liberdade da representação dos componentes de processo ou procedimento”. O’Toole propõe um quarto nível, onde os elementos de apoio podem ser adaptados para cada unidade da organização.

Olson [2004] descreve uma série de orientações para a definição de processos, curtos e usáveis. *Information Technology* é uma técnica para aprendizagem e representação de informação baseada nos estudos de Horn da ciência cognitiva e um método estruturado para desenvolver e representar informações [Infomap 2005]. É baseado em sete princípios: controle de acesso ao detalhe, blocos,

consistência, hierarquia, gráficos integrados, identificação e relevância. Esta técnica identifica uma arquitetura com sete tipos de informações: princípios, processos, procedimentos, estruturas, conceitos, fatos e classificação. O conceito de princípio é semelhante ao conceito de políticas das arquiteturas de Olson e O'Toole, e novamente é realçado a relação hierárquica entre política (no caso princípios), processos e procedimentos.

### **2.2.2 Área de melhoria de processo de software**

Esta área tem sido geralmente denominada de Melhoria de Processo de Software (MPS) [e.g., Zahran 1998, Caputo 1998, ISO/IEC 15504-5 2006]. Também tem sido utilizada a denominação mais extensa e mais precisa de Melhoria de Processo de Software Baseada em Modelos de Maturidade da Capacidade de Processo [e.g., Ahern et al. 2001].

O conceito de **melhoria (ou engenharia) de processo de software** é definido neste trabalho como: uma abordagem para a melhoria de organizações intensivas em software, baseada em modelos de capacidade de processo de software, por meio do estabelecimento, avaliação e melhoria da capacidade de seus processos mais importantes, relacionados às atividades de aquisição, fornecimento, operação, desenvolvimento, manutenção, gerência, melhoria e/ou apoio de sistemas de software, com o objetivo de satisfazer de forma mais eficiente e eficaz os objetivos estratégicos da organização. Normalmente os objetivos estratégicos são uma seleção e composição de fatores como controle e redução dos prazos, custos, e outros recursos para o desenvolvimento, manutenção e operação de sistemas intensivos em software, que satisfaçam os usuários e a comunidade envolvida, com um número mínimo de erros na execução desses sistemas.

Outra denominação para a melhoria de processo de software que tem sido utilizada é a de Engenharia de Processo de Software [e.g., Kinnula 1999, Croll 2003, Khaled El Emam 2004].

Kinnula propôs um modelo de arquitetura para um sistema de Engenharia de Processo de Software composto por três submodelos: hierarquia, infra-estrutura e modelo de processo. Esse modelo é ilustrado na Figura 8.

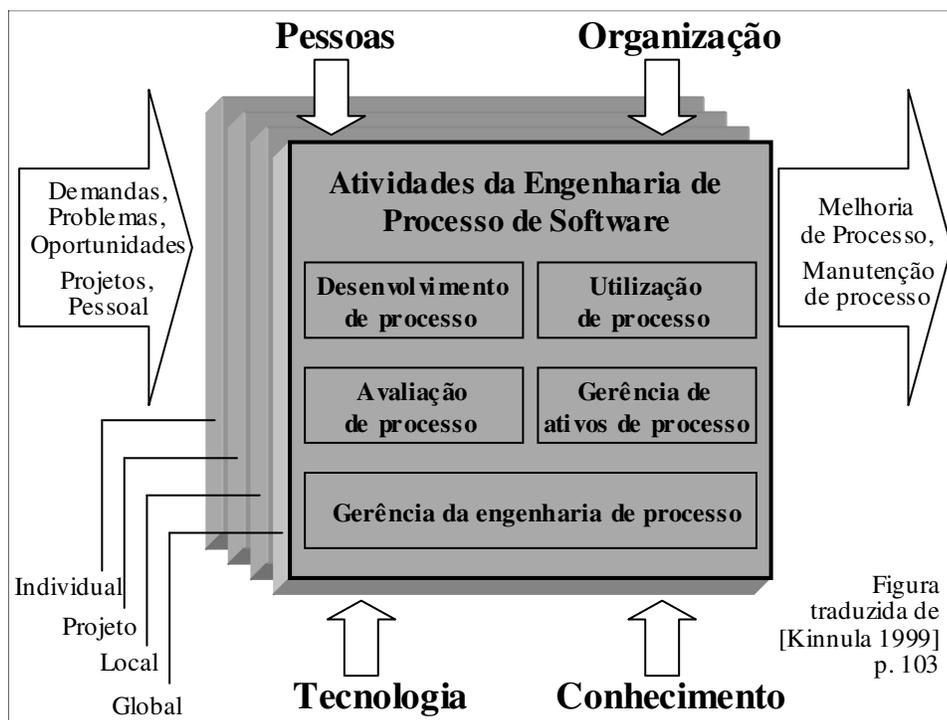


Figura 8 - Modelo de Arquitetura para o Sistema de EPS [Kinnula 1999]

Engenharia de Processo de Software, segundo Kinnula, é uma forma de aprendizagem organizacional que cobre o domínio de processos de software e que inclui os seguintes cinco processos:

- desenvolvimento de processo, com o propósito de projetar, implementar, refinar e testar uma mudança em um processo para melhorar a capacidade do processo em atingir as metas definidas do processo;
- utilização de processo, com o propósito de estabelecer uma instância de um processo por meio da utilização de ativos existentes para o processo e a realização dele na prática;
- avaliação de processo, com o propósito de coletar, arquivar, estabelecer uma linha básica (*baseline*), analisar e validar a corretude dos dados necessários para a engenharia de processo;
- gerência de ativos de processo, com o propósito de captar o conhecimento e tecnologia existentes, e as melhorias para os processos em um conjunto de ativos de processo que possam ser reutilizados em futuras instâncias do processo; e
- gerência da engenharia de processo, com o propósito de gerenciar e orientar as atividades de engenharia de processo e desenvolver e manter o sistema de engenharia de processo de software [Kinnula 1999, p. 99-101].

Engenharia de Processo de Software, segundo Kinnula, é uma forma de aprendizagem organizacional que cobre o domínio de processos de software e que inclui os seguintes cinco processos:

- desenvolvimento de processo, com o propósito de projetar, implementar, refinar e testar uma mudança em um processo para melhorar a capacidade do processo em atingir as metas definidas do processo;
- utilização de processo, com o propósito de estabelecer uma instância de um processo por meio da utilização de ativos existentes para o processo e a realização dele na prática;
- avaliação de processo, com o propósito de coletar, arquivar, estabelecer uma linha básica (*baseline*), analisar e validar a correteza dos dados necessários para a engenharia de processo;
- gerência de ativos de processo, com o propósito de captar o conhecimento e tecnologia existentes, e as melhorias para os processos em um conjunto de ativos de processo que possam ser reutilizados em futuras instâncias do processo; e
- gerência da engenharia de processo, com o propósito de gerenciar e orientar as atividades de engenharia de processo e desenvolver e manter o sistema de engenharia de processo de software [Kinnula 1999, p. 99-101].

No capítulo sobre processos de engenharia de software do SWEBOK versão 2, El Emam propõe os termos engenharia de processo de software<sup>6</sup> e processos da engenharia de software. Segundo El Emam [2004], a área de processo de engenharia de software pode ser examinada em dois níveis. “O primeiro nível compreende as atividades técnicas e gerenciais nos processos de ciclo de vida de software que são executados durante a aquisição, desenvolvimento, manutenção e aposentadoria de software. O segundo é um meta nível, que compreende a definição, implementação, medição, gerenciamento, mudança e melhoria dos processos do ciclo de vida de software propriamente dito” [El Emam 2004, p. 9-1]. O primeiro nível se confunde com a própria engenharia de software, enquanto que o segundo nível pode ser considerado como uma disciplina autônoma, algo como a engenharia da engenharia de software. Podemos denominar o primeiro nível de Processos da Engenharia de Software e o segundo de Engenharia de processo de Software.

---

<sup>6</sup> Na versão atual do SWEBOK foi mantido o conceito mas não é mais utilizado este nome. Em conversa informal com um dos editores do SWEBOK (Alain Abran) foi comentada a decisão de não denominar nenhuma das áreas da engenharia de software de engenharia.

A denominação gestão de processos de software também tem sido utilizada para a área de melhoria de processo de software, com significado similar, como, por exemplo, [Garg 1996] e [Chrissis et al. 2003]. “As áreas de processo de gerência de processo contêm as atividades entre projetos relacionadas à definição, planejamento, recursos, utilização, implementação, monitoramento, controle, avaliação, medição e melhoria dos processos” [SEI 2002a, p. 48]. Garg classifica as funções de uma organização de desenvolvimento de software em (1) construir o produto de software, que é tratada pela engenharia de produto de software; (2) medir, controlar, modificar e gerenciar projetos de software, que é tratada pela gerência de projeto de software; e (3) definir, simular, medir e melhorar os processos da organização, que é tratada pela engenharia de processo de software [Garg 1996].

Uma outra denominação da área de melhoria de processo de software é integração de processo e melhoria de produto [Chrissis et al. 2003], no caso mais direcionada para o CMMI.

### 2.2.3 Contexto e objetivos estratégicos

Toda iniciativa de melhoria de processo de software em uma determinada organização deve estar alinhada ao **contexto e objetivos estratégicos** da organização. Para esse alinhamento podem ser utilizadas referências tais como BSC, SFO, BITS, SWOT e MAE.

*Balanced Scorecard* (BSC, Marcador Balanceado) é um framework para representação do desdobramento da visão em uma estratégia com as perspectivas financeira, clientes, processos e aprendizado que juntas descrevem a estratégia, o operacional e indicadores [Kaplan e Norton 1993]. O BSC foi expandido pelos próprios autores para a abordagem SFO (*Strategy-Focused Organization*, Organização focada na estratégia) que torna a estratégia um processo contínuo em uma organização com a participação de todos [Kaplan e Norton 2000]. Esta abordagem é composta por cinco princípios para uma organização focada em estratégia: Mobilizar para a mudança por meio da liderança dos executivos, traduzir a estratégia em termos operacionais, alinhar a organização à estratégia, motivar para tornar a estratégia o trabalho de todos todo dia, e governar para tornar a estratégia um processo contínuo.

O BSC foi adaptado pelo *European Software Institute* (ESI) para organizações intensivas em software, com o nome BITS (*Balanced Information Technology Scorecard*). BITS inclui uma quinta perspectiva, chamada de “pessoas”, e altera o nome da perspectiva “aprendizado” por “infra-estrutura e inovação” [Reo 2001]. BITS é composto por três fases: Início, Condução e Monitoração. Na fase de início, é analisado o alinhamento entre negócio e estratégia de melhoria. Na condução, é desenvolvida

uma estratégia para atingir os objetivos estratégicos. Na fase de monitoração, é desenvolvido um sistema de gerência quantitativa.

A técnica de análise SWOT (*Strengths, Weakness, Opportunities, Threats*) é um resultado de uma pesquisa conduzida pelo *Stanford Research Institute* entre 1960 e 1970 [Humphrey 2004]. Esta pesquisa buscava entender porque planejamentos corporativos geralmente falhavam. A análise SWOT orienta a identificação de fatores que devem influenciar o desempenho e resultados de uma organização. SWOT identifica os pontos fortes (“*strengths*”) e fraquezas (“*weakness*”) internos de uma organização e as oportunidades (“*opportunities*”) e ameaças (“*threats*”) do ambiente externo.

Uma definição mais detalhada destas quatro variáveis é dada em Castro et al. [2005]. Pontos fortes são condições do ambiente interno de uma organização, que apresentam situação atual favorável, em relação ao seu desempenho geral. Fraquezas são condições do ambiente interno de uma organização, que apresentam situação atual desfavorável, em relação ao seu desempenho geral. Ambas as condições de pontos fortes e fraquezas podem se relacionar a capacidades, estrutura de apoio à pesquisa, recursos financeiros, desempenho organizacional e alianças estratégicas. Oportunidades são variáveis do ambiente externo, de alta importância futura e positiva sobre as atividades e o desempenho da organização, enquanto que ameaças são variáveis do ambiente externo, de alta importância futura e negativa sobre as atividades e o desempenho da organização [Castro et al. 2005, p. 127-129].

A Matriz de Análise Estratégica (MAE) é definida em Castro et al. [2005, p. 93] como um instrumento simples e poderoso que junta oportunidades, ameaças, pontos fortes e fraquezas. A partir destas quatro variáveis, podem ser derivadas estratégias pró-ativas, para aproveitar oportunidades, ou reativas, para minimizar ameaças, sempre considerando os pontos fortes e fraquezas da organização.

#### **2.2.4 Modelo e engenharia de modelo**

Melhoria de processo utiliza um **modelo**, ou mais de um modelo, de referência para orientar as ações de melhoria. Um modelo de referência pode ser definido como “um framework para entendimento de relacionamentos significantes entre as entidades de algum ambiente, e para o desenvolvimento de padrões (*standards*) ou especificações consistentes que apoiam o ambiente. Um modelo de referência é baseado em um pequeno número de conceitos unificados e pode ser utilizado como base para educação e explicação dos padrões para não especialistas na área” [CCSDS 2001].

O conceito de modelo é utilizado em praticamente todos os elementos da engenharia de uma maneira geral e da engenharia de software e a engenharia de processo de software em particular.

Existem várias definições de modelos, entre elas, “uma simplificação de um sistema construída com um objetivo intencional, de tal forma que o modelo possa responder questões no lugar do sistema real” [Bézivin e Gerbé 2001, apud Favre 2005a] e “um conjunto de declarações sobre um sistema em estudo” [Seidewitz 2003]. Uma definição mais extensa de modelo e de modelagem é apresentada por Rothenberg [1989, apud Bézivin 2003]:

“Modelagem, em um sentido mais amplo, é o uso custo-efetivo de alguma coisa no lugar de outra coisa para um propósito cognitivo. Ele nos habilita a utilizar alguma coisa que é mais simples, segura ou mais barata que a realidade no lugar da realidade para algum propósito. Um modelo representa realidade para um determinado propósito; o modelo é uma abstração da realidade no sentido que ele não pode representar todos os aspectos da realidade. Isso nos permite tratar com o mundo de uma forma simplificada evitando a complexidade, periculosidade e irreversibilidade da realidade.”

Uma mesma parte do mundo pode ser representada por diferentes modelos [Figura 9].

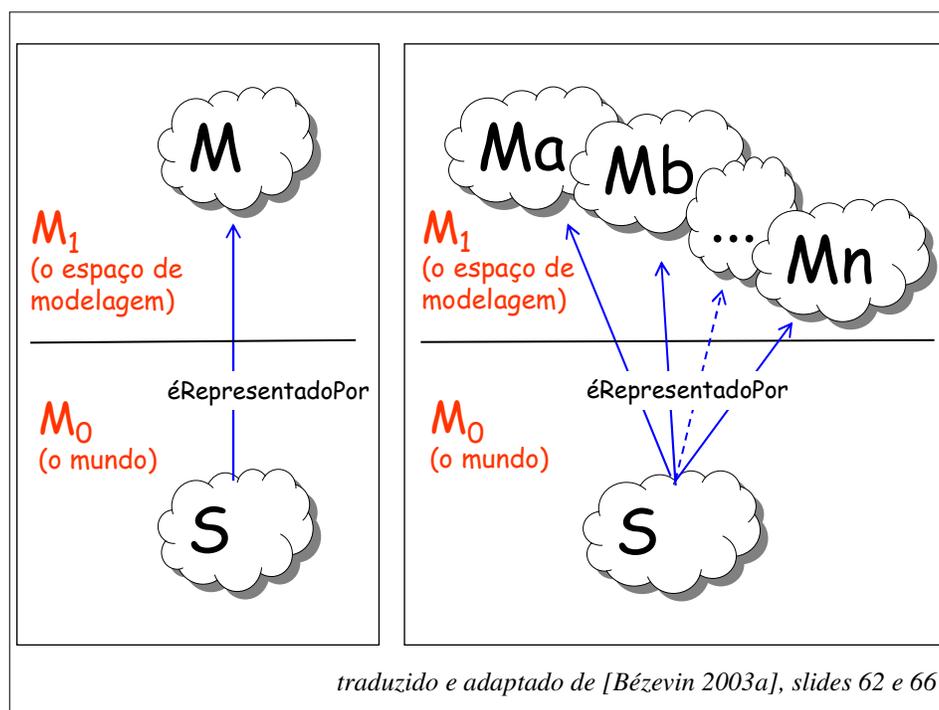


Figura 9 – Modelos como representação de sistemas do mundo

Conforme ilustrado na Figura 9, adaptada de Bézivin [2003a, slides 62 e 66], um modelo M é uma representação simplificada de uma parte do mundo chamada de sistema S e um determinado sistema S pode ter vários modelos diferentes, identificados por exemplo como Ma, Mb, ..., Mn, onde

cada modelo representa um ou mais determinado aspecto do sistema e somente esses aspectos. Com isso pode ser identificado o espaço do mundo ( $M_0$ ) para os sistemas e o espaço de modelagem ( $M_1$ ) para os modelos.

O princípio da ‘substitutabilidade’ limitada define que o propósito de um modelo é sempre habilitar a resposta a um conjunto específico de perguntas substituindo o sistema, exatamente da mesma forma que o sistema responderia a questões similares [Bézivin 2003a]. Segundo Selic [2003], um modelo para ser útil e efetivo deve possuir, em um grau suficiente, as cinco características de ser uma abstração, ser fácil de ser entendido, ser preciso, permitir a previsão e ser significativamente mais barato de ser construído e analisado do que o sistema que ele representa.

George Box alerta que “todos os modelos são errados, mas alguns são úteis” [1979]. Box reconhece as limitações necessárias para a construção e utilização de modelos e alerta para um foco maior na utilidade dos modelos. Com isto o foco maior é no processo de significação de um modelo, como um signo, para construção e análise de um sistema, de na linha da semiótica triádica de Peirce [1914].

Um modelo de um sistema é uma descrição ou especificação desse sistema e seu ambiente para um determinado propósito [OMG 2003]. Quando um modelo é uma descrição de um sistema já existente, o modelo é denominado de modelo descritivo, enquanto que quando um modelo é uma especificação de um sistema a ser construído, o modelo é denominado de modelo prescritivo [Bézivin 2003, p. 19]. Em ambos os casos, o sistema é representado pelo modelo. Essa distinção entre modelos descritivos e prescritivos não é sempre clara, principalmente pela noção de co-evolução de modelo e sistema [Favre 2005a, p. 19].

Quando o sistema modelado é um modelo, o modelo deste modelo pode ser chamado de metamodelo. Metamodelo é uma definição precisa das construções e regras necessárias para a criação de modelos semânticos [Metamodel 2005]. Metamodelos possibilitam a explicitação e a verificação automática de quais aspectos particulares devem ser considerados para constituir o modelo.

Bézivin [1998] defende que **engenharia de modelos** está tornando-se uma parte essencial da engenharia de software e com isto metamodelagem e ontologia têm papéis importantes. Segundo Bézivin [1998], “uma ontologia é uma descrição explícita e precisa de conceitos e relacionamentos que existem em um domínio particular, tais como uma determinada organização, um campo de estudo e uma área de aplicação.”

**Engenharia de modelos de capacidade de processo** é uma denominação para uma área existente mas ainda não referenciada como tal e portanto ainda não nomeada. Representa a engenharia

de construção de modelos de capacidade de processo. Esta engenharia vem sendo realizada, por exemplo, por grupos do SEI, na construção do modelo SW-CMM e dos modelos do CMMI; do ISO/IEC WG10, na construção das várias versões do modelo ISO/IEC 15504-5 e mais recentemente do MPS.BR, na construção do modelo MR-MPS.

Favre define dois termos Engenharia de Modelos e Engenharia Dirigida a Modelos como: “engenharia de modelos é a produção disciplinada e racionalizada de modelos, enquanto que a engenharia dirigida por modelos é um subconjunto da engenharia de sistemas no qual o processo conta fortemente com o uso de modelos e engenharia de modelos” [Favre 2005a, p 26]. Engenharia Dirigida a Modelos está consolidada como MDE (*Model-Driven Engineering*).

MDE é ao mesmo tempo uma generalização de MDA (*Model-Driven Architecture*, Arquitetura Dirigida por Modelos) definida pela OMG, e de OOSE (*Object-Oriented Software Engineering*, Engenharia de Software Orientada a Objetos). Enquanto OOSE é baseada no princípio que “tudo é um objeto”, MDE é baseado em “tudo é um modelo”. Desta forma “um sistema é representado por um modelo” e “um modelo é conforme um metamodelo” [Figura 10].

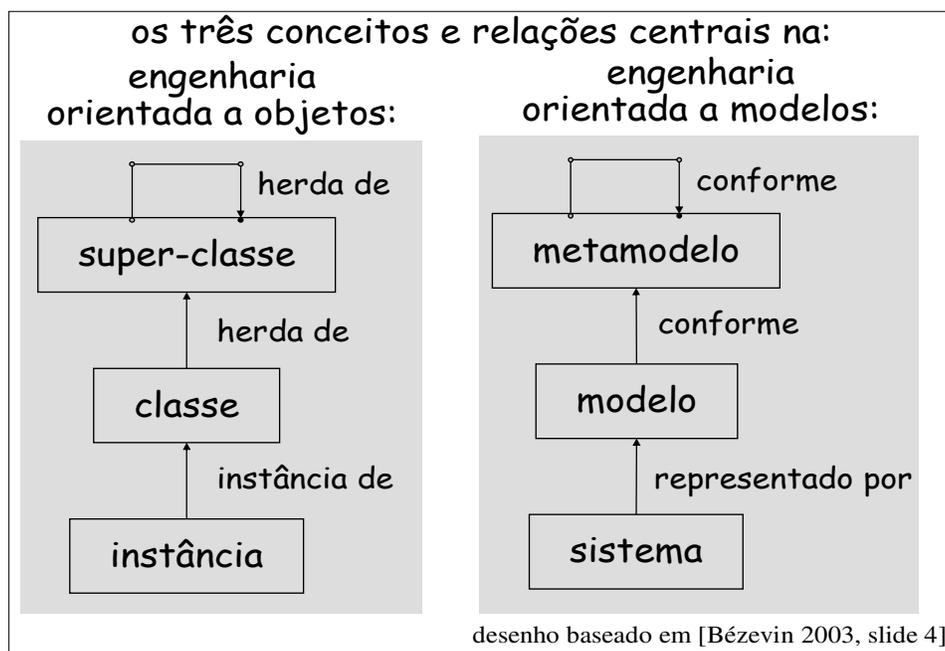


Figura 10 – Conceitos e relações centrais da OOSE e MDE

Um modelo pode ser uma representação formal ou informal. Uma representação é dita como formal quando é baseada em uma linguagem com forma (sintaxe), significado (semântica) e regras (de análise, inferência ou prova para suas construções) bem definidas. Um metamodelo pode ser definido mais informalmente como “um modelo de um modelo” ou mais formalmente como “um modelo de

uma linguagem de modelagem” [Favre 2005c]. Para MDE apenas representações formais são consideradas para modelo e metamodelos. A correspondência entre um sistema e um modelo é definida precisamente por um metamodelo, como uma ontologia simples [Bézivin2003a] [Figura 11].

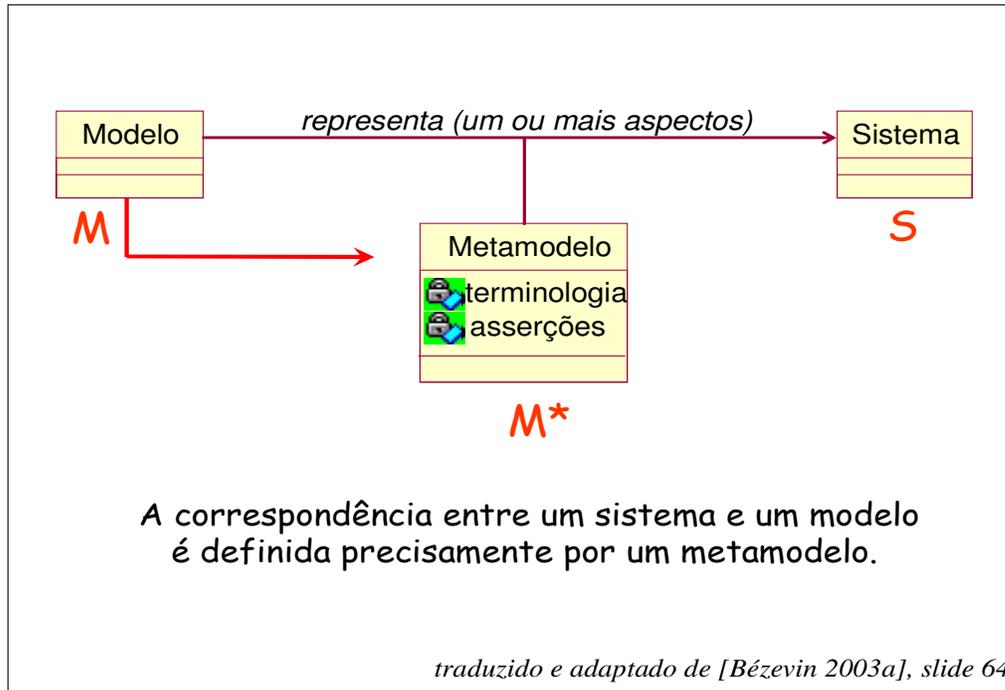


Figura 11 - Metamodelos como ontologias simples [Bézivin2003a]

A OMG consolidou e tem utilizado uma hierarquia com quatro camadas de modelagem e metamodelagem, chamadas de M0, M1, M2 e M3 [Figura 12].

A camada M0 representa o mundo real por meio de dados. A camada M1 representa um modelo do ponto de vista definido para caracterizar os dados. A camada M2 representa o modelo do modelo M1, ou seja um metamodelo. A camada M3 representa o modelo do metamodelo M2.

A camada M3 de meta-metamodelagem forma uma fundação para a arquitetura de metamodelagem. A primeira responsabilidade desta camada é definir a notação para especificação de metamodelos. Um meta-metamodelo define um modelo em uma abstração mais alta do que a do metamodelo e tipicamente é mais compacta que a do metamodelo que ele descreve. Um meta-metamodelo pode definir múltiplos metamodelos e pode existir múltiplos meta-metamodelos associados com cada metamodelo.

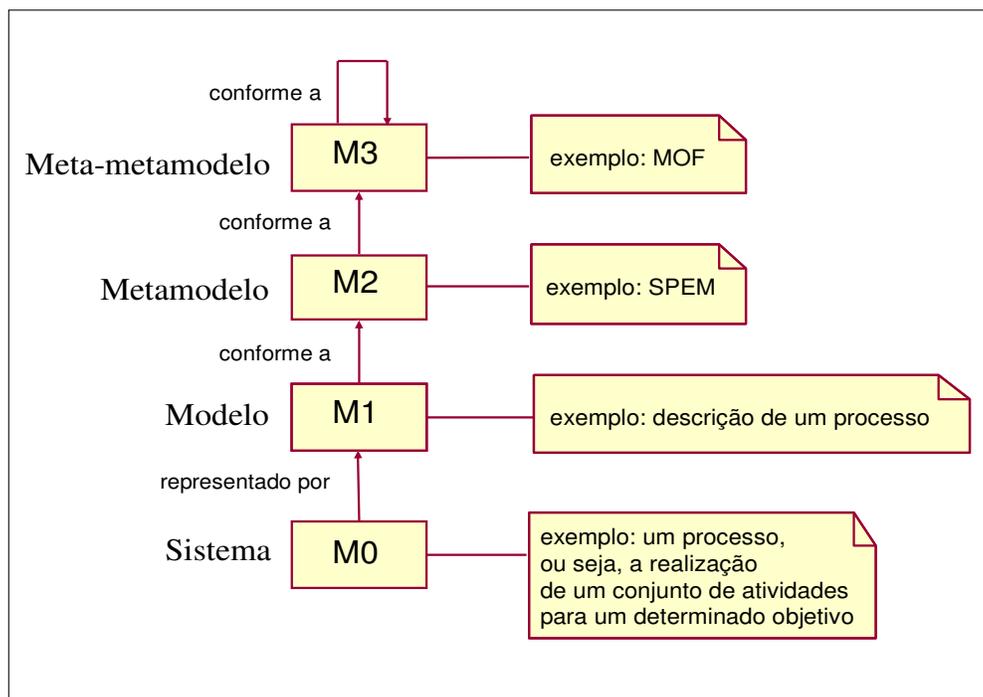


Figura 12 – Metamodelagem de quatro camadas e exemplo para processo

MOF (*Meta Object Facility*, Estrutura para Meta Objetos) define uma linguagem abstrata e um framework para especificação, construção e gerência de metamodelos independentes de tecnologia (*technology neutral*) [OMG 2002a]. Exemplos de metamodelos são UML, SPEM e o próprio MOF. MOF também define um framework para repositórios de implementação de modelos descritos pelos metamodelos. A especificação de MOF inclui uma definição formal do meta-metamodelo MOF, que é uma linguagem abstrata para especificação de metamodelos MOF.

O MDE da OMG estrutura um cenário com um único meta-metamodelo M3 (que é MOF), uma biblioteca de metamodelos compatíveis, cada um definindo uma Linguagem para Espaço de Projeto (*Design Space Language - DSL*) e com cada modelo definido na linguagem de seu único metamodelo [Bézivin2003a, slide 53].

UML (*Unified Modeling Language*, Linguagem de Modelagem Unificada) é uma linguagem-padrão para a visualização, especificação, construção e documentação de artefatos de software e sistemas [Booch et al. 1999, p. 13, OMG 2003a]. O modelo conceitual de UML é composto pelos blocos de construção básicos, regras que determinam como esses blocos podem ser combinados, e alguns mecanismos comuns. Existem três blocos básicos: itens, relacionamentos e diagramas. Existem quatro tipos de relacionamentos: dependência, associação, generalização e realização. UML dispõe de



## 2.2.5 Abordagem para melhoria de processo

Independente da definição específica de melhoria ou engenharia de processo de software, ela é realizada nas organizações intensivas em software por meio de abordagens. Uma **abordagem para melhoria ou engenharia de processo de software** é uma orientação para um conjunto de ações para a melhoria de processo em uma organização intensiva em software.

A ISO/IEC 15504-5 define melhoria de processo de software como um grupo de processos com três processos: estabelecimento de processo de software, avaliação de processo de software e melhoria de processo de software. O termo melhoria de processo de software é utilizado em dois sentidos, como o todo e também como uma das partes desse todo. O grupo de processo de melhoria de processo “consiste de processos realizados para definir, implementar, avaliar e melhorar os processos realizados em uma unidade organizacional” [ISO/IEC 15504-5 2006, p. 8]. O propósito do processo de **estabelecimento de processo** é “estabelecer um conjunto de processos organizacionais para todos processos do ciclo de vida à medida que eles se aplicam às atividades de negócio” [ISO/IEC 15504-5 2006, p. 70]. O propósito do processo de **avaliação de processo** é “determinar o quanto que os processos padrões da organização contribuem para o atendimento de seus objetivos estratégicos e apoiar o foco da organização na necessidade de melhoria contínua de processo” [ISO/IEC 15504-5 2006, p. 72]. O propósito do processo de **melhoria de processo** é “a melhoria contínua da eficiência e eficácia da organização por meio dos processos utilizados e mantidos de forma alinhada às necessidades de negócio” [ISO/IEC 15504-5 2006, p. 73].

Várias abordagens para melhoria de processo de software têm como referência o ciclo PDCA (*Plan-Do-Check-Act*, Planeja-Executa-Verifica-Atua), desenvolvido por Shewhart na década de 1930 e popularizado por Deming [1986]. O ciclo PDCA é iniciado com o planejamento do que será feito, estabelecimento de metas e definição dos métodos para atingir as metas. A segunda fase envolve a realização das atividades com os métodos definidos e conforme o planejado para atendimento das metas. A terceira envolve a verificação dos resultados em relação às metas definidas. Finalmente, a quarta fase envolve correções de rotas se for necessário e tomadas de ações corretivas ou de melhoria, caso necessário. Variações do ciclo PDCA incluem o ciclo SDCA e o ciclo DMAIC. O ciclo SDCA (*Standard-Do-Check-Act*, Padroniza-Executa-Verifica-Atua) é composto pelas fases de padronização, desenvolvimento, verificação e ação. O ciclo DMAIC (*Define, Measure, Analyse, Improve, Control*) é definido como parte da metodologia Six Sigma inventada por Bill Smith para a Motorola em 1986. As

fases do ciclo DMAIC são definir oportunidades, medir desempenho, analisar oportunidades, melhorar o desempenho e controlar o desempenho.

Duas abordagens para melhoria de processo de software, ambas baseadas no ciclo PDCA, são a abordagem IDEAL [McFeeley 1996] e o ciclo de melhoria da ISO/IEC 15504. A abordagem AMP1 foi desenvolvida a partir de uma combinação destas duas abordagens. A Figura 14 ilustra as fases dos quatro ciclos (PDCA, IDEAL, 15504 e AMP1).

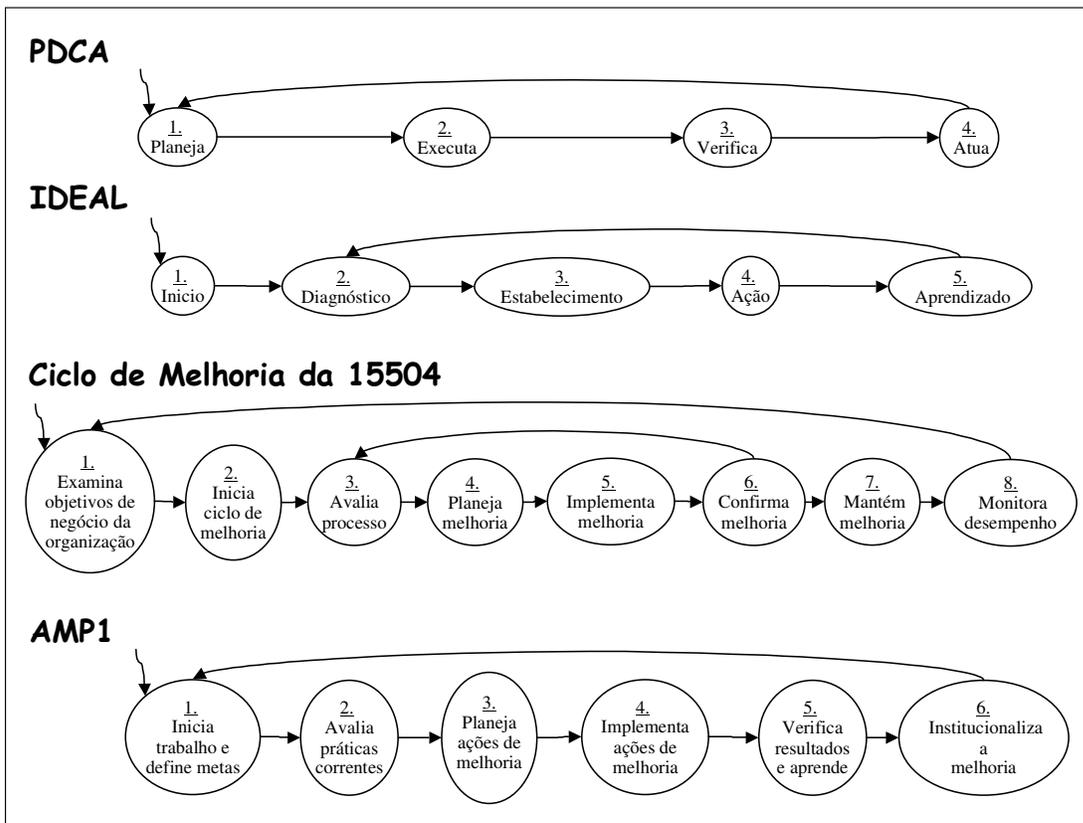


Figura 14 – Quatro abordagens similares para ciclos de melhoria de processo

A abordagem IDEAL (*Initiating, Diagnosing, Establishing, Acting, Learning*) é composta por cinco fases. A fase 1, denominada de “início”, é realizada com estímulo para melhoria, com uma justificativa de negócio para a melhoria, define contexto e estabelece patrocínio, e estabelece infraestrutura para melhoria. A fase 2, denominada de “diagnóstico”, é realizada com avalia e caracteriza práticas correntes, e desenvolve recomendações e documenta resultados. A fase 3, denominada de “estabelecimento” é realizada com define estratégia e prioridades, planeja ações, e estabelece equipes de ação. A fase 4, denominada de “ação”, é realizada com define processo e medições, planeja e realiza pilotos, e planeja, executa e acompanha instalações. A fase 5, denominada de “aprendizado”, é realizada com documenta e analisa ações, e revisa abordagem organizacional.

O ciclo para melhoria de processo da ISO/IEC 15504 foi publicado inicialmente na parte 7 das versões SPICE V1.0 e TR [Dorling 1995, ISO/IEC TR 15504-7 1998] e a versão atual, com algumas modificações, está publicada na parte 4 da versão IS [ISO/IEC 15504-4 2004] [Figura 15].

O ciclo da versão atual é composto por oito passos: examina metas de negócio da organização, inicia ciclo de melhoria de processo, avalia capacidade corrente, desenvolve plano de ação, implementa melhorias, confirma melhorias, mantém melhorias e monitora desempenho.

O ciclo para melhoria de processo da ISO/IEC 15504 tem sido utilizado como referência para outras abordagens, incluindo por exemplo a abordagem denominada IMPACT (*Initiate, Measure, Plan, Act, Confirm and Transfer*; Iniciar, Medir, Planejar, Atuar, Confirmar e Transferir) [ESI 1999] e a abordagem de *People-Process-Product and Cultural Layer* (Pessoas-Processo e Camada Cultural) [van Loon 2004].

Potter e Sakry [2002] definem a ‘Abordagem Baseada em Problemas e Metas’ (*Goal-Problem Approach*). Nesta abordagem as ações de melhoria são orientadas aos problemas que impedem o atendimento das metas de negócio da organização. Para tanto, as práticas dos modelos de processo podem ser utilizadas como referência. Para esta abordagem são definidas três fases: desenvolver plano, implementar plano e verificar andamento.

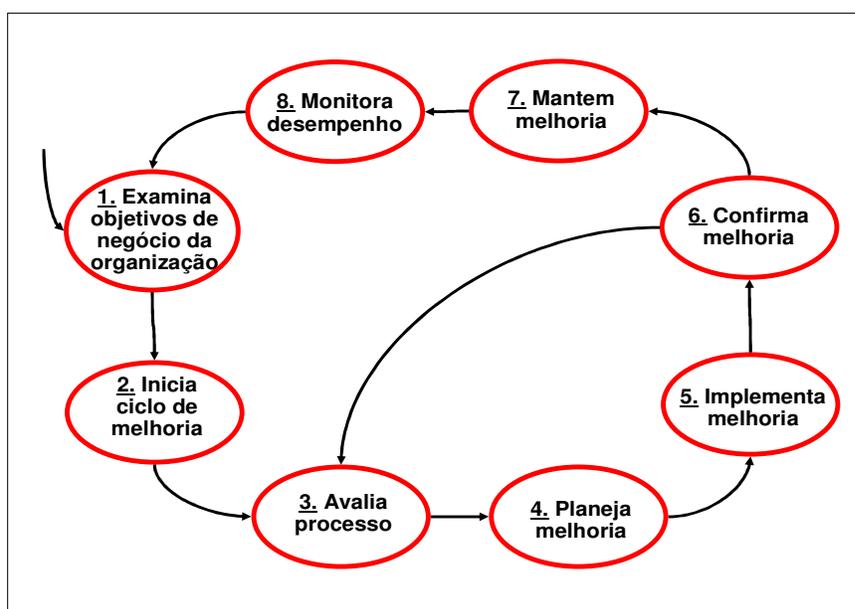


Figura 15 – Passos para um ciclo de melhoria de processo [ISO/IEC 15504-4]

Os passos para determinação da capacidade da ISO/IEC 15504 foi publicado inicialmente na parte 8 das versões SPICE V1.0 e TR [Dorling 1995, ISO/IEC TR 15504-8 1998] e a versão atual, com algumas modificações, está publicada na parte 4 [ISO/IEC 15504-4 2004][Figura 16].

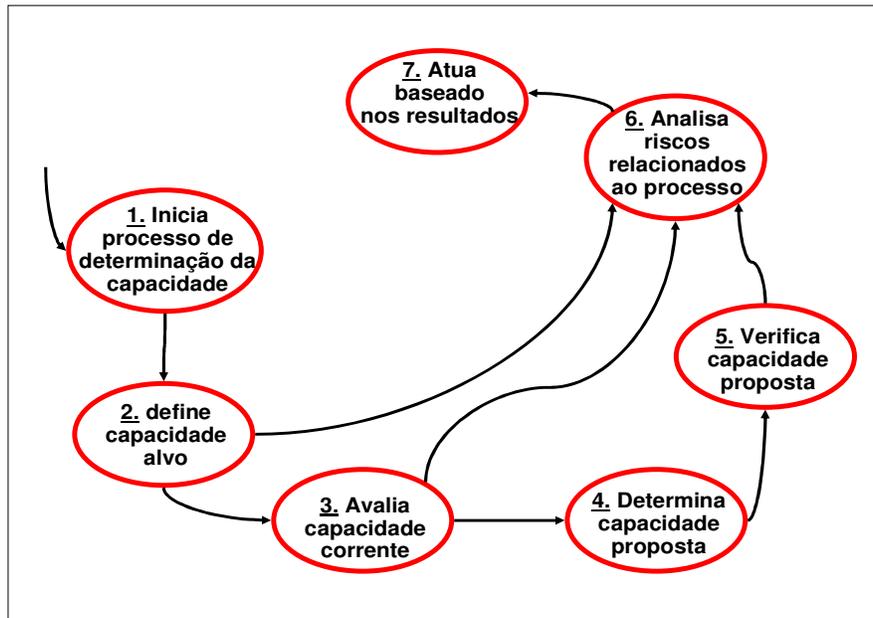


Figura 16 – Passos para determinação da capacidade [ISO/IEC 15504-4]

Os sete passos são: inicia determinação de capacidade de processo, define capacidade alvo, avalia capacidade corrente, determina capacidade proposta, verifica capacidade proposta, analisa riscos relacionados a processo, e atua baseado nos resultados.

### 2.2.6 Modelo de capacidade de processo

Um **modelo de capacidade de processo** é um conjunto de boas práticas, organizadas de forma sistemática, segundo o conceito de capacidade de processo. Alguns modelos acrescentam o termo “maturidade” para indicar o uso dos níveis de maturidade, como, por exemplo, os modelos SW-CMM e CMMI-SE/SW. Como esses modelos definem maturidade em função da capacidade de processo, o termo “capacidade” é suficiente para caracterizar todos os modelos da área.

**Outros tipos de modelo de referência, que não sejam modelo de capacidade de processo,** são modelos que têm sido utilizados como referências adicionais para melhoria de processo.

**Framework de modelos de capacidade de processo** é uma estrutura comum de vários modelos e o conjunto desses modelos.

**Arquitetura de modelos de capacidade de processo** esta relacionada com como um modelo é estruturado. **Arquitetura estagiada** é caracterizada por uma conjunto de níveis de maturidade de processo, enquanto que a **arquitetura contínua** é caracterizada por um conjunto de áreas de processo e um conjunto de níveis de capacidade de processo. A arquitetura contínua separa as melhores práticas

em duas grandes categorias: aquelas relacionadas a “o que fazer”, organizadas em processos específicos, e aquelas relacionadas ao “quão bem fazer qualquer coisa que seja feita”, organizadas em níveis de capacidade genéricos. Esta divisão não é absoluta, existindo práticas que podem ser classificadas nas duas categorias dependendo do ponto de vista.

A Figura 17 ilustra o relacionamento de quatro tipos de modelos com o processo de uma organização. Um modelo de capacidade de processo está diretamente relacionado ao processo de uma organização, quando utilizado como referência para uma melhoria de processo ou avaliação de processo. Esse processo pode estar sendo representado por um modelo do processo da organização, que por sua vez pode ser uma adaptação de um modelo de processo genérico. O processo da organização também pode utilizar como referência para a melhoria outros tipos de modelos, que não sejam modelos de capacidade de processo. Os quatro tipos de modelos têm um relacionamento indireto entre eles, por meio do processo da organização.

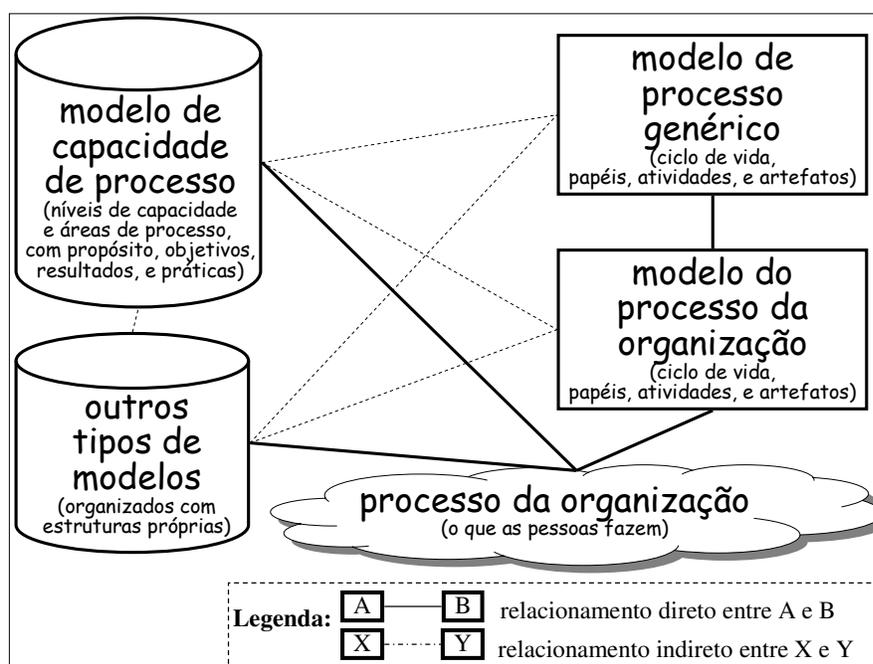


Figura 17 – Quatro tipos de modelos e o processo de uma organização

## 2.2.7 Capacidade de processo

O conceito chave que caracteriza a melhoria de processo e a diferencia de outras abordagens relacionadas a processo é o de capacidade de processo. Na definição de melhoria de processo de Ahern et al. [2001] é realçado e definido o termo capacidade de processo: “Melhoria de processo (de software) baseada em modelo envolve o uso de um modelo (de capacidade de processo) para guiar a

melhoria dos processos de uma organização (por meio do aumento da capacidade dos processos)”. “Essencialmente **capacidade de processo** é a inerente habilidade do processo de produzir resultados planejados. Com o aumento da capacidade de um processo, ele torna-se mais previsível e mensurável, e as causas mais significativas de baixa qualidade e produtividade são controladas e eliminadas. Uma organização torna-se mais “madura” à medida que ela melhora continuamente a capacidade de seus processos” [Ahern et al. 2001, p. 4]. A ISO/IEC 15504-1 define capacidade de processo como “uma caracterização da habilidade de um processo em atender objetivos requeridos ou contribuir (juntamente com outros processos) para o atendimento de objetivos requeridos” [ISO/IEC 15504-1 2004, p. 4]. Do ponto de vista do controle estatístico de processo, capacidade de processo refere-se à “previsibilidade do desempenho de um processo sob controle estatístico” [Florac e Carleton 1999, p. 176]. Desta forma, “quando um processo está sob controle estatístico em relação a um determinado conjunto de atributos, nós temos uma base válida para prever, dentro de certos limites, como será o desempenho do processo no futuro, e enquanto o processo continuar dentro do mesmo estado de controle estatístico, praticamente todos os resultados de medições desses atributos estarão dentro dos seus limites de processo naturais, o que é definido como capacidade de processo” [Florac e Carleton 1999, p. 176-177]. Na visão da ISO/IEC 15504 e do CMMI, esta definição é apropriada apenas para os níveis 4 e 5 de capacidade. Desta forma capacidade de processo é considerada de uma forma mais ampla, desde o nível 0 ao nível 5.

**Desempenho de processo** é “a extensão na qual a execução de um processo atinge seu propósito” [ISO/IEC 15504-1 2004] e “uma medida dos resultados reais de um processo, que é caracterizada por medições de processo, como, por exemplo, esforço, tempo de ciclo, e eficiência em remover defeitos, e medições de produto. Como, por exemplo, confiabilidade, densidade de defeitos e tempo de resposta” [Chrissis et al. 2003].

Dois outros conceitos chave da área, no caso os conceitos de áreas de processo e níveis de capacidade de processo são relacionados entre si e em um terceiro conceito também chave: perfil de capacidade de processo. Uma **área de processo** descreve uma abstração de um conjunto de atividades, descritas no nível 1 de capacidade. Os **níveis de capacidade** descrevem o “quão bem” esse conjunto de atividades é realizado em relação a uma área de processo.

A Tabela 7 descreve como exemplo de uma área de processo, o processo de engenharia de domínio do modelo ISO/IEC 15504-5 [2006].

Tabela 7 – Processo de engenharia de domínio da ISO/IEC 15504-5 [2006]

<b>Identificação do processo</b>	REU.3	
<b>Nome do processo</b>	Engenharia de Domínio	
<b>Propósito do processo</b>	O propósito do processo de Engenharia de Domínio é desenvolver e manter modelos e arquiteturas do domínio e ativos para o domínio	
<b>Resultados do processo</b>	Como resultado de uma implementação com sucesso do processo de Engenharia de Domínio:	
	1) as formas de representação para os modelos do domínio e arquiteturas do domínio são selecionadas;	
	2) as fronteiras do domínio e seus relacionamentos com outros domínios são estabelecidas;	
	3) um modelo do domínio que represente as características essenciais comuns e diferentes características, capacidades, conceitos e funções do domínio são desenvolvidas;	
	4) uma arquitetura do domínio descrevendo a família de sistemas abrangida pelo domínio é desenvolvida;	
	5) ativos pertencentes ao domínio são especificados;	
	6) ativos pertencentes ao domínio são especificados são adquiridos ou desenvolvidos, e mantidos durante seus ciclos de vida; e	
	7) os modelos do domínio e arquiteturas do domínio são mantidos durante seu ciclo de vida.	
<b>Práticas base</b>	REU.3.BP.1 Selecionar representação do domínio. Selecionar as formas de representação para o domínio, classificações do domínio e outras descrições de templates necessárias para serem utilizadas para os modelos e arquiteturas do domínio, de acordo com os padrões de reuso da organização. [resultado 1]	
	REU.3.BP.2 Estabelecer fronteiras do domínio. Estabelecer fronteiras do domínio e seus relacionamentos com outros domínios. [resultados 2,3,4]	
	REU.3.BP.3 Desenvolver modelo do domínio. Desenvolver um modelo do domínio que represente os pontos essenciais e diferentes características, capacidades, conceitos e funções do domínio. [resultados 2, 4]	
	REU.3.BP.4 Desenvolver arquitetura do domínio. Desenvolver uma arquitetura do domínio descrevendo a família de sistemas abrangida pelo domínio. [resultado 5]	
	REU.3.BP.5 Definir ativos do domínio. Definir os ativos pertencentes ao domínio. [resultado 6]	
	REU.3.BP.6 Adquirir ou desenvolver, e manter ativos. Adquirir ou desenvolver, e manter durante seu ciclo de vida, os ativos pertencentes ao domínio. [resultado 6]	
	REU.3.BP.7 Manter modelos e arquiteturas do domínio. Manter os modelos do domínio e arquiteturas do domínio durante seu ciclo de vida. [resultado 7]	
<b>Produtos de trabalho</b>	<b>Entrada:</b>	<b>Saída:</b>
	01-02 objeto reusável [resultado: 6]	01-02 objeto reusável [resultado: 6]
	04-02 arquitetura de domínio [resultado: 1, 2, 7]	04-02 arquitetura de domínio [resultado: 1, 2, 4, 7]

	04-03 modelo de domínio [resultado: 7]	04-03 modelo de domínio [resultado: 3, 7]
	08-02 plano de aquisição [resultado: 6]	
	08-12 plano de projeto [resultado: 6]	
	13-16 solicitação de mudança [resultado: 6, 7]	
		13-21 registro de controle de mudança [resultado: 6, 7]
	14-03 registro de ativos de hardware [resultado: 5]	
	14-07 registro de ativos de software resultado: 5]	
	17-01 especificação de ativos [resultado: 6]	17-01 especificação de ativos [resultado: 5]
	17-06 especificação de interface de domínio [resultado: 2, 6, 7]	17-06 especificação de interface de domínio [resultado: 2]
	18-04 padrão de representação de domínio [resultado: 1]	18-04 padrão de representação de domínio [resultado: 1]

A Tabela 8 descreve como exemplo a definição do nível 2 de capacidade, sem seus atributos de processo, do modelo ISO/IEC 15504-5 [2006].

Tabela 8 – Nível 2 de capacidade, sem seus atributos, da ISO/IEC 15504-5 [2006]

<p><b>Nível 2: Processo Gerenciado</b>  O processo executado, descrito previamente, é agora implementado de uma forma gerenciada (planejada, monitorada e ajustada) e seus produtos do trabalho são estabelecidos, controlados e mantidos apropriadamente.  Os atributos de processo PA 2.1 e PA 2.2 demonstram o atendimento desse nível de capacidade.</p>
--

A Tabela 9 descreve como exemplo, o atributo de processo PA 2.1, sem seus indicadores, do nível 2 de capacidade do modelo ISO/IEC 15504-5 [2006].

Tabela 9 - Atributo PA 2.1, sem os indicadores, da ISO/IEC 15504-5 [2006]

<p><b>PA 2.1 Atributo de gerência de execução</b>  O atributo de gerência de execução define uma medida da extensão da qual a execução do processo é gerenciada. Como um resultado do sucesso total desse atributo:</p>
a) objetivos de desempenho do processo são identificados;
b) execução do processo é planejada e acompanhada;
c) execução do processo é ajustada para satisfazer planos;
d) responsabilidades e autoridades para executar o processo são definidas, atribuídas e comunicadas;
e) recursos e informações necessárias para a executar o processo são identificados, disponibilizados, alocados e utilizados; e
f) interfaces entre as partes envolvidas são gerenciadas para garantir comunicação efetiva e clara da atribuição de responsabilidades.

A Tabela 10 descreve como exemplo, o atributo de processo PA 2.2, com seus indicadores, do nível 2 de capacidade do modelo ISO/IEC 15504-5 [2006].

Tabela 10 – Atributo PA 2.2, com os indicadores, da ISO/IEC 15504-5 [2005]

<b>PA 2.2 Atributo de gerência de produto de trabalho</b>
O atributo PA 2.2: Atributo de Gerência de Produto de Trabalho define uma medida da extensão na qual a execução do processo é gerenciada para produzir resultados de trabalho que são apropriadamente identificados, documentados e controlados.
Como resultado do sucesso total desse atributo:
a) requisitos para produtos de trabalho do processo são definidos;
b) requisitos para a documentação e controle dos produtos de trabalho são definidos;
c) Produtos de trabalho apropriadamente identificados, documentados e controlados;
d) Produtos de trabalho são revisados de acordo com o especificado e planejado, e ajustados às necessidades para adequar-se aos requisitos.
Nota 1: Os requisitos para documentação e controle de produtos de trabalho podem incluir requisitos para a identificação das mudanças e estado da revisão, aprovação e reaprovação de produtos de trabalho e a confecção de versões relevantes de produtos de trabalho aplicáveis, disponíveis em pontos de uso.
Nota 2: Os produtos de trabalho referidos nesta cláusula são os resultantes da execução com sucesso do processo.
<b>Práticas genéricas para PA 2.2</b>
<b>GP 2.2.1 Define os requisitos para os produtos de trabalho.</b>
Os requisitos para os produtos de trabalho a serem produzidos são definidos. Requisitos podem incluir a definição do conteúdo e da estrutura.
Critérios de qualidade dos produtos de trabalho são identificados.
Revisões apropriadas e critérios de aprovação para os produtos de trabalho são definidos.
<b>GP 2.2.2 Define os requisitos para documentação e controle dos produtos de trabalho.</b>
Requisitos para documentação e controle dos produtos de trabalho são definidos. Esses requisitos podem incluir requisitos para (1) distribuição, (2) identificação dos produtos de trabalho e seus componentes, e (3) rastreabilidade
Dependências entre produtos de trabalho são identificadas e entendidas.
Requisitos para aprovação dos produtos de trabalho a serem controlados são definidos.
<b>GP 2.2.3 Identifica, documenta e controla os produtos de trabalho.</b>
Os produtos de trabalho a serem controlados são identificados.
Um controle de mudança é estabelecido para produtos de trabalho.
Os produtos de trabalho são documentados e controlados de acordo com requisitos.
Versões dos produtos de trabalho são assinalados para configurações de produto quando aplicável.
Os produtos de trabalho são tornados disponíveis por meio de mecanismos de acesso apropriados.
O estado de revisão dos produtos de trabalho pode ser facilmente obtido.
<b>GP 2.2.4 Revisa e ajusta produtos de trabalho para satisfazer os requisitos definidos.</b>
Produtos de trabalho são revistos em relação aos requisitos definidos de acordo com o planejado.
Problemas e questões identificados nas revisões de produtos de trabalho são resolvidos.
<b>Recursos genéricos para PA 2.2</b>

Métodos e ferramentas para gerência de requisitos; [PA 2.2 Resultados a, b, c]
Sistema de gerência de configuração; [PA 2.2 Resultados b, c]
Ferramentas para elaboração e apoio a documentos; [PA 2.2 Resultados b, c]
Procedimentos para controle e identificação de documentos; [PA 2.2 Resultados b, c]
Experiências e métodos para revisão de produto de trabalho; [PA 2.2 Resultados d]
Ferramentas e métodos para gerência de revisões; [PA 2.2 Resultados d]
Intranets, extranets e/ou outros mecanismos de comunicação; [PA 2.2 Resultados b, c]
Mecanismos para gerência de problemas e questões. [PA 2.2 Resultados d]
<b>Produtos de trabalho genéricos para PA 2.2</b>
08-00 Plano [PA 2.2 Resultados b]
Expressa política ou estratégia selecionada para gerenciar produtos de trabalho.
Descreve requisitos para desenvolver, distribuir e manter produtos de trabalho.
Define ações de controle de qualidade para gerenciar a qualidade do produto de trabalho.
13-00 Registro [PA 2.2 Resultados d]
Demonstra revisão de produto de trabalho e contribui para rastreabilidade.
Descreve não conformidades detectadas durante revisões de produto de trabalho.
Provê evidência que as mudanças estão sob controle.
14-00 Registro [PA 2.2 Resultados c]
Registra o estado de documentos ou produto de trabalho.
16-00 Repositório [PA 2.2 Resultados c]
Contem e torna disponíveis produtos de trabalho e/ou itens de configuração.
Apoia o acompanhamento de mudanças de produtos de trabalho.
18-00 Padrões [PA 2.2 Resultados a]
Define os requisitos funcionais e não funcionais dos produtos de trabalho.
Identificas as dependências entre os produto de trabalho.
Identifica critérios de aprovação para os documentos.
20-00 Template [PA 2.2 Resultados a, b]
Define os atributos associados aos produtos de trabalho a serem criados.
21-00 Produto de trabalho [PA 2.2 Resultados a, b, c, d]
Demonstra produtos de trabalho específicos do processo a serem gerenciados.

### 2.2.8 Perfil de capacidade de processo

O conceito de **Perfil de Capacidade de Processo** (PCP) representa a combinação destas duas categorias. A Figura 18 representa um exemplo de PCP do modelo ISO/IEC 15504-5.

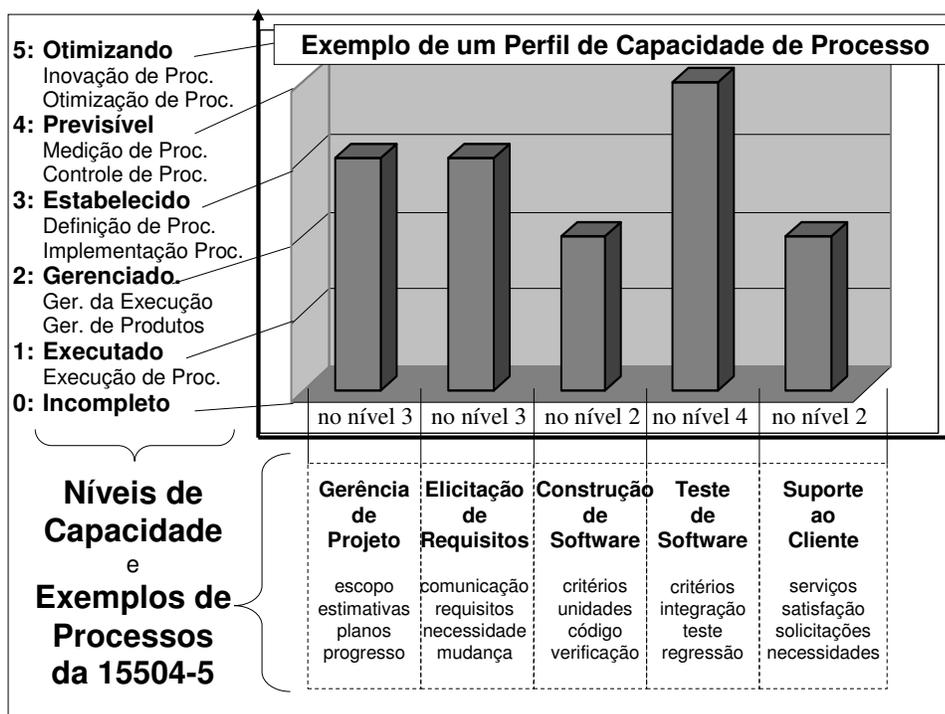


Figura 18 – Exemplo de um PCP no modelo ISO/IEC 15504-5

O PCP da Figura 18 é composto por cinco processos (ou áreas de processo) da ISO/IEC 15504-5, cada um em um determinado nível de capacidade. O processo de gerência de projeto, que envolve atividades relacionadas a escopo, estimativas, planos e progresso do projeto, está no nível 3 de capacidade. O processo de Elicitação de requisitos também está no nível 3. O processo de construção de software está no nível 2. O processo de teste de software está no nível 4 e o processo de suporte ao cliente está no nível 2.

A Figura 19 representa outro exemplo de PCP, no caso com área de processo (por coincidência também em número de cinco) do modelo CMMI-SE/SW. Nesse PCP, a área de processo desenvolvimento de requisitos está no nível 2 de capacidade, gerência de requisitos está no nível 3, planejamento de projeto no nível 2, gerência de riscos no nível 4, e acompanhamento de projeto no nível 3.

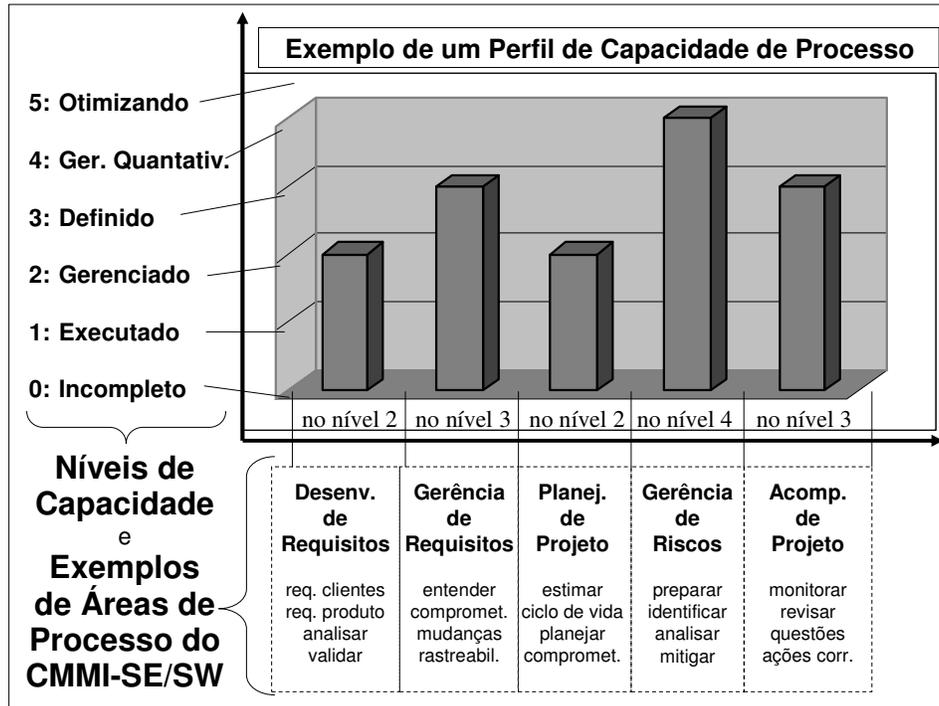


Figura 19 –Outro exemplo de PCP, no modelo CMMI-SE/SW

**Nível de maturidade** é um conjunto predefinido de áreas de processo em determinado nível de capacidade.

**Perfil resultado de avaliação** é um perfil de capacidade de processo resultante de uma avaliação de processo. **Perfil alvo para melhoria** é um perfil de capacidade de processo utilizado como alvo e referência para um ciclo de melhoria.

## 2.2.9 Medição

A área de **medição** de software e sistemas foi recentemente reavaliada pela comunidade para a definição da Norma ISO/IEC 15939 [2002]. Este trabalho utilizou como base as técnicas de medição com três níveis, como, por exemplo, o *Goal-Question-Metric* (GQM). Esta norma define três conceitos chaves: necessidade de informação, modelo de medição e um processo genérico de medição. Ao mesmo tempo foi definido o *Practical Software (and System) Measurement* (PSM) [Aguiar 2002, McGarry et al. 2002], pelo mesmo grupo, e com os mesmos conceitos, que também foram utilizados para a definição da área de processo de medição e análise dos modelos do CMMI e do processo de medição da ISO/IEC 15504-5. Desta forma esses três conceitos básicos, sejam na ISO/IEC 15939 ou no PSM têm se tornado dominantes na área [Card e Jones 2004].

O modelo de medição define o Produto de Informação para satisfazer uma Necessidade de Informação [Figura 20].

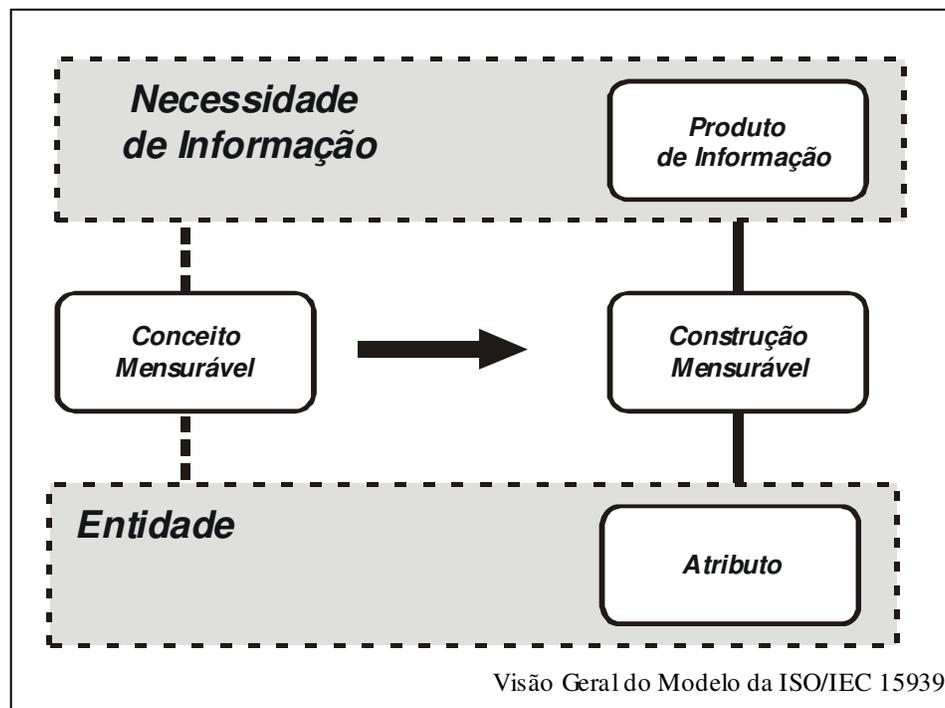


Figura 20 – Visão geral do modelos de medição da ISO/IEC 15939

Esse Produto de Informação é produzido por um Construção Mensurável a partir de Atributos que representam Entidades.

Um Conceito Mensurável é uma idéia sobre como uma Necessidade de Informação pode ser satisfeita a partir de Entidades. Um Conceito Mensurável pode ser implementado por diferentes Construções Mensuráveis. Uma Construção Mensurável é composta por Medidas Básicas, Medidas Derivadas e Indicadores [Figura 21].

Uma Medida Básica inclui um atributo mensurável de uma Entidade, um método para quantificação do atributo e um valor resultante da aplicação do método. Relacionados a uma medida básica, temos os conceitos de escala de medição, unidade de medição, observação (como o ato de designar um valor) e unidade de observação.

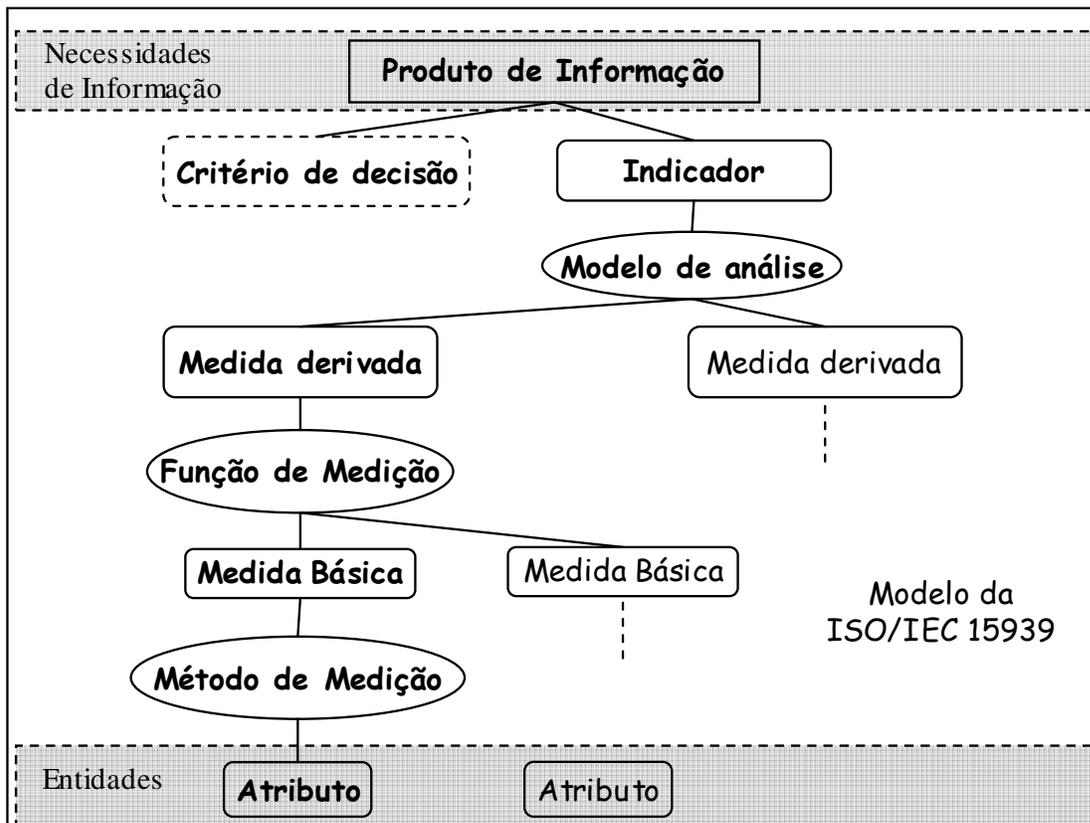


Figura 21 – Modelo de informações de medição [ISO/IEC 15939 2002 p.20]

Uma Medida Derivada incorpora informações sobre dois ou mais atributos ou várias observações de um mesmo atributo. Uma Medida Derivada inclui dois ou mais valores de Medidas Básicas e/ou Medidas Derivadas, uma função matemática combinando os valores, e um valor resultante da aplicação da função.

Um Indicador é uma medida que fornece uma estimativa ou avaliação de atributos especificados com relação a uma Necessidade de Informação. Um Indicador inclui: um ou mais valores de Medidas Básicas e/ou Derivadas, um Modelo de Análise combinando os valores, um valor resultante da aplicação do modelo, um critério de decisão utilizado para avaliar o valor do indicador.

Outro impacto do PSM e ISO/IEC 15939 é a não utilização do termo “métrica” para software e sistemas. Foi constatado que esse termo tem uma definição bem precisa na matemática, que não é adequada à área de software e sistemas, e que diferentes grupos tinham diferentes definições para esse termo. A Figura 22 ilustra um exemplo de medição em PSM.

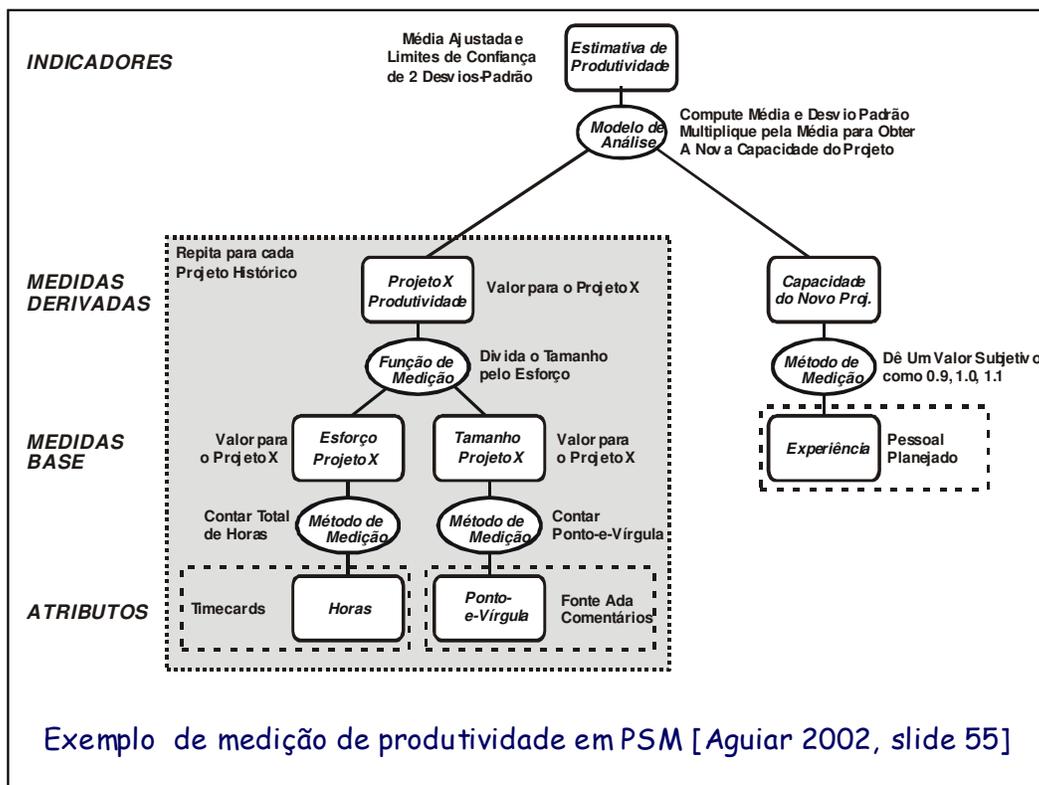


Figura 22 – Exemplo de medição em PSM [Aguar 2002, slide 55]

O processo de medição envolve atividades de estabelecer e manter comprometimento para medição, planejar o processo de medição, realizar o processo de medição e avaliar a medição [Figura 23].

A atividade “estabelecer e sustentar comprometimento para medição” envolve as tarefas de (a) aceitar os requisitos para medição; e (b) atribuir recursos. A atividade “planejar o processo de medição” envolve as tarefas de (a) caracterizar a unidade organizacional, (b) identificar necessidades de informação, (c) selecionar medições, (d) definir procedimentos para coleta e análise de dados e divulgação dos produtos de informação, (e) definir critérios para avaliar os produtos de informação e processo de medição, (f) revisar, aprovar e prover recursos para as atividades de medição, e (g) adquirir e implementar tecnologia de apoio. A atividade de “realizar o processo de medição” envolve as tarefas de (a) integrar os procedimentos, (b) coletar dados, (c) analisar dados e desenvolver produtos de informação, e (d) comunicar resultados. A atividade “avaliar medição” envolve as tarefas de (a) avaliar os produtos de informação e o processo de medição, e (b) identificar melhorias potenciais.

As necessidades de informações são geralmente derivadas de duas fontes: objetivos e metas a serem alcançadas, e obstáculos que ameaçam o atendimento desses objetivos e metas. O PSM organiza estas necessidades de informação em sete categorias: (i) prazo e progresso, (ii) recursos e custos, (iii) tamanho e estabilidade do produto, (iv) qualidade do produto, (v) desempenho do processo, (vi) eficácia da tecnologia, e (vii) satisfação do cliente.

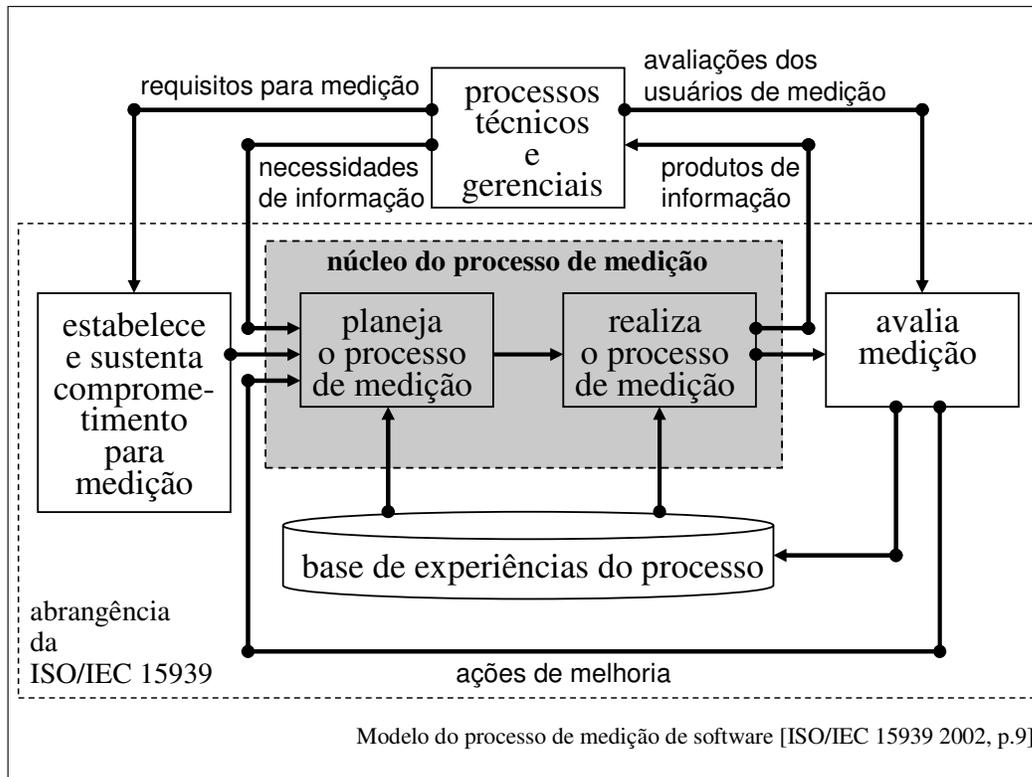


Figura 23 - Modelo do processo de medição de software [ISO/IEC 15939 2002, p.9]

O PSM e a ISO/IEC 15939 são baseados em três fundamentos para o sucesso de um programa de medição:

- medição é um processo consistente e flexível que tem que ser customizado para as necessidades de informações específicas e as características únicas de um determinado projeto ou organização,
- tomadores de decisão, sejam técnicos ou gerentes de negócio, têm que entender o que está sendo medido, e
- os resultados de medição têm que ser utilizados na tomada de decisões [Jones 2003].

## 2.3 Modelos de capacidade de processo

Dois frameworks para modelos capacidade de processo<sup>7</sup> e métodos de avaliação de processo são mais relevantes por sua utilização no setor de software e representação das arquiteturas dominantes: CMMI e ISO/IEC 15504. O framework é de certo modo um framework baseado no framework da ISO/IEC 15504. Nessa seção são apresentados exemplos de modelos desses frameworks, além de outros modelos importantes para a melhoria de processo de software. A Figura 24 identifica alguns dos modelos de capacidade de processo descritos nesta seção, acrescido de dois modelos incluídos por razões históricas (SEI SW-CMM e Bootstrap). Para cada modelo é indicada a data de início de seu desenvolvimento, as datas e identificações de suas versões principais e, em alguns casos, a data de aposentadoria do modelo.

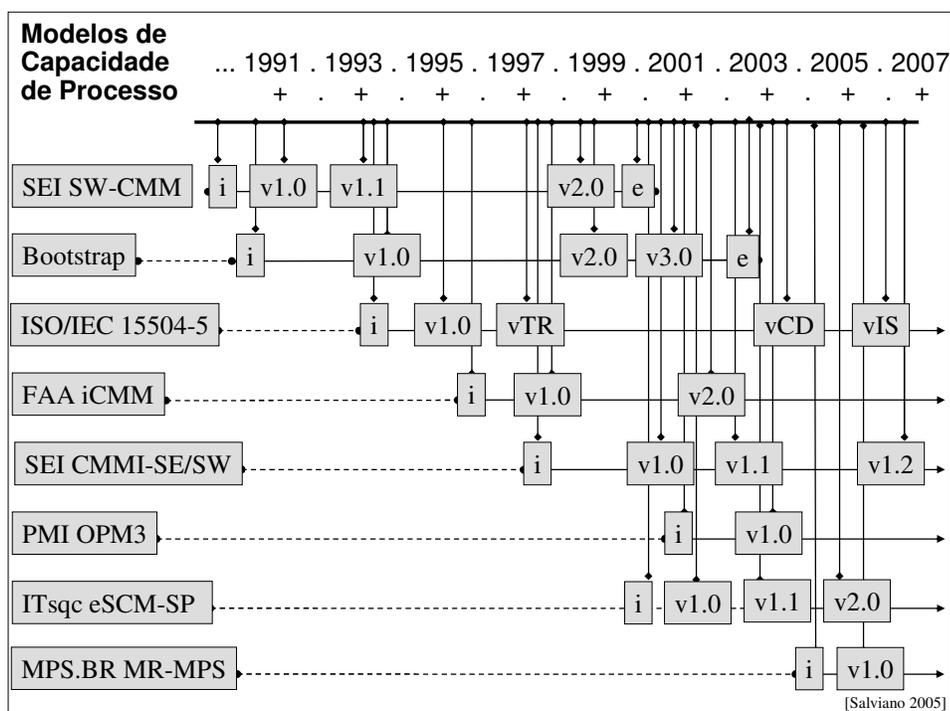


Figura 24 - Histórico e versões dos modelos de capacidade

<sup>7</sup> Alguns modelos acrescentam o termo “maturidade” para indicar o uso dos níveis de maturidade, como, por exemplo, os modelos SW-CMM e CMMI-SE/SW. Como esses modelos definem maturidade em função da capacidade de processo, o termo “capacidade” é suficiente para caracterizar também esses modelos.

### 2.3.1 Framework ISO/IEC 15504 e modelo ISO/IEC 15504-5

O termo “ISO/IEC 15504” designa a Norma Internacional ISO/IEC 15504 para Avaliação de Processos, desenvolvida pela ISO/IEC, com apoio do projeto SPICE (Software Process Improvement and Capability dEtermination) [El Emam et al. 1998, Salviano 2001a, ISO/IEC15504 2003]. Esse desenvolvimento foi iniciado em 1993 após a realização pela ISO de um estudo no qual foi concluído que existia um consenso internacional sobre a necessidade e os requisitos para uma Norma internacional de avaliação de processo de software e que deveria ser adotada uma forma de desenvolvimento na qual versões intermediárias fossem utilizadas pelo setor de software. Foi criado então o projeto SPICE com uma equipe de especialistas internacionais para apoiar o desenvolvimento das versões iniciais da futura Norma e coordenar a utilização destas versões pela comunidade. Devido a isto a Norma é também conhecida como SPICE.

A Figura 25 ilustra os principais elementos e relacionamentos do modelo conceitual da ISO/IEC 15504.

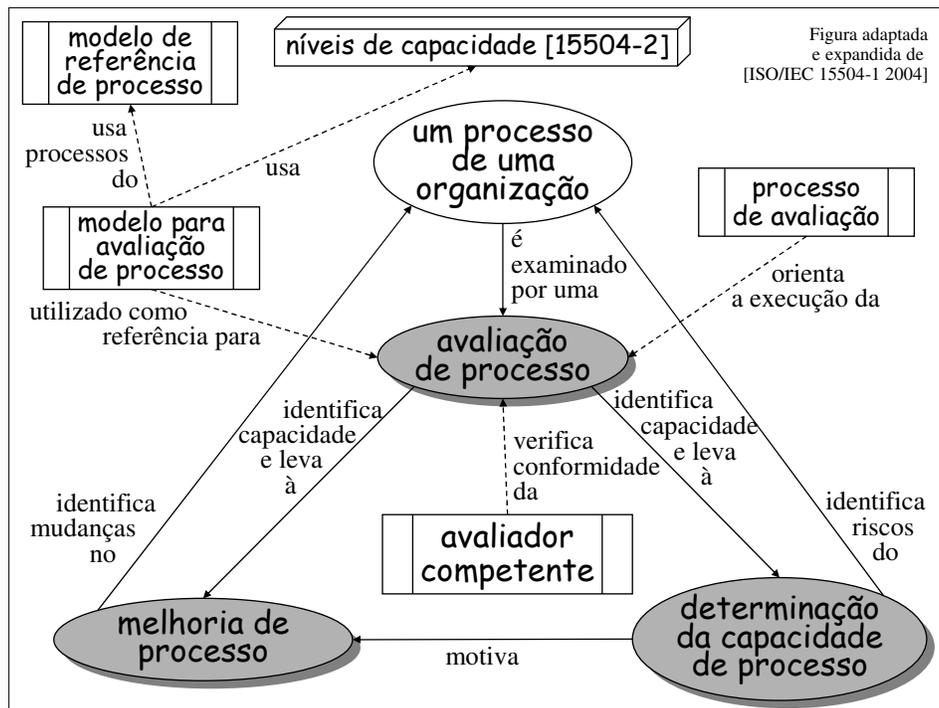


Figura 25 – Modelo conceitual da ISO/IEC 15504

Um processo de uma organização é examinado por uma avaliação de processo. Uma avaliação de processo examina de forma disciplinada um processo em relação a um Modelo para Avaliação de Processo, é realizada seguindo a orientação de um processo de avaliação e sua conformidade é

---

verificada por um avaliador competente. Um Modelo para Avaliação de Processo utiliza os níveis de capacidade definidos na ISO/IEC 15504-2 e processos de um ou mais modelos de referência de processo.

Uma avaliação de processo pode ser realizada no contexto de melhoria de processo ou de determinação da capacidade de processo. Na melhoria de processo, o resultado de uma avaliação identifica a capacidade e é utilizado para identificar mudanças no processo avaliado. Na determinação da capacidade, o resultado de uma avaliação também identifica a capacidade, mas é utilizado para identificar riscos do processo avaliado e também pode ser utilizado como motivação para a melhoria do processo.

Entre 1993 e 2003, a 15504 teve várias versões, nas quais houve um aumento de flexibilidade. Como marcos desta evolução podem ser destacadas as versões identificadas por SPICE V1.0 [Dorling 1995], ISO/IEC TR 15504 [ISO/IEC-TR15504 1998] e ISO/IEC 15504 [ISO/IEC 15504 2003]. Na versão SPICE V1.0, a então futura Norma, que ainda era conhecida apenas como SPICE, definia como normativo um modelo para avaliação de processos da engenharia de software, com um conjunto definido de processos, seis níveis de capacidade e indicadores para a avaliação. Na versão ISO/IEC TR 15504 foi abstraído do conceito de modelo para avaliação de processo, o conceito de modelo de referência de processo, com um conjunto de processos e níveis de capacidade, sem os indicadores para avaliação. Na versão TR foi mantido o foco na engenharia de software. Esse modelo de referência passou a ser normativo e foi mantido o modelo para avaliação de processo como um exemplo de modelo compatível com a futura Norma. Na versão ISO/IEC 15504, foram abstraídos do conceito de modelo de referência os níveis de capacidade, sem serem mais específicos para engenharia de software, e definidos os requisitos para um Modelo de Referência de Processo e um Modelo para Avaliação de Processo, também sem serem específicos para engenharia de software. A definição dos processos para engenharia de software foi evoluída e colocada na nova versão da ISO/IEC 12207 [2002] como um exemplo de um Modelo de Referência de Processo para a engenharia de software. O modelo exemplo para avaliação de processo para a engenharia de software foi mantido e evoluído, passando a referenciar os processos da nova versão da 12207. Com isto a versão ISO/IEC 15504 é genérica para qualquer área de tecnologia, inclusive engenharia de software.

Na prática existem duas “15504”. Uma é a Norma ISO/IEC 15504, também conhecida como Framework 15504 ou SPICE. Outra é o modelo exemplo, denominado de ISO/IEC 15504-5. Esta Norma está descrita em cinco documentos. Podemos dizer que as partes 2 e 3, descrevem, respectivamente, o framework e orientações para uso do framework, e as partes 5 e 4 descrevem,

respectivamente, o modelo exemplo e orientações para utilização desse e outros modelos para melhoria ou determinação da capacidade.

Segundo a 15504, o nível de capacidade de cada processo é consequência da pontuação dos atributos de processo. Cada atributo de processo é identificado pelo número do nível a qual pertence (1, 2, 3, 4 ou 5) e por um identificador no nível (.1 ou .2). Cada atributo representa uma característica associada ao nível. A partir das observações resultantes da avaliação é atribuída uma das seguintes quatro pontuações ao atributo: “N” (o atributo não foi atingido pelo processo), “P” (o atributo foi atingido apenas parcialmente), “L” (foi atingido largamente) ou “F” (completamente, em inglês, *fully*). Para estar em um nível de capacidade, um processo tem que ter pontuações “L” ou “F” nos atributos do nível e “F” em todos os atributos dos níveis anteriores.

A ISO/IEC 15504 define também um guia para orientação da melhoria de processo, tendo como referência um modelo de capacidade de processo e como uma das etapas a realização de uma avaliação de processo. Esse guia sugere oito etapas seqüenciais, que inicia com a identificação de estímulos para a melhoria e o exame das necessidades da organização. Em seguida existem ciclos de melhoria, nos quais um alvo da melhoria é identificada, uma avaliação das práticas correntes em relação esse alvo é realizada, um planejamento da melhoria é feito, seguido pela implementação, confirmação, manutenção e acompanhamento da melhoria.

A visão da ISO/IEC JTC1 SC7 para modelos de referência de processo, modelos para avaliação de processo e métodos de avaliação de processo pode ser interpretada pela função dos seguintes principais atores:

- a ISO/IEC 15504-2 define níveis de capacidade de processo, requisitos para modelos para avaliação, métodos de avaliação e outros;
- a ISO/IEC 12207 Amd. 2 define processos para engenharia de software, que podem ser utilizados como um modelo de referência de processo;
- a ISO/IEC 15504-5 define um modelo compatível exemplo, para engenharia de software, com um modelo de referência de processo baseado nos processos da ISO/IEC 12207 Amd. 2 e com os níveis de capacidade da ISO/IEC 15504-2;
- um organismo relevante em uma determinada área pode definir um modelo para avaliação de processo e/ou um método de avaliação, compatíveis com a ISO/IEC 15504, na sua área de atuação;
- e
- uma organização intensiva em uma determinada área de atuação pode então utilizar modelos para avaliação de processo e métodos de avaliação de processo desta área de atuação.

A Figura 26 relaciona modelos para avaliação de processo e métodos de avaliação de processo que seguem esta visão.

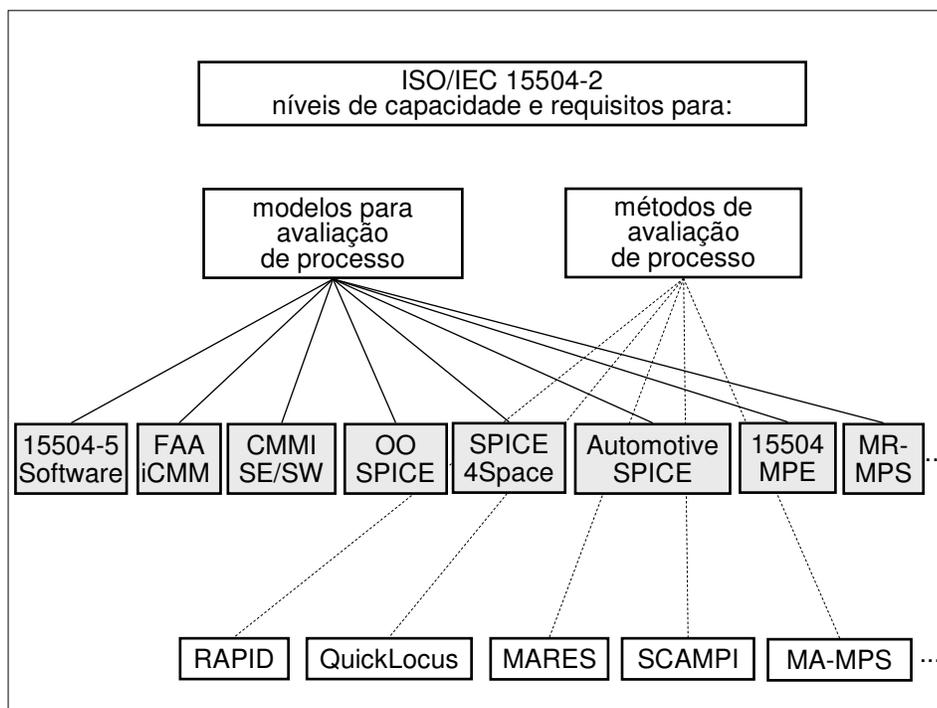


Figura 26 – Exemplos de modelos e métodos na visão da ISO/IEC JTC1 SC7

A ISO/IEC 15504-2 define níveis de capacidade de processo, requisitos para modelos para avaliação, métodos de avaliação, e outros. A ISO/IEC 12207 define processos para engenharia de software, que podem ser utilizados como um modelo de referência de processo. A ISO/IEC 15504-5 define um modelo compatível exemplo, para engenharia de software, com um modelo de referência de processo baseado nos processos da ISO/IEC 12207 e com os níveis de capacidade da ISO/IEC 15504-2. Um organismo relevante em uma determinada área pode definir um modelo para avaliação de processo e/ou um método de avaliação, compatíveis com a ISO/IEC 15504, na sua área de atuação. Uma organização intensiva em uma determinada área de atuação pode então utilizar modelos para avaliação de processo e métodos de avaliação de processo desta área de atuação.

Os modelos para avaliação de processo referenciados na Figura 26 são os modelos ISO/IEC 15504-5 [2006], FAA iCMM [Ibrahim et al. 2001], CMMI-SE/SW [Chrissis et al. 2003], OOSPICE [Stallinger et al. 2002], SPICE4SPACE [Völcker et al. 2002], Automotive SPICE [Automotive SIG 2005] e MR-MPS [Weber et al. 2005a]. Os métodos de avaliação de processo referenciados na Figura 26 são os métodos RAPID [Rout et al. 2000], QuickLocus [Kohan 2003], MARES [Anacleto et al. 2004a], SCAMPI [SEI 2001a] e MA-MPS [Weber et al. 2005a].

Em 1998, a 15504 foi publicada como Relatório Técnico pela ISO, denominada de *ISO/IEC TR 15504: Information Technology - Software Process Assessment* (Tecnologia da Informação - Avaliação de Processos de Software), e composta por 9 partes [ISO/IEC TR 15504 1998]. Em 2003, a 15504 foi publicada como Norma Internacional, com cinco partes.

A ISO/IEC 15504 criou a chamada arquitetura contínua de modelos de referência de processo. Nesta arquitetura, o modelo tem duas dimensões: Uma na qual são definidos processos (“dimensão de processos”) e outra na qual são definidos níveis de capacidade nos quais cada um dos processos pode estar sendo estabelecido em uma organização (“dimensão de capacidade”).

A Figura 27 relaciona a estrutura detalhada do modelo ISO/IEC 15504-5 com seus principais elementos, incluindo aqueles oriundos de suas principais referências: o PRM ISO/IEC 12207 Amd. 2 e a ISO/IEC 15504-2.

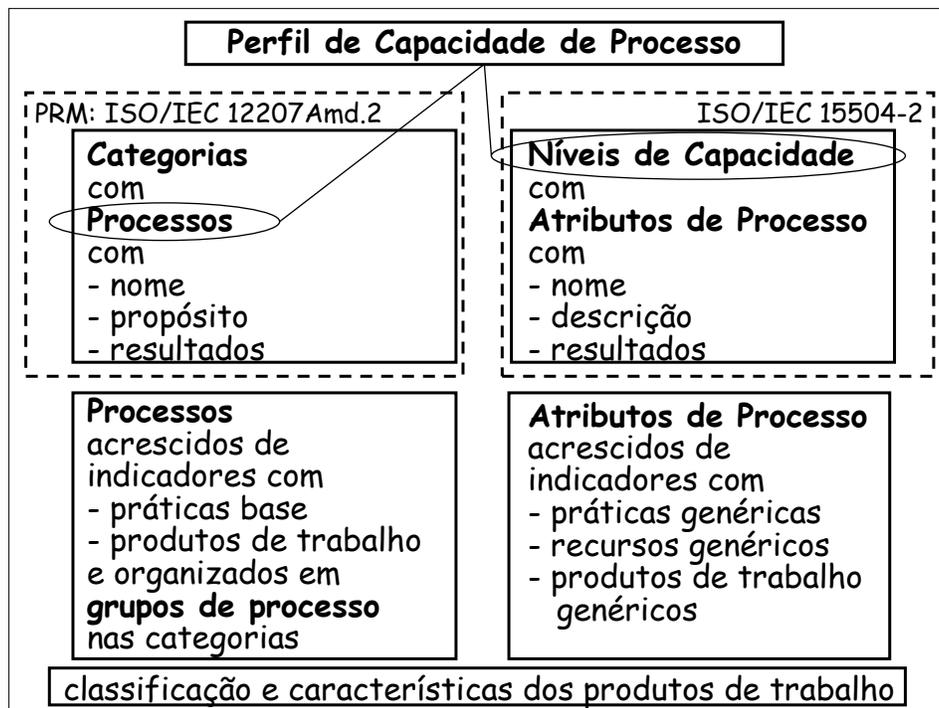


Figura 27 – Estrutura detalhada do modelo ISO/IEC 15504-5

A Figura 28 relaciona a estrutura simplificada do modelo ISO/IEC 15504-5, com a identificação e nome das três categorias de processo, nove grupos de processo, quarenta e oito processos, seis níveis de capacidade e nove atributos de processo.

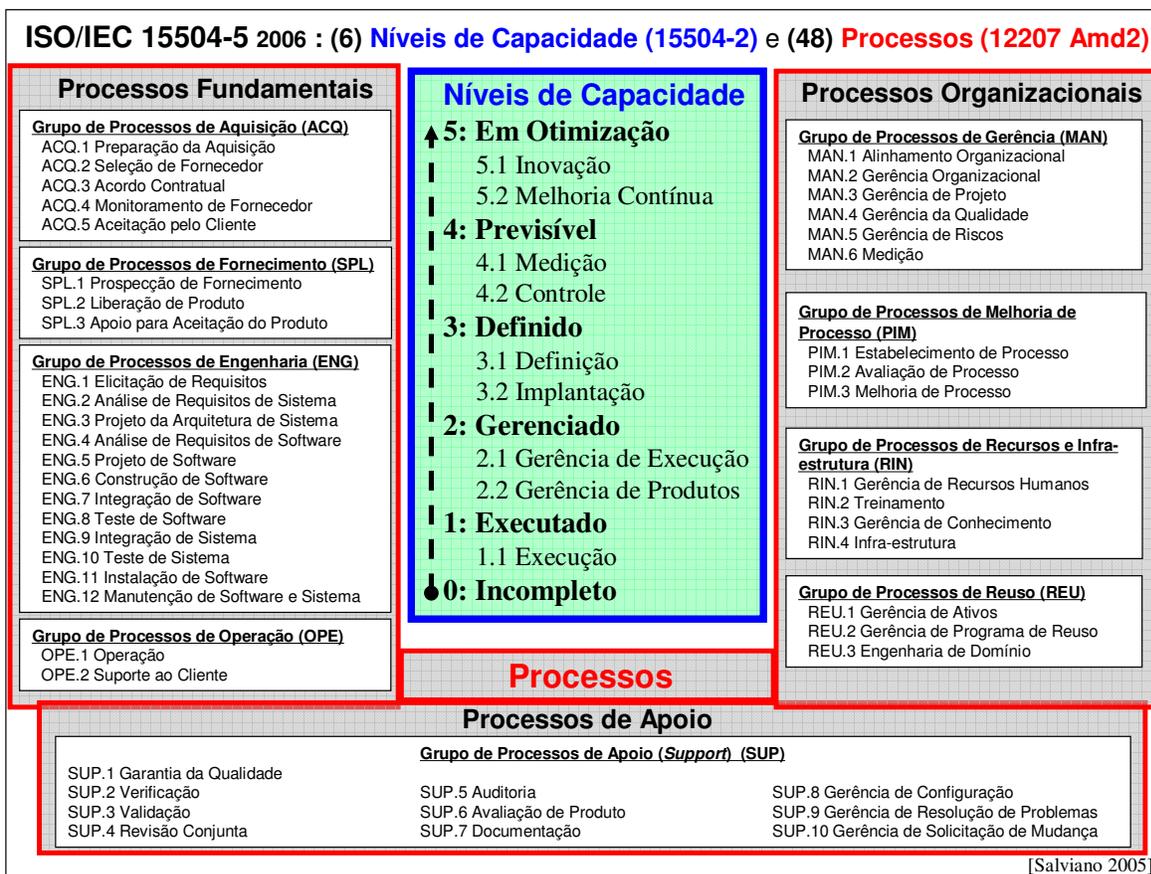


Figura 28 – Estrutura e elementos do modelo ISO/IEC 15504-5 [2006]

O modelo para avaliação de processo da ISO/IEC 15504-5 define 48 processos, universais e essenciais para as atividades de aquisição, fornecimento, desenvolvimento, manutenção, apoio, gerência e melhoria de software. Esses 48 processos estão organizados em 9 grupos que, por sua vez, estão organizados em 3 categorias. O modelo utiliza os seis níveis de capacidade definidos na ISO/IEC 15504-2.

### 2.3.2 Modelo FAA iCMM

Um esforço de integração de frameworks e modelos tem sido realizado pela Administração Federal de Aviação dos Estados Unidos (*Federal Aviation Administration* FAA) desde 1997, como base para o modelo FAA iCMM (*Integrated Capability Maturity Model*, Modelo Integrado da Maturidade da Capacidade) [Ibrahim 2000, Ibrahim et al. 2001]. A versão iCMM 2.0, publicada em 2001, integra várias normas, padrões, modelos e documentos, incluindo: EIA/IS 731 Systems Engineering 1998, ISO 9001:2000, President's Quality Award Program 2000, Malcolm Baldrige National Quality Award

Program 2000, CMMI-SE/SW/IPPD v1.02 2000, ISO/IEC CD 15288 2001, ISO/IEC 12207: 1995, ISO/IEC TR 15504:1998 (Parts 5 and 7), e CMMI-SE/SW/A (draft), 2000 [Ibrahim et al. 2001].

A Figura 29 ilustra a estrutura mais detalhada do modelo iCMM v2.0, composta por áreas de processo, níveis de capacidade, perfil de capacidade de processo, e exemplos desses perfis, como níveis de maturidade.

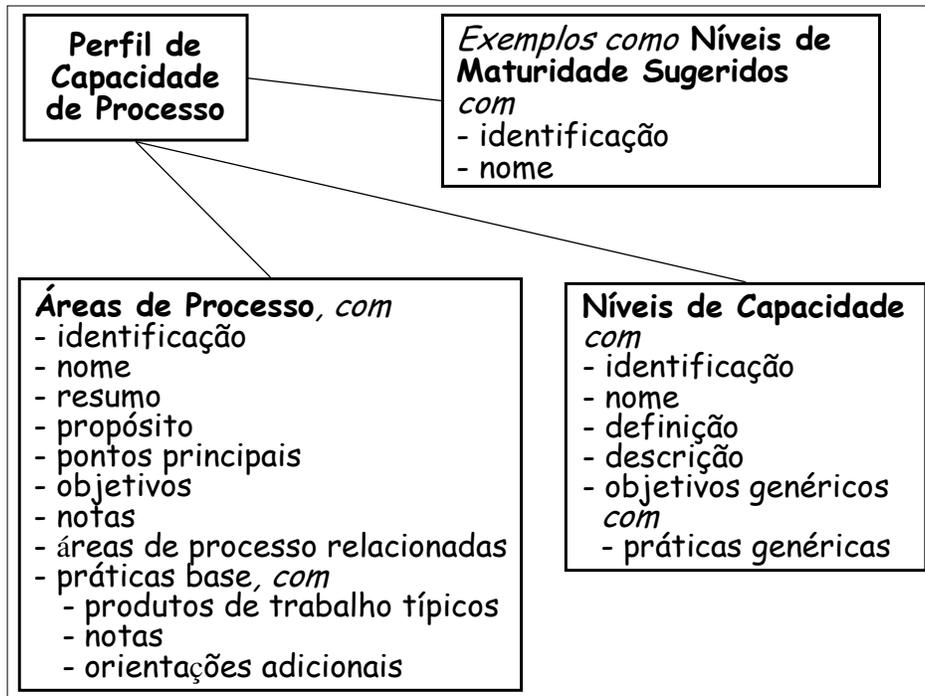


Figura 29 – Estrutura mais detalhada do modelo iCMM v2.0

A Figura 30 relaciona a estrutura simplificada do modelo iCMM versão 2.0, com a identificação e nome das três categorias de processo, dos vinte e três processos, dos seis níveis de capacidade e de nove dos outros modelos integrados nesta versão.

Uma das características do iCMM é ser um modelo contínuo, com uma sugestão de níveis de maturidade. A alocação das áreas de processo aos níveis de maturidade no iCMM foi realizado usando as seguintes orientações [Ibrahim et al. 2001, p. 2-37].

- Áreas de processo que explicam práticas genéricas do nível. Estas práticas são essenciais para a institucionalização do processo no nível.
- Áreas de processo que resultam da integração de áreas de processo de arquitetura estagiada desse nível de modelos do CMMI e outros modelos estagiados. Quando existe diferença no estágio que os modelos alocam as áreas de processo, o nível mais baixo utilizado entre os modelos é selecionado para o iCMM.

- Outras áreas de processo consideradas essenciais para estabelecer e sustentar a capacidade organizacional nesse nível.

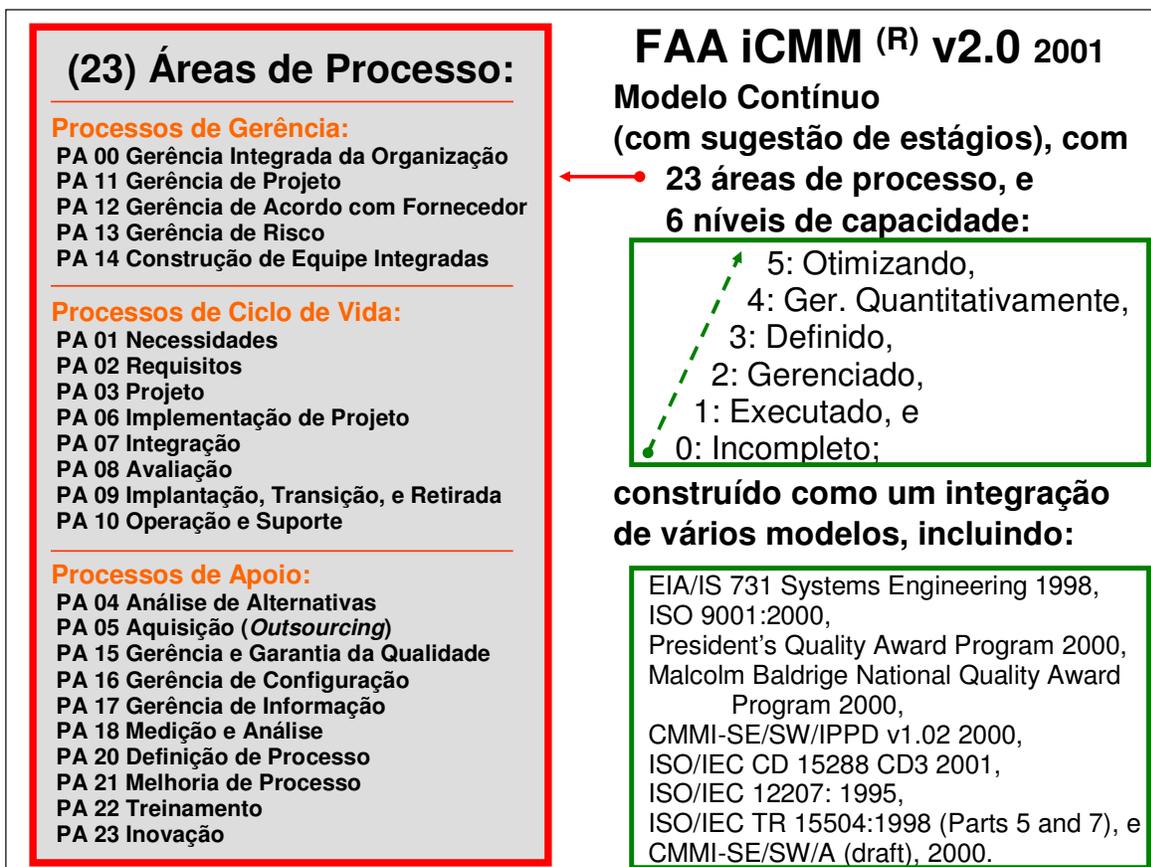


Figura 30 – Estrutura simplificada e elementos do modelo iCMM v2.0

Em geral, o grupo de áreas de processo em um nível de maturidade é considerado como uma fundação essencial para o próximo incremento do progresso refletido no nível de maturidade seguinte.

### 2.3.3 Framework SEI CMMI e modelo CMMI-SE/SW

A sigla CMMI representa as iniciais de *Capability Maturity Model Integration* (Integração de Modelos de Maturidade da Capacidade) e nomeia pelo menos três entidades relacionadas entre si: um projeto, o framework resultante desse projeto e os modelos desse framework [Chrissis et al. 2003].

O Projeto CMMI, que pode ser traduzido como “Projeto de Integração dos Modelos de Maturidade da Capacidade”, está sendo executado pelo *Software Engineering Institute* (SEI), em cooperação com a indústria e governo, para consolidar um framework de modelos, os modelos desse framework e seus produtos associados, incluindo material de treinamento e método de avaliação. O framework foi desenvolvido inicialmente baseado na evolução e integração de três modelos: a versão

2.0 do SW-CMM (*Capability Maturity Model for Software*), o SE-CMM: EIA 731 (*System Engineering Capability Maturity Model*) e o IPD-CMM (*Integrated Product Development Capability Maturity Model*).

Esta integração e evolução teve como objetivo principal a redução do custo da implementação de melhoria de processo multidisciplinar baseada em modelos. Multidisciplinar porque além da engenharia de software, o CMMI cobre também a engenharia de sistemas, aquisição, e a cadeia de desenvolvimento de produto. Esta redução de custo é obtida por meio da eliminação de inconsistências; redução de duplicações, melhoria da clareza e entendimento, utilização de terminologia comum, utilização de estilo consistente, estabelecimento de regras de construção uniforme, manutenção de componentes comuns, garantia da consistência com a Norma ISO/IEC 15504 e sensibilidade às implicações dos esforços legados [Ahern et al. 2001].

O Framework CMMI é composto basicamente por um repositório de dados e processos definidos para inserção de texto no repositório, para gerência do repositório, e para a geração de modelos de maturidade da capacidade e dos materiais de apoio para avaliação e treinamento. O repositório contém componentes comuns e compartilhados que são essenciais para todos os modelos, componentes compartilhados que são compartilhados por vários modelo, mas não todos os modelos; componentes únicos que são específicos para uma determinada disciplina; e regras para construção de modelos. Os produtos resultantes do framework incluem os modelos, materiais de treinamento e método de avaliação para uma disciplina e combinações de disciplinas. O framework provê o mecanismo necessário para integrar disciplinas adicionais no conjunto de produtos do CMMI no futuro.

Em 2002 foram lançados quatro modelos do Framework CMMI todos na versão 1.1. Cada modelo é identificado pelo CMMI seguido dos indicadores das disciplinas cobertas. Em cada modelo, a sigla CMMI pode ser traduzida para “Modelo Integrado de Maturidade da Capacidade”. Os quatro modelos são denominados de CMMI-SW, CMMI-SE/SW, CMMI-SE/SW/IPPD e CMMI-SE/SW/IPPD/SS, cobrindo composições das seguintes quatro disciplinas: Engenharia de Sistemas (*System Engineering SE*), Engenharia de Software (*Software Engineering SW*), Desenvolvimento de produtos e processos integrados (*Integrated Product and Process Development IPPD*) e Aquisição (*Supplier Sourcing SS*). Está previsto para julho de 2006 o lançamento das versões 1.2 relacionadas a esses quatro modelos.

A arquitetura dos modelos do CMMI é composta basicamente pela definição de um conjunto de áreas de processo, organizadas em duas representações diferentes: uma como um modelo estagiado, semelhante ao SW-CMM, e outra como um modelo contínuo (semelhante à ISO/IEC 15504).

A Figura 31 ilustra a estrutura detalhada dos modelos do CMMI versões 1.1 que é composta por níveis de maturidade, áreas de processo e níveis de capacidade.

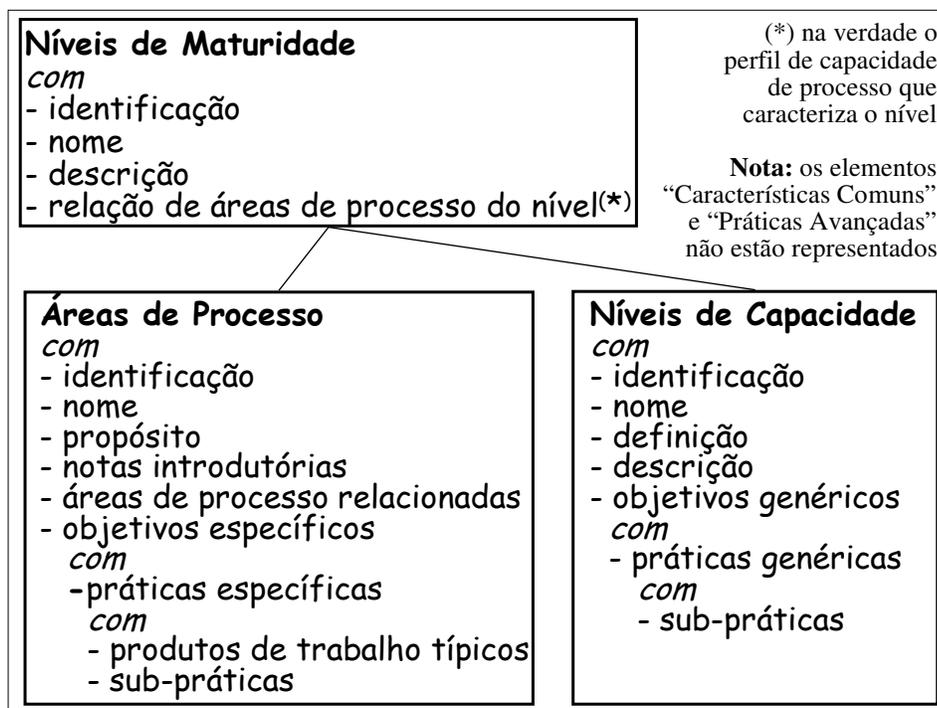


Figura 31 – Estrutura mais detalhada dos modelos do CMMI

Apesar dos níveis de maturidade serem definidos com uma relação de áreas de processo, esta definição indica na verdade um perfil de capacidade de processo com estas áreas de processo relacionadas e todas as áreas relacionadas nos níveis anteriores, todos em um determinado nível de capacidade. Dois elementos não foram representados na Figura 31: Características Comuns e Práticas Avançadas. Características Comuns é uma forma de organizar os objetivos genéricos e é utilizada apenas nas representação estagiadas. Práticas Avançadas são práticas que devem ser consideradas apenas em determinados níveis de capacidade e são utilizadas apenas nas representações contínua. Esses dois conceitos violam a característica de relacionamento entre as duas representações e estão sendo eliminados nas versões 1.2 que estão previstas para serem lançadas em julho de 2006.

A Figura 32 ilustra a estrutura simplificada do modelo CMMI-SE/SW versão 1.1 com a identificação e nome das quatro categorias de processo, vinte e duas áreas de processo, seis níveis de capacidade e os cinco níveis de maturidade, com a relação das áreas de processo em cada nível.

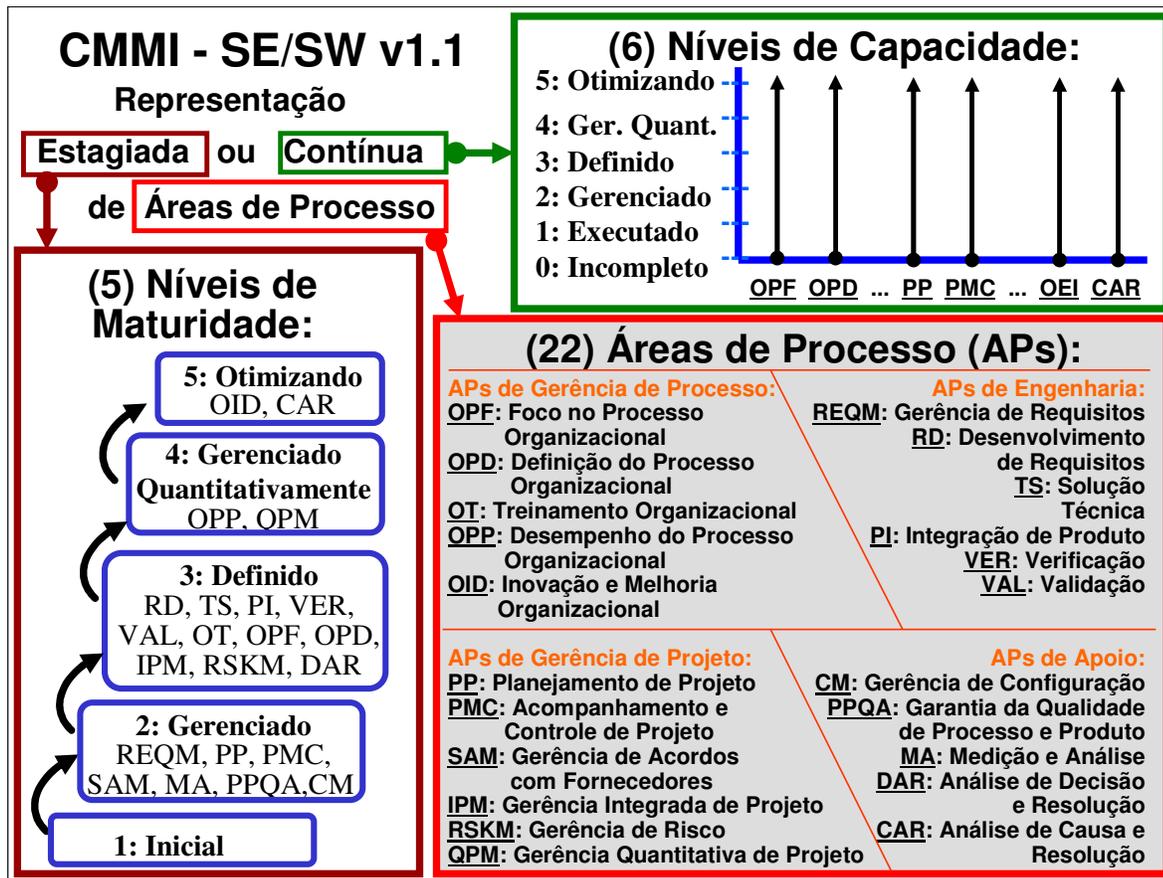


Figura 32 – Estrutura simplificada e elementos do modelo CMMI-SE/SW v1.1

As 22 áreas de processo do modelo CMMI-SE/SW v1.1 estão agrupadas em quatro grupos (gerência de processos, gerência de projetos, engenharia e suporte) e estão identificadas na Figura 32 por uma sigla (baseada no nome em inglês) e uma tradução do nome para o português.

Na representação estagiada, do modelo CMMI-SE/SW, as 22 áreas de processo estão agrupadas em 4 níveis de maturidade: níveis 2, 3, 4 e 5 (o nível 1 não contém nenhuma área de processo) [SEI 2002b]. Em relação a esta representação, o processo de desenvolvimento de software ou sistema de uma unidade organizacional, pode estar classificado em um dos seguintes cinco níveis de maturidade: nível 1 - Inicial (*Initial*), nível 2 - Gerenciado (*Managed*), nível 3: Definido (*Defined*), nível 4: Gerenciado Quantitativamente (*Quantitatively Managed*), e nível 5: Em Otimização (*Optimizing*).

Na representação contínua, as mesmas 22 áreas de processo são definidos seis níveis de capacidade de processo [SEI 2002a]. Nesta representação, o conjunto de atividades correspondente a cada uma destas áreas de processo, pode ter sua capacidade de execução classificada em um dos seguintes seis níveis de capacidade de processo: nível 0: Incompleto (*Incomplete*), nível 1: Executado (*Performed*), nível 2: Gerenciado (*Managed*), nível 3: Definido (*Defined*), nível 4: Gerenciado

Quantitativamente (*Quantitatively Managed*), e nível 5: Em Otimização (*Optimizing*). Cada nível de capacidade é definido por um conjunto de características que o processo deve satisfazer para estar naquele nível.

As versões 1.2 correspondentes à evolução das versões 1.1 estão planejadas para serem lançadas em julho de 2006 e as principais alterações são as seguintes [Phillips 2005, slides 24-25]:

- eliminação dos conceitos de prática avançada e característica comum,
- combinação da área de processo ISM (que existia apenas no modelo CMMI-SE/SW/IPPD/SS) com a área de processo e eliminação do modelo CMMI-SE/SW/IPPD/SS,
- adição de orientações para uso para hardware,
- reconhecimento, com a adição de hardware, que provê modelos de desenvolvimento separados não é mais útil,
- documentação dos modelos e de suas duas representações em um único documento, semelhante à adotada em Chrissis et al. [2003],
- a caracterização como não aplicável de áreas de processo para os níveis de maturidade serão restringidas de forma significativa, como por exemplo, para SAM,
- melhoria do texto em geral baseado nas mais de mil solicitações de mudança, como, por exemplo, melhoria da descrição dos níveis mais altos de maturidade e da terminologia de avaliação,
- adição de duas práticas para ambientes de trabalho específicos, uma para OPD relacionada a característica organizacional e outra para IPM relacionada a característica de projeto,
- melhoria do glossário, como, por exemplo, gerência de alto nível, rastreabilidade bidirecional e sub-processo,
- melhoria do texto de introdução, e
- consolidação e simplificação da cobertura de IPPD.

### **2.3.4 MPS.BR e modelo MR-MPS**

MPS-BR (Melhoria de Processo do Software Brasileiro) é uma iniciativa com dois objetivos iniciais principais:

- “Desenvolver e aprimorar um Modelo de Referência de Processo (MR-MPS) e um Método de Avaliação de Processo (MA-MPS, em conformidade com as Normas ISO/IEC 12207 e ISO/IEC 15504, mantendo compatibilidade com os modelos do CMMI; e

- Disseminar o modelo MR-MPS em organizações públicas e privadas, com foco na grande massa de micro, pequenas e médias empresas, em todas as regiões brasileiras, a um custo acessível” [Weber et al. 2005a]

Segundo Weber et al. [2004a] “não é objetivo do Projeto MPS.BR definir algo novo no que se refere a normas e modelos. Sua novidade está na estratégia de implementação, criada para a realidade brasileira”.

A Figura 33 ilustra a estrutura mais detalhada dos elementos do modelo MR-MPS versão 1.0.

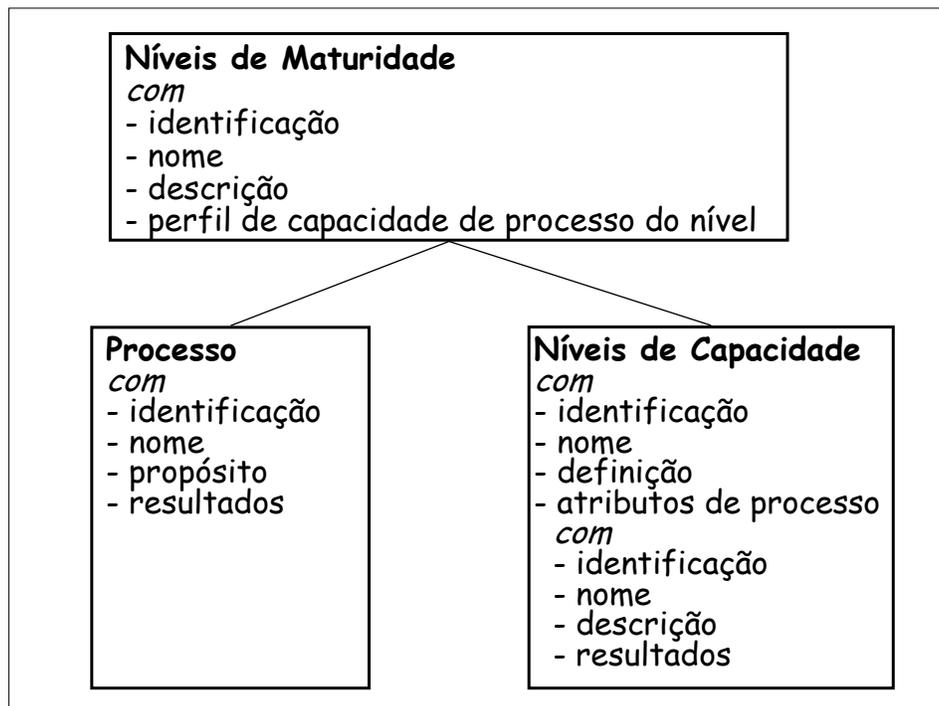


Figura 33 – Estrutura mais detalhada dos elementos do modelo MR-MPS v1.0

Baseado em uma visão que a maioria das empresas tinha uma preferência pelos modelos estagiados com SW-CMM e do CMMI, que o custo de uma avaliação em relação a esses modelos é alto, que principalmente as micros e pequenas empresas precisam de níveis de maturidade menores, o modelo MR-MPS foi definido como uma utilização dos processos das Normas ISO/IEC 12207 Amd2 e da ISO/IEC 15504-5, mas particularmente da versão CD2, por meio de sete níveis de maturidade, seqüenciais e cumulativos, representados como perfis de capacidade de processo, segundo a ISO/IEC 15504.

A Figura 34 ilustra a estrutura simplificada e os elementos do modelo MR-MPS versão 1.0.

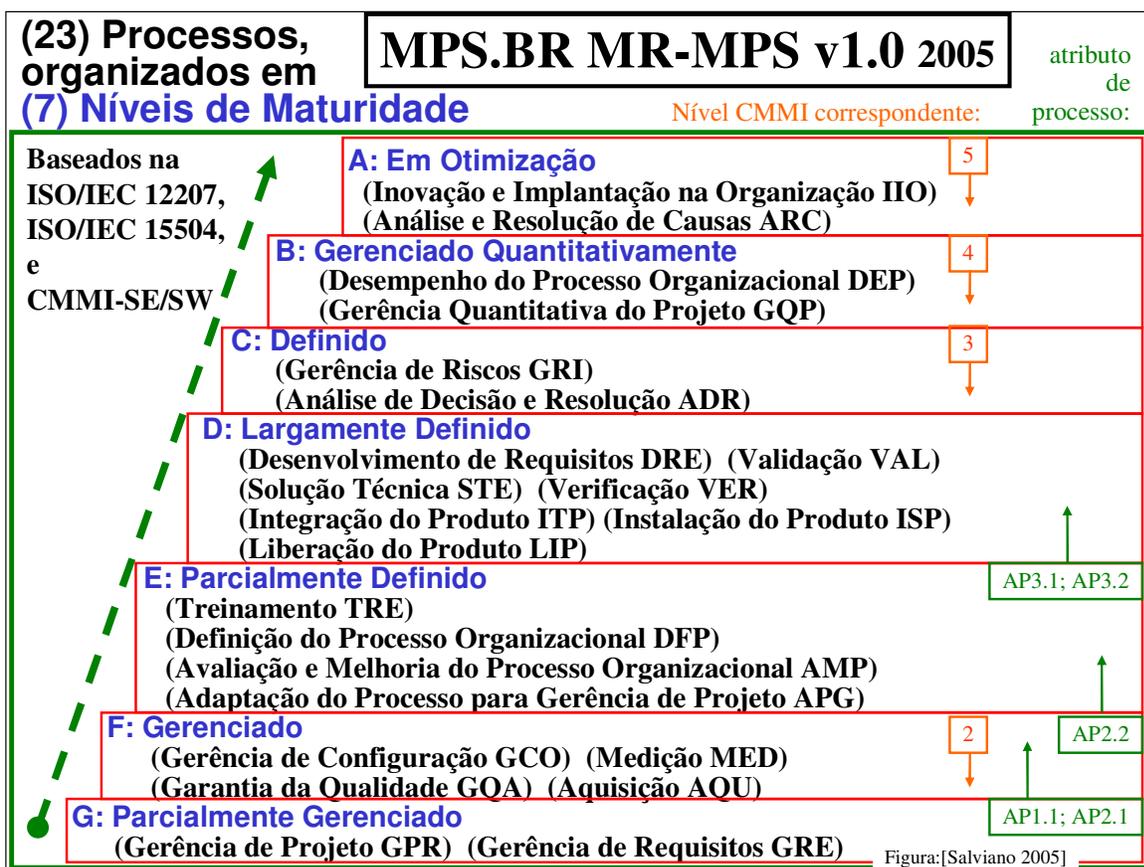


Figura 34 – Estrutura simplificada e elementos do modelo MR-MPS v1.0

O MPS.BR está sendo desenvolvido por sete instituições-âncora: SOFTEX (coordenadora do projeto), COPPE/UFRJ (coordenadora da Equipe Técnica do Modelo), RIOSOFT, CenPRA, Núcleo SOFTEX Campinas, CESAR e CELEPAR.

### 2.3.5 Outros modelos de capacidade de processo

Esta seção apresenta mais sete modelos de capacidade de processo.

OPM3 (*Organizational Project Management Maturity Model*, Modelo de Maturidade da Gerência Organizacional de Projeto) é um modelo do PMI para a gerência organizacional por projetos [PMI 2003]. Gerência Organizacional por Projetos é a aplicação de conhecimento, habilidades, ferramentas e técnicas para atividades organizacionais e de projeto para atingir os objetivos de uma organização por meio de projetos. OPM3 combina atividades de gerência de projeto, gerência de programas e gerência de portfólios de projetos e programas para realizar uma estratégia organizacional por meios de projetos. A versão do modelo OPM3 publicada em novembro de 2003 contém 590 melhores práticas em gerência organizacional, organizadas em duas dimensões: uma de

domínios, com gerência de projeto, gerência de programa e gerência de portfólio; e outra de níveis de maturidade, com padronização, medição, controle e melhoria contínua.

O modelo eSCM-SP (*eSourcing Capability Model for Service Providers*, Modelo de Capacidade de *eSourcing* para Provedores de Serviços) está sendo desenvolvido pelo ITsqc (*Information Technology Service Qualification Center*, Centro de Qualificação de Serviços em Tecnologia da Informação) [Hyder et al. 2004a, Hyder et al. 2004b]. O modelo ITsqc eSCM-SP v2, define 84 práticas, organizadas em 3 dimensões: ciclo de vida, área de capacidade e nível de capacidade. A dimensão de ciclo de vida é composta por inicia, entrega, completa, e durante. A dimensão de área de capacidade é composta por gerência de conhecimento, gerência de pessoas, gerência de desempenho, gerência de relacionamentos, gerência de tecnologia, gerência de ameaças, contratação, projeto e implantação de serviços, entrega de serviços, e transferência de serviços. A dimensão de nível de capacidade é composta por cinco níveis seqüenciais e cumulativos: (1) provê serviços, (2) consistentemente satisfaz requisitos, (3) gerencia desempenho organizacional, (4) agrega valor proativamente, e (5) sustenta excelência.

COBIT (*Control Objectives for Information and Related Technology*, Objetivos de Controle para Tecnologias de Informação e Relacionadas) representa um conjunto de melhores práticas e orientações largamente adotadas, publicada pelo *IT Governance Institute* [ITGI 2005]. COBIT versão 4.0 especifica 34 processos da tecnologia de informação distribuídos em quatro domínios: planejamento e organização, aquisição e implementação, entrega e apoio, e, monitoramento e avaliação. No COBIT existe um relacionamento entre os quatro domínios e com características da informação. A informação interage com os objetivos estratégicos e de governança, e também com a infra-estrutura de TI. Para os 34 processos o COBIT define objetivos de controle em alto nível, orientações para gerenciamentos, objetivos de controle detalhados e um modelo de maturidade para medir e controlar o sucesso dos processos. O modelo de maturidade é composto por seis níveis de capacidade seqüenciais e cumulativos, que podem ser considerados como “níveis de capacidade”. Esses níveis são, em ordem decrescente: 5: otimizado (*optimised*), 4: gerenciado e medido, 3: processo definido, 2: repetível mas intuitivo, 1: inicial / “ad-hoc”, e 0: não existente.

O modelo SM<sup>MM</sup> (*Software Maintenance Capability Maturity Model*, Modelo de Maturidade da Capacidade de Manutenção de Software), versão 2.0, contem quatro domínios de processo, dezoito áreas chave de processo, setenta e quatro *roadmaps* e 443 práticas [April et al. 2004]. Algumas áreas chave de processo são específicas para manutenção de software, outras são derivadas dos modelos do CMMI e de outros modelos, e ajustadas para a manutenção de software.

KMMM (*Knowledge Management Maturity Model*, Modelo de Maturidade de Gerência de Conhecimento) é uma metodologia desenvolvida pelo centro de competência em gerência de conhecimento da Siemens AG [Ehms e Langen 2002]. KMMM é composto por três componentes: um modelo de desenvolvimento, um modelo de análise e um processo de auditoria. O modelo de desenvolvimento define cinco níveis de maturidade, baseados nos níveis de maturidade do modelo SW-CMM, e com os mesmos nomes: inicial, repetível, definido, gerenciado e em otimização. Como esses níveis de maturidade estão definidos independentemente de qualquer área ou condição específica de gerência de conhecimento, em termos da terminologia da ISO/IEC 15504, eles estão mais próximos do conceito de níveis de capacidade. O modelo de análise é composto por oito áreas chave, baseado no modelo EFQM (*European Foundation for Quality Management*) [EFQM 2003]. Essas oito áreas são: estratégia e metas de conhecimento, ambientes e parcerias, pessoas e competências, colaboração e cultura, liderança e apoio, estruturas e formas de conhecimento, tecnologia e infra-estrutura, e processos, papéis e organização.

PMMM (*Project Management Maturity Model*, Modelo de Maturidade da Gerência de Projeto) é um modelo desenvolvido pelo *PM Solutions* para medir a maturidade da gerência de projeto em uma organização [Crawford 2002]. O PMMM segue os cinco níveis de maturidade do modelo SW-CMM para examinar a maturidade de cada uma das nove áreas de conhecimento do PMBOK.

UMM (*Usability Maturity Model*, Modelo de Maturidade de Usabilidade) é um modelo para processos do desenvolvimento centrado em humanos (HCD: *Human-Centred Development*, Desenvolvimento Centrado no Ser Humano) [Earthy 2000]. O UMM utiliza os seis níveis de capacidade de processo da ISO/IEC 15504 e define oito processos: garantir conteúdo de HCD na estratégia de sistemas, planejar e gerenciar processo de HCD, especificar os requisitos dos usuários e organizacionais, entender e especificar o contexto de uso, produzir soluções de projeto, avaliar projetos em relação aos requisitos, e introduzir e operar o sistema.

### **2.3.6 Outros modelos de referência utilizados para melhoria de processo**

Esta seção descreve outros modelos, que mesmo não sendo modelos de capacidade de processo, têm sido utilizados como referências adicionais para melhoria de processo. Esses modelos são ISO 9001, EFQM, PNQ, ISO/IEC 12207, PMBOK, RUP e SWEBOK. Três modelos representativos da área de excelência organizacional são os modelos da ISO 9001, EFQM e PNQ, cujas estruturas simplificadas e elementos principais estão ilustrados na Figura 35.

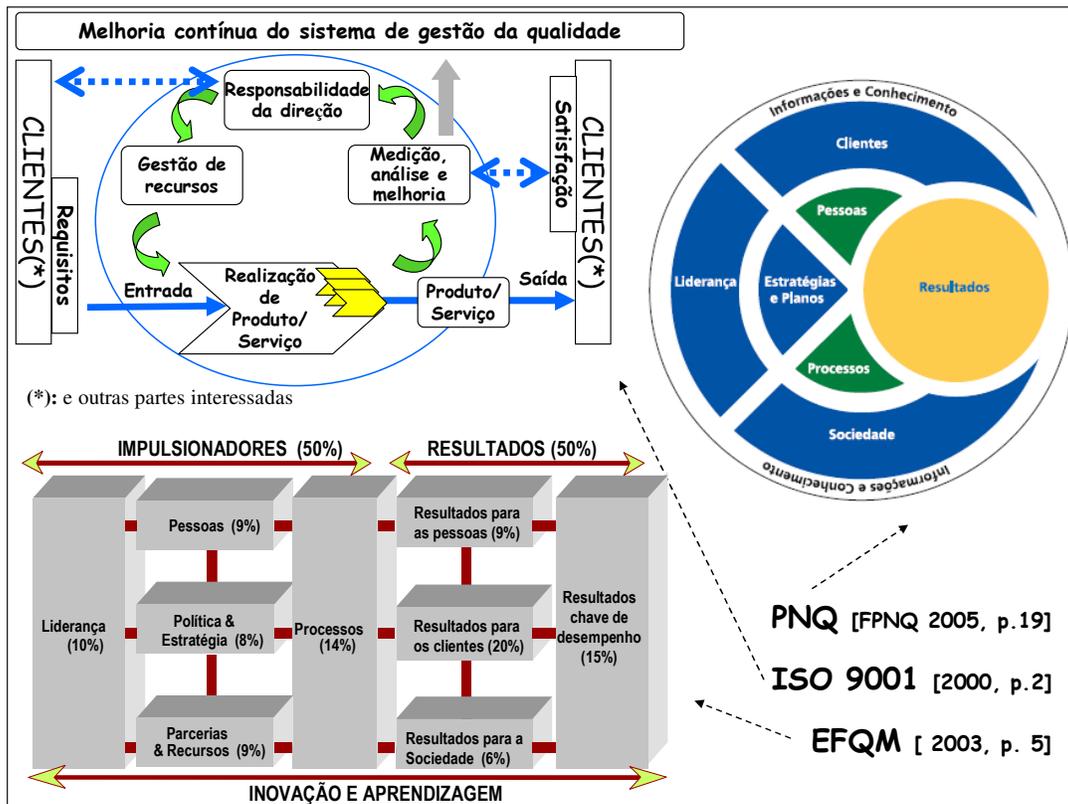


Figura 35 – Modelos de excelência da ISO 9001, EFQM e PNQ

As normas da série ISO 9000, com o par coerente ISO 9001 [2000] e ISO 9004 [2000] foram desenvolvidas para apoiar organizações, de todos os tipos e tamanhos, na implementação e operação de sistemas da qualidade eficazes. A Norma ISO 9001:2000, por exemplo, especifica requisitos para um sistema de gestão da qualidade. Sua aplicação é ampla, abrangendo tanto empresas ou unidades de manufatura como de serviços, incluindo as que desenvolvem, mantêm e fornecem software. A ISO 9001 está baseada em quatro princípios chave: satisfação do cliente, abordagem de processo, indicadores de desempenho e melhoria contínua.

As seções 4 a 8 da ISO 9001 [2000] definem um modelo para um sistema de gestão da qualidade baseado em processo, com o total de 23 requisitos. A seção 4, sistema de gestão da qualidade, define requisitos gerais (4.1) e requisitos de documentação (4.2). A seção 5, responsabilidade da direção, define os requisitos para comprometimento da direção (5.1), foco no cliente (5.2), política da qualidade (5.3) planejamento da qualidade (5.4), responsabilidade, autoridade e comunicação (5.5), e análise crítica pela direção (5.6). A seção 6, gestão de recursos, define requisitos para provisão de recursos (6.1), recursos humanos (6.2), infra-estrutura (6.3) e ambiente de trabalho (6.4) para a qualidade. A seção 7, realização do produto, define requisitos para planejamento da realização do

produto (7.1), processos relacionados ao cliente (7.2), projeto e desenvolvimento do produto (7.3), aquisição (7.4), produção e fornecimento de serviço (7.5) e controle de dispositivos de medição e monitoramento (7.6). A seção 8, medição, análise e melhoria, define requisitos para generalidades (8.1), medição e monitoramento (8.2), controle de produto não-conforme (8.3), análise de dados (8.4) e melhorias (8.5).

O EFQM (*European Foundation for Quality Management*, Fundação Européia para Gestão da Qualidade) define um Modelo de Excelência baseado em oito conceitos: (1) orientação para os resultados, (2) foco no cliente, (3) liderança e constância de propósitos, (4) gestão por processos e por fatos, (5) desenvolvimento e envolvimento das pessoas, (6) aprendizagem, melhoria e inovação contínuas, (7) desenvolvimento de parcerias, e (8) responsabilidade social corporativa [EFQM 2003]. Para cada conceito são descritas uma definição, orientações sobre como é colocado na prática, relação de benefícios e caracterização de três níveis para responder à pergunta “onde uma organização se posiciona atualmente em relação ao conceito”. Esses três níveis são “início”, “a caminho” e “madura”. Para o conceito de “gestão por processos e por fatos”, por exemplo, são definidos: (i) início: os processos para alcançar os resultados desejados estão definidos; (ii) a caminho: a informação e os dados comparativos são utilizados para estabelecer objetivos ambiciosos; e madura: a capacidade dos processos é completamente compreendida e utilizada para conduzir a melhoria do desempenho.

O PNQ (Prêmio Nacional da Qualidade) é um modelo sistêmico da gestão, desenvolvido pela Fundação Prêmio Nacional da Qualidade [FPNQ 2005]. Os critérios de excelência do PNQ são baseados em doze conceitos: (1) visão sistêmica, (2) aprendizado organizacional, (3) proatividade, (4) inovação, (5) liderança e constância de propósitos, (6) visão de futuro, (7) foco no cliente e no mercado, (8) responsabilidade social, (9) gestão baseada em fatos, (10) valorização das pessoas, (11) abordagem por processos, e (12) orientação por resultados. Com base nesses doze conceitos, o PNQ define oito critérios: (1) liderança, (2) estratégias e planos, (3) clientes, (4) sociedade, (5) informações e conhecimento, (6) pessoas, (7) processos, e (8) resultados.

A Norma ISO/IEC 12207 - Processos de Ciclo de Vida de Software [ABNT 1998, ISO/IEC 12207 2002] tem como objetivo principal o estabelecimento de uma estrutura comum para os processos de ciclo de vida de software como forma de ajudar as organizações a compreenderem todos os componentes presentes na aquisição e fornecimento de software e assim, conseguirem firmar contratos e executarem projetos de forma mais eficaz.

PMBOK (*Guide to the Project Management Body of Knowledge*, Guia para o Corpo de Conhecimento da Gerência de Projeto) é um guia para o conhecimento que caracteriza a disciplina de gerência de projeto, desenvolvido pelo *Project Management Institute* (PMI) [PMI 2004] [Figura 36].

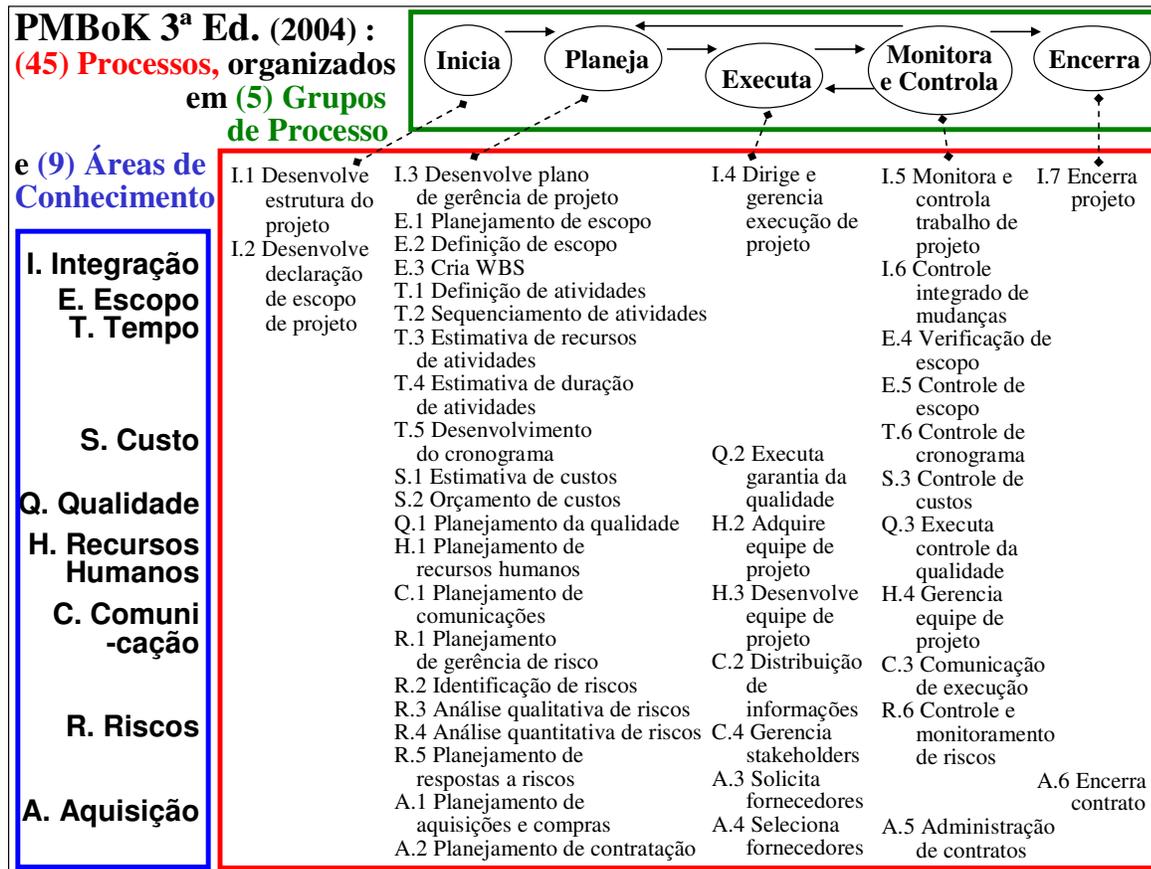


Figura 36 - Estrutura simplificada e elementos do modelo PMBOK Third Edition

O PMBOK define 45 processos que são classificados em duas dimensões: uma por nove áreas de conhecimento e outra por cinco grupos de processo. As nove áreas de conhecimento são respectivamente integração, escopo, tempo, custo, qualidade, recursos humanos, comunicação, riscos e aquisição. Os cinco grupos de processo são: inicia, planeja, executa, monitora e controla, e encerra. Os 45 processos estão identificados na Figura 36.

O RUP (*Rational Unified Process*, Processo Unificado da Rational) é um processo genérico para desenvolvimento de software, fundamentado em um conjunto de seis melhores práticas [Booch et al. 1999]. Estas melhores práticas são: (a) desenvolver software de forma iterativa, (b) gerenciar os requisitos, (c) utilizar arquitetura baseada em componentes, (d) modelar o software de forma visual, (e) verificar a qualidade do software, e (e) controlar as mudanças do software. As características mais importantes da forma iterativa de desenvolvimento são: (i) tratamento dos riscos durante o progresso,

---

o qual é medido em termos de produtos, e não de documentação ou estimativas de engenharia, (ii) integração contínua, (iii) liberações frequentes executáveis, e (iv) envolvimento contínuo do usuário final. Os elementos e estrutura simplificada do RUP estão organizados em termos de nove disciplinas e quatro fases. As nove disciplinas são respectivamente, modelagem de negócio, requisitos, análise e projeto, implementação, teste, implantação, gerência de configuração, gerência de projeto, e ambiente. As quatro fases são respectivamente iniciação, elaboração, construção e transição.

O SWEBOK (*Guide to the Software Engineering Body of Knowledge*, Guia para o Corpo de Conhecimento da Engenharia de Software) é um guia para o núcleo do conhecimento que caracteriza a disciplina de engenharia de software [Abran et al. 2004]. O desenvolvimento do SWEBOK está sendo coordenado por um projeto da *IEEE Computer Society* e ACM e busca cinco objetivos:

- caracterizar o conteúdo da disciplina de engenharia de software;
- prover tópicos de acesso à disciplina de engenharia de software;
- promover uma visão mundial consistente da engenharia de software;
- esclarecer o escopo e definir os limites da engenharia de software com respeito a outras disciplinas como ciência da computação, gerência de projeto, engenharia da computação e matemática; e
- prover uma fundação para o desenvolvimento de currículos e material para certificações individuais.

O projeto foi iniciado em 1993. A versão 2004 do SWEBOK identifica dez áreas de conhecimento da engenharia de software: requisitos de software, projeto de software, construção de software, teste de software, manutenção de software, gerência de configuração de software, gerência de engenharia de software, processos da engenharia de software, métodos e ferramentas da engenharia de software, e qualidade de software. Cada área é então dividida em sub-áreas. As sub-áreas da área de engenharia de processo de software, por exemplo, são mudança e implementação de processo, definição de processo, avaliação de processo, medição de processo e produto. O SWEBOK também identifica oito outras disciplinas relacionadas à engenharia de software: engenharia da computação, ciência da computação, gerência, matemática, gerência de projeto, gerência da qualidade, ergonomia de software e engenharia de sistemas.

## 2.4 Tendências de modelos e abordagens para melhoria de processo

Esta seção descreve as principais tendências de pesquisa e desenvolvimento identificadas como relevantes para este trabalho. Estas tendências estão relacionadas geralmente a composições das arquiteturas estagiada e contínua.

### 2.4.1 Relacionamentos e integração de modelos

A Tabela 11 indica referências para comparações e mapeamento entre os oito principais modelos considerados.

Tabela 11 – Referências para comparações diretas entre modelos

Modelos:	Referências:												
	R01	R02	R03	R04	R05	R06	R07	R08	R09	R10	R11	R12	R13
01: SEI SW-CMM v1.1	X	X				X					X		X
02: SEI CMMI-SE/SW E* v1.1	X			X	X	X		X	X	X	X	X	
03: SEI CMMI-SE/SW C* v1.1	X			X	X	X		X	X	X	X	X	
04: ISO/IEC 15504-5	X	X	X	X	X	X	X			X			
05: FAA iCMM v2.0	X					X							
06: MPS-BR MR-MPS v1.0										X			
07: ISO 9001 : 2000	X	X				X	X	X	X				X
08: PMI PMBOK Third Edition												X	
09: Outros	X	X	X			X			X	X	X		

Na Tabela 11 as referências citadas são respectivamente: R01 [Sheard 2001, Croll 2003], R02 [Halvorsen e Conradi 1999], R03 [Salviano 2000], R04 [Lepasaar et al. 2000, Lepasaar e Mäkinen 2001], R05 [Rout et al. 2001], R06 [Ibrahim et al. 2001], R07 [Dorling et al. 2003], R08 [Mutafelija e Stromberg 2003], R09 [Hyder et al. 2004], R10 [Weber et al. 2005a], R11 [Chrissis et al. 2003], R12 [Sherer e Thrasher 2005, Sheakley 2002] e R13 [Paulk et al. 1995].

Um mapeamento entre vários frameworks, modelos, métodos e guias, incluindo SW-CMM, CMMI, FAA iCMM, ISO/IEC 15504, ISO 9000 e ISO/IEC 12207, tem sido realizado pelo SPC (*Software Quality Consortium*, Consórcio de Qualidade de Software), e o resultado desde mapeamento é conhecido por Quagmire Framework [Sheard 2001]. Esse mapeamento é representado pelo diagrama reproduzido na Figura 37.

Uma ferramenta proprietária da SPC foi construída para indicar relacionamentos entre os frameworks e modelos: QuagTool [Croll 2003]. Como resultado desta comparação, quatro tendências foram identificadas: evolução, proliferação, integração e coordenação, e consolidação [Sheard 2001].

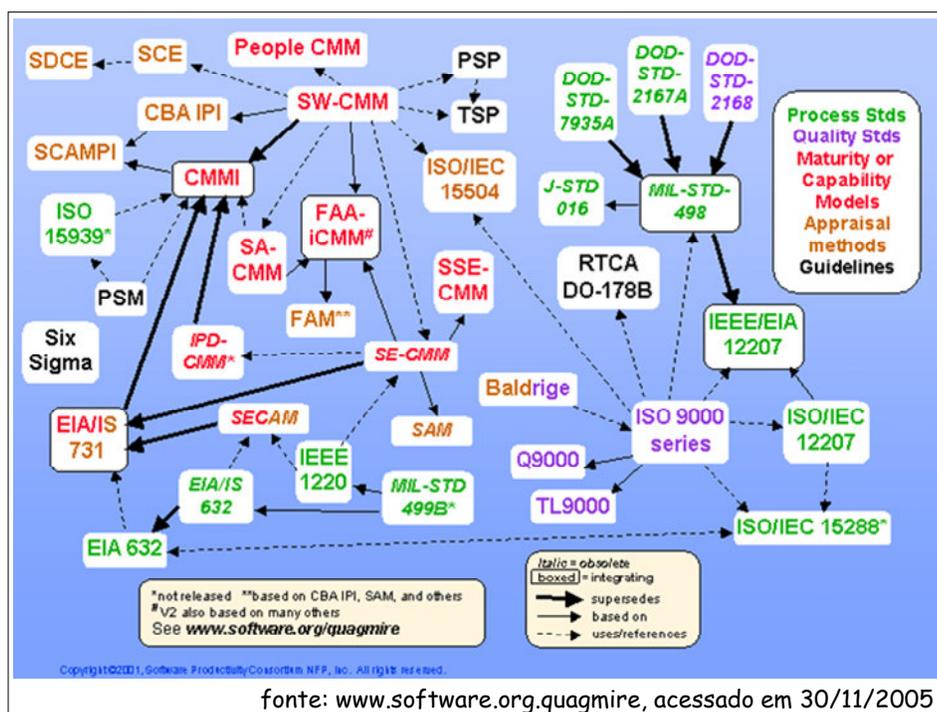


Figura 37 – Framework Quagmire de modelos ©2001 SPC

Halvorsen e Conradi [1999] apresentam e comentam quatro classes de métodos de comparações de modelos e frameworks de melhoria e avaliação de processo de software: comparação por característica, comparação por mapeamento, comparação bilateral e comparação por necessidades.

Nas comparações por características, são definidas características relacionadas às áreas de interesse dos modelos e frameworks, e os resultados descrevem como cada um possui ou não cada característica. Comparação por mapeamento é o processo de criar mapas dos conceitos e elementos de um framework ou modelo para os outros, identificando sobreposições, correlações fortes e fracas, e não correlações entre eles. Uma comparação bilateral é feita com a comparação detalhada entre dois modelos ou frameworks, geralmente do ponto de vista de um deles e descrição do outro em termos desse ponto de vista. Finalmente uma comparação por necessidades é feita em termos de necessidades específicas identificadas por uma organização que devem ser consideradas para a escolha de um framework ou modelo.

Utilizando as quatro categorias apresentadas por Halvorsen e Conradi [1999] podemos classificar as principais comparações de modelos e frameworks de melhoria e avaliação de processo de

software. Em alguns casos, mais de uma classe de comparação pode ser atribuída. Nesse caso é atribuída aquela que se entendeu como a mais relevante.

Uma comparação por características de seis frameworks e modelos (TQM, CMM v1.1, ISO 9000:1994, ISO/IEC TR 15504, EF/QIP/GQM e SPIQ) é descrita por Halvorsen e Conradi [1999]. Para esta comparação são utilizadas vinte e cinco características, agrupadas em cinco categorias: geral, processo, organização, qualidade e resultado.

Uma comparação bilateral entre Software Patterns e ISO/IEC 15504 Framework é descrita em [Salviano 2000].

Uma comparação estrutural entre a ISO/IEC TR 15504 e a representação contínua de modelos do CMMI é realizada por Lepasaar et al. [2000]. Esta comparação foi utilizada para a definição de um metamodelo de processo construído como uma integração entre a ISO/IEC TR 15504 e a representação contínua do CMMI [Lepasaar e Mäkinen 2001].

Uma comparação por mapeamento para analisar a compatibilidade entre as representações da ISO/IEC TR 15504-2 e o CMMI está descrita em [Rout et al. 2001]. Os objetivos principais desta análise foram apoiar a avaliação da apropriabilidade do CMMI para uso pelo Departamento de Defesa da Austrália para gestão da aquisição de sistemas intensivos em software, e verificar a possibilidade de adaptação de métodos de avaliação compatíveis com a ISO/IEC 15504 para utilização em relação ao modelo CMMI.

Um esforço de integração de frameworks e modelos tem sido realizado pela Administração Federal de Aviação dos Estados Unidos (Federal Aviation Administration FAA) desde 1997, como base do modelo FAA iCMM [Ibrahim et al. 2001]. A versão iCMM 2.0, publicada em 2001, integra vários frameworks e modelos, incluindo SW-CMM, SE-CMM, SA-CMM, ISO/IEC TR 15504, ISO 9001:2000, ISO/IEC CD 15288, ISO/IEC 12207, EIA/IS 731, *Malcom Baldrige National Award*, e CMMI.

Habra e Alexandre [2003] modelaram estruturas de cinco modelos de maturidade de processos em UML. As estruturas modeladas foram do SW-CMM 1.1, Trillium 3.0, ISO/IEC TR 15504-2, ISO/IEC TR 15504-5, Bootstrap 3.0, CMMI-SE 1.1 representação contínua, e CMMI-SE 1.1 representação estagiada.

Comparações entre ISO/IEC 15504 e ISO 9001:2000 têm sido realizadas pelo projeto SPICE9000. Um modelo para avaliação de processo SPICE9000 foi produzido como uma integração dos requisitos da ISO 9001 representados como um modelo compatível com a ISO/IEC 15504 [Dorling et al. 2003].

Mutafelija e Stromberg [2003] comparam ISO 9001:2000 com o CMMI. Hyder et al. [2004] comparam o SCM-SP v2 com o CobiT v3 ([www.isaca.org](http://www.isaca.org)), ISO 9001:2000 ([www.iso.ch](http://www.iso.ch)), BS 15000:2002 ([www.itsmf.com](http://www.itsmf.com)), CMMI v1.1 ([www.sei.cmu.edu/cmmi](http://www.sei.cmu.edu/cmmi)) e COPC r3.3 ([www.copc.com](http://www.copc.com)). Weber et al. [2005a] descrevem o MPS.BR como uma composição entre o CMMI, ISO/IEC 15504-5 e ISO/IEC 12207. Chrissis et al. [2003] descrevem o CMMI e o relaciona ao SW-CMM, SE-CMM e IPPD-CMM. Sherer e Thrasher [2005] e Sheakley [2002] comparam o CMMI com o PMBOK. Paulk et al. [1995] descrevem o SW-CMM e o relaciona à ISO 9001.

Conforme já descrito, vários modelos de capacidade de processo foram construídos e com isto foi identificada a área de engenharia de modelos de capacidade. Porém, existe pouca literatura sobre o assunto. Uma publicação relaciona oitos passos para construção de modelos de maturidade, que foram utilizados na construção do modelo SM<sup>MM</sup> [April et al. 2004a, April et al. 2004b]:

- “Entender a área de conhecimento;
- Analisar Normas para buscar processos, atividades e melhores práticas;
- Analisar Frameworks e o SWEBOK para criar domínios e áreas chave de processo;
- Decidir as práticas para serem incluídas no modelo e o nível de maturidade de cada;
- Construir ou detalhar a arquitetura do modelo;
- Encontrar uma organização e conduzir estudos de caso;
- Revisar o conteúdo com pessoas experientes da área; e
- Melhorar o modelo quando necessário.”

## 2.4.2 Comparações e combinações das arquiteturas estagiada e contínua

Um dos debates mais interessantes na área de melhoria de processo é aquele referente às duas arquiteturas que dominam os modelos de capacidade de processo: a arquitetura estagiada (ou por estágios, em inglês *staged*) e a arquitetura contínua (em inglês *continuous*).

A arquitetura estagiada foi definida originalmente por Watts Humphrey [1987] com a definição dos cinco níveis de maturidade que formaram o modelo SW-CMM [Paulk et al. 1994]. Esta definição foi consolidada nos modelos do CMMI, agora com o nome de “representação estagiada” [Chrissis et al. 2003]. Uma comparação entre as arquiteturas estagiada e contínua é apresentada nos modelos do CMMI e repetida por Chrissis et al. [2003, p.11-15, 92-95], como representação contínua e estagiada, conforme descrita na Tabela 12.

Tabela 12 – Representações contínua e estagiada do CMMI

<b>Representação Contínua do CMMI</b>	<b>Representação Estagiada do CMMI</b>
Possibilita liberdade explícita para selecionar a ordem da melhoria que melhor atenda os objetivos estratégicos da organização e trate as áreas de risco da organização.	Permite que as organizações tenham um caminho de melhoria predeterminado e provado.
Permite uma maior visibilidade da capacidade atingida em cada área de processo.	Foca em um conjunto de processos que prove uma organização com uma capacidade específica que é caracterizada por cada nível de maturidade.
Provê uma pontuação de nível de capacidade que é utilizada primeiramente para melhoria da organização e é raramente comunicada externamente.	Provê um nível de maturidade que é geralmente utilizado em comunicações gerenciais internas, comunicações externas à organização, e durante aquisições com um meio para a qualidade.
Permite melhoria de processos diferentes serem realizadas em diferentes taxas	Resume resultados de melhoria de processo em uma forma simples, um único número de nível de maturidade
Reflete uma nova abordagem que ainda não tem dados para demonstrar que está relacionado a retorno de investimento	Baseado em uma história relativamente longa de uso que inclui estudos de caso e dados que demonstram retorno de investimento provado
Provê uma migração fácil do modelo SECM para o CMMI	Provê uma migração fácil do modelo SW-CMM para o CMMI
Permite uma comparação fácil da melhoria de processo com a ISO/IEC 15504 por que a organização das áreas de processo é derivada da 15504	Permite comparação com a 15504, mas a organização das áreas de processo não corresponde à organização utilizada na ISO/IEC 15504

Segundo Chrissis et al. [2003, p.11-15, 92-95], três categorias de fatores podem influenciar a decisão sobre a escolha da representação contínua ou estagiada: fatores de negócio, culturais e legados.

Estas posições já existiam desde a época do modelo SW-CMM e estão descritas por exemplo por Curtis [1998]. Bill Curtis comenta que “a melhoria dos processos não é suficiente pois é necessário também a mudança da organização” [Curtis 1998, p.10]. Ao comentar o debate entre modelos estagiados e contínuos, ele defende que “os modelos contínuos são para aqueles que desejam liberdade para escolher o foco da melhoria [atuando na melhoria de cada processo em separado] e que os modelos estagiados são para aqueles que desejam a mudança da cultura da organização [Curtis 1998, p.13].

Tim Kasse publicou uma comparação entre as representações estagiada e contínua do CMMI em termos de quatro dimensões: o foco, os mitos e entendimentos errados, influências negativas para melhoria de processo, e influências positivas também para a melhoria de processo [Tabela 13] [Kasse 2004, p. 266-269].

Tabela 13 – Comparações entre as representações estagiada e contínua [Kasse 2004]

<b>Foco das representações</b>	
<b>Representação estagiada do CMMI</b>	<b>Representação contínua do CMMI</b>
“Maturidade organizacional:	“Capacidade de área de processo organizacional:
Representado pelo grau de melhoria de processo em relação a um conjunto pré-definido de áreas de processo;	O quanto um processo é explicitamente documentado, gerenciado, medido, controlado e melhorado continuamente;
Escolhido para atender as necessidades de melhoria de processo de uma organização;	Representa melhoria na implementação e efetividade de uma área de processo individual;
Processos ordenados e então agrupados baseado em relacionamentos predefinidos de maturidade organizacional que tratam as necessidades de negócio de várias organizações;	Apoia a melhoria contínua de áreas de processo individuais que são críticas às necessidades de negócio da organização;
Provê um indicador da maturidade dos processos de uma organização para responder à questão: Qual será o desempenho e resultado mais prováveis do próximo projeto a ser realizado?”	Provê um indicador de melhoria em relação a uma área de processo para responder à questão: Qual é uma boa estratégia para implementar uma melhoria desta área de processo? “
<b>Mitos e entendimentos errados das representações</b>	
<b>Representação estagiada do CMMI</b>	<b>Representação contínua do CMMI</b>
“Gerência de requisitos tem que ser completamente implementadas antes da implementação do desenvolvimento de requisitos;.	“Uma organização pode focar sua iniciativa de melhoria em qualquer grupo de áreas de processo que ela queira sem se preocupar com possíveis dependências entre elas;
Garantia da qualidade e gerência de configuração se aplicam apenas para projetos no nível de maturidade 2 do CMMI;	Nível de capacidade 1 para um determinada área de processo é fácil de ser obtida (por exemplo, gerência quantitativa de projeto);
Um programa de medição da organização deve estar completamente implementado quando ela atingir o nível 2 de maturidade do CMMI;	Uma organização pode de forma prática evoluir qualquer área de processo para o nível de capacidade 5;
Revisões por pares não deve ser implementado até que a organização esteja pronta para as atividades do nível 3 de maturidade do CMMI;	O foco na representação contínua e nos níveis de capacidade irão remover o foco no número que é associado à representação estagiada;
Gerência de riscos não deve estar implementada até a organização estar trabalhando no nível 3 de maturidade do CMMI;	Existe um caminho de evolução claro para todas as áreas de processo que podem ser seguidos de forma simples para ajudar uma organização evoluir para o nível de capacidade 5 para todas ou a maioria das áreas de processo;
Organizações não terão mais nenhum problema com fornecedores depois de atingir o nível 2 de maturidade do CMMI	A utilização da representação contínua não requer a priorização da implementação de suas áreas de processo
Atividades de engenharia não são necessárias para atingir o nível 2 de maturidade do CMMI.	As áreas de processo da engenharia não são úteis para a aplicação na engenharia de software porque elas são baseadas

	primeiramente nas necessidades da engenharia de sistemas.”
Treinamento organizacional não é necessário até que a organização esteja pronta para começar os processos do nível 3 de maturidade do CMMI.”	
<b>Influências negativas das representações para melhoria de processo</b>	
<b>Representação estagiada do CMMI</b>	<b>Representação contínua do CMMI</b>
“Dá a idéia às organizações que elas podem implementar apenas iniciativas de melhorias que focam nas áreas de processo que estão incluídas em um determinado nível de maturidade;	“Parece indicar que uma organização tem uma flexibilidade extrema em escolher as áreas de processo para implementação em qualquer ordem desejada sem considerar as dependências entre elas;
Não provê orientações para como implementar de forma incremental áreas de processo como, por exemplo, Solução Técnica;	Permite uma ênfase na implementação de atividades da engenharia e diminui a importância das atividades de gerência.”
Não provê nenhuma orientação para como deve ser a implementação de qualquer área de processo do nível 3 de maturidade do CMMI a partir da perspectiva de um nível de maturidade menor;	
Foca a organização mais no atendimento do nível de maturidade do que em melhorias mensuráveis que apoiam os objetivos estratégicos da organização;	
Pode influenciar a organização a priorizar práticas gerenciais e negligenciar práticas técnicas.“	
<b>Influências positivas das representações para melhoria de processo</b>	
<b>Representação estagiada do CMMI</b>	<b>Representação contínua do CMMI</b>
“Ajuda a organização a priorizar seus esforços de melhoria, especialmente quando: a iniciativa de melhoria de processo da organização está apenas começando; a organização está em um nível baixo de maturidade e a organização tem pouca experiência em estabelecimento de processo;	“Com o uso das áreas de processo da engenharia e das práticas genéricas, a representação contínua pode provê um caminho de melhoria da capacidade individual para cada área de processo.
Foca a organização em colocar no lugar as funções de gerência de projeto que são necessárias para apoiar todas as atividades de engenharia e de gerência durante toda sua jornada de melhoria de processo.”	Se as necessidades de negócio demandarem, o caminho para evoluir uma área de processo particular ou uma área de categoria para níveis altos de capacidade pode ser facilmente identificado e medido
	Auxilia a organização a construir um perfil alvo de áreas de processo que ajudará a organização

	a resolver objetivos estratégicos conhecidos.”
--	--

A arquitetura contínua foi definida originalmente pelo grupo que desenvolveu a Norma ISO/IEC 15504 com a definição dos seis níveis de capacidade e de processos que são independentes do nível de capacidade [ISO/IEC TR 15504-5 1998, ISO/IEC 15504-5 2006]. Esta definição também foi utilizada nos modelos do CMMI, agora com o nome de “representação contínua” [Chrissis et al. 2003]. Uma comparação entre as arquiteturas estagiada, conforme utilizada no SW-CMM, e contínua, conforme utilizada na ISO/IEC PDTR 15504-5 [1995] é descrita no curso da 15504 [ESI 1996, p. 125-127], conforme descrito na Tabela 14.

Tabela 14 - Arquitetura contínua da 15504 e estagiada do SW-CMM

<b>Arquitetura Contínua da 15504</b>	<b>Arquitetura Estagiada do SW-CMM</b>
Estrutura com duas dimensões: níveis de capacidade e processos.	Estrutura com uma dimensão: níveis de maturidade
Permite estratégias flexíveis para a melhoria.	Define um caminho de melhoria
Níveis de capacidade para cada processo	Um único nível de maturidade para toda a organização.
Resultados precisam ser simplificados para serem entendidos	Resultados são fáceis de serem entendidos
Resultados são mais completos	Resultados muito simplificados

Shaeffer, diretor de engenharia de sistemas do DoD e patrocinador do CMMI, relata problemas encontrados em organizações nos níveis 4 e 5 de maturidade [Shaeffer 2004]. É ressaltado que estas organizações tiveram melhorias, que são relatadas, mas também tem problemas, que não são divulgados. O problema, segundo Shaeffer, está relacionado aos efeitos negativos dos níveis de maturidade, incluindo a ‘supervalorização’ pelas organizações do foco nos níveis de maturidade em detrimento da melhoria contínua, e o fato das organizações serem tentadas a ver um determinado nível de maturidade do CMMI como um fim e não como um meio para um fim. A proposta de Shaeffer, em nome do DoD, é acabar com os níveis de maturidade, e manter apenas os níveis de capacidade.

A solução para o problema da “mania por níveis de maturidade”, segundo Shaeffer, envolve “o desejo do DoD de: mudar o foco de níveis de maturidade para perfis de capacidade; remover a tentação dos níveis de maturidade e da síndrome do “*one size fits all*”(um único tamanho serve para todos); desencorajar o uso dos níveis de maturidade como critério de seleção e substituí-los por perfis alvo do CMMI baseados em riscos e avaliação de capacidade e desenvolver medições significativas de capacidade de processo baseadas não em níveis de maturidade, como, por exemplo, nível 3, mas em desempenho de processo” [Shaeffer 2004, slide 13]. Isto, ainda segundo Shaeffer, tem como objetivo a melhoria do impacto do CMMI no desempenho de programas de melhoria de processo. Com isto, cada

organização mostraria o seu perfil de capacidade de processo, com o nível de capacidade de cada área de processo.

Rout critica os níveis de maturidade do CMMI por permitir a aplicação de práticas sofisticadas de gerência a potenciais práticas defeituosas de engenharia [Rout 2004]. Em uma experiência duas equipes desenvolveram uma aplicação para leitura de crachás de funcionários utilizando cada uma diferentes abordagens. A equipe “CMM4” utilizou seu processo avaliado como nível 4 de maturidade do CMM, enquanto a equipe “Métodos Formais” utilizou um método formal. Ambas as equipes falharam em atingir as metas do projeto. A equipe “Métodos Formais” falhou principalmente pela falta de processo, enquanto que a equipe “CMM4” falhou pelo julgamento inadequado dos requisitos do cliente que não foram analisados pelos projetistas e implementadores [Glass 2003, Smidts et al. 2002, ambos apud Rout 2004]. A interpretação desse resultado, segundo Rout, é que a equipe “CMM4” é uma instância típica de institucionalização de práticas pobres de engenharia, e por isto o risco associado com a abordagem inadequada para engenharia de requisitos não foi reconhecida. Isto foi induzido pelos níveis de maturidade correntes do CMM e do CMMI.

Como uma solução é proposta a criação de mais um nível de maturidade, anterior ao nível 2 de maturidade, com foco na engenharia no nível 1 de capacidade, e pequenos ajustes no nível 2 de maturidade. A Figura 38 ilustra os níveis atuais e a nova proposta, ambas expressas com o mapeamento para a representação contínua do CMMI.

No modelo proposto, é definido o nível 1 de maturidade que representa a situação onde uma organização domina os princípios básicos daquelas disciplinas que são o núcleo de seu negócio, no caso a engenharia de software para o desenvolvimento de software.

Outra variação dos níveis de maturidade é feita no modelo de referência do MPS-BR [Weber et al. 2005a]. O modelo MR-MPS define sete níveis de maturidade, dos quais quatro (níveis A, B, C e F) são equivalentes aos quatro níveis de maturidade do CMMI (respectivamente os níveis 5, 4, 3 e 2) e três são níveis intermediários entre os níveis 2 e 3 do CMMI (os níveis D e E) e entre os níveis 1 e 2 (o nível G) [Figura 39]. A racionalidade para a inclusão de níveis de maturidade no MPS-BR é reduzir o esforço de um nível para outro.

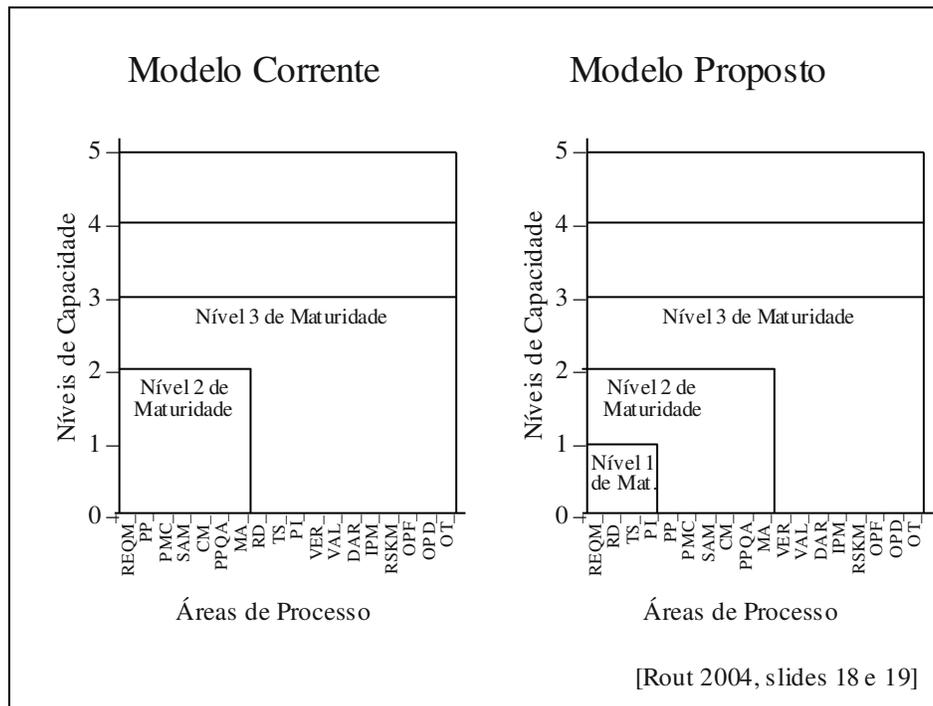


Figura 38 – Proposta alternativa para níveis de maturidade do CMMI-SE/SW

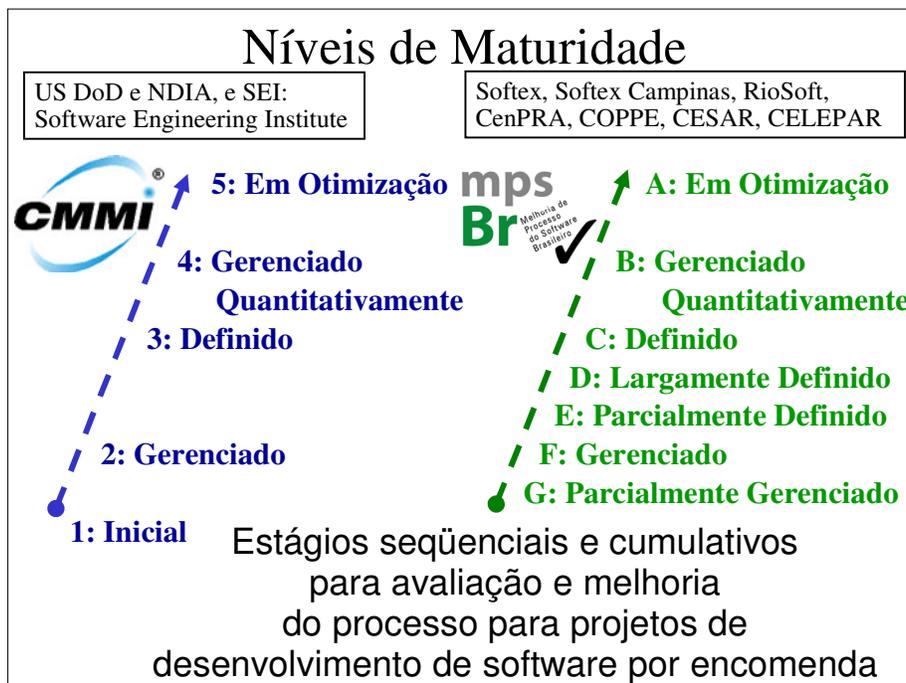


Figura 39 – Relacionamento entre os níveis de maturidade do CMMI e MPS-BR

Myers procura mostrar os relacionamentos entre as representações contínua e estagiada do CMMI e realçar que mesmo quando é utilizada a arquitetura estagiada, tipicamente, outras áreas de processo são também melhoradas [Myers 2003] [Figura 40].

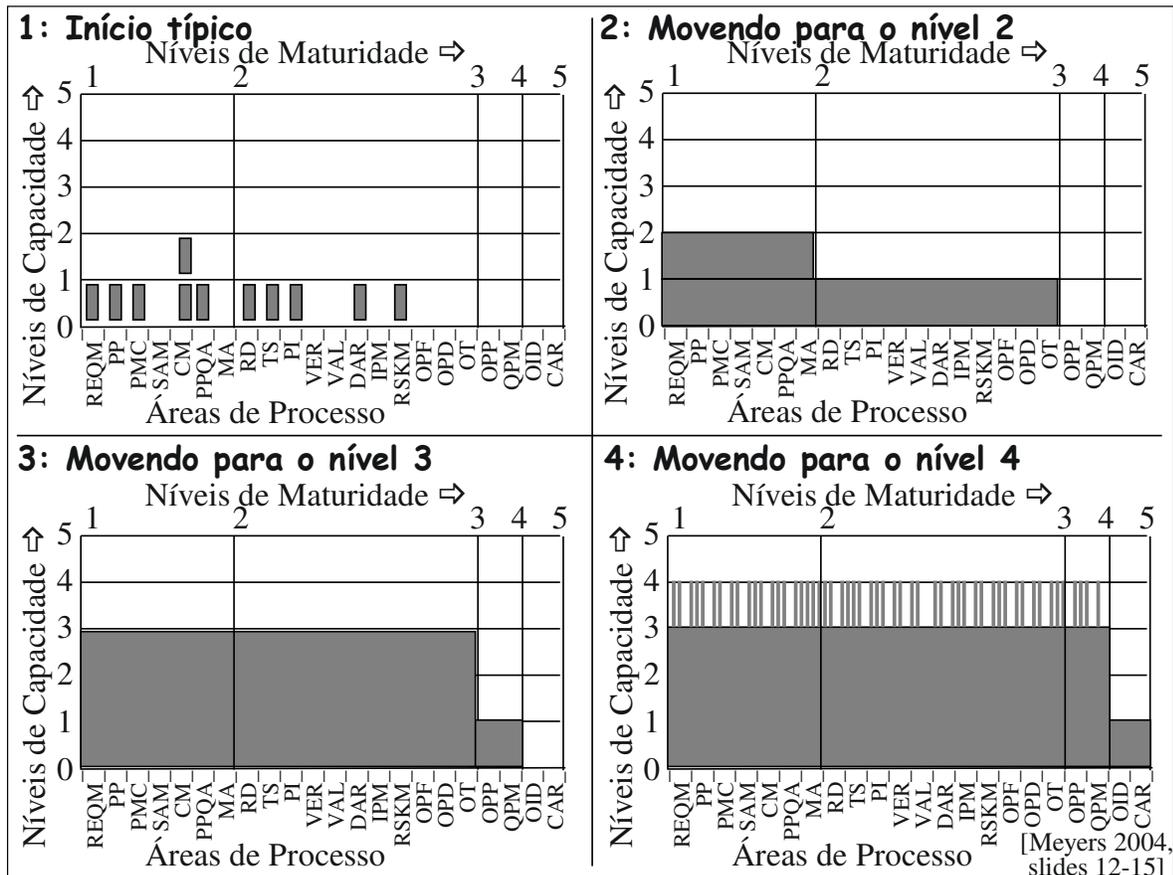


Figura 40 - Perfis de processo nos nível 1 a 4 de maturidade do CMMI-SE/SW

Segundo Myers, ao iniciar um programa de melhoria orientado pela representação estagiada do modelo CMMI-SE/SW, uma organização típica tem o perfil ilustrado no primeiro quadrante, onde as áreas de processo REQM, PP, PMC, RD, TS, PI, RSKM e DAR estão no nível 1 de capacidade e a área de processo CM no nível 2 de capacidade. O segundo quadrante ilustra o perfil da organização para mover para o nível 2 de maturidade. Nesse caso, além das áreas REQM, PP, PMC, SAM, CM, PPQA e MA estarem todas no nível 2 de capacidade, conforme necessário para o nível 2 de maturidade, todas as outras áreas RD, TS, PI, VER, VAL, DAR, IPM, RSKM, OPF, OPD e OT estão no nível 1 de capacidade. O terceiro quadrante ilustra o perfil da organização para mover do nível 2 para o nível 3 de maturidade. Nesse caso, além das áreas REQM, PP, PMC, SAM, CM, PPQA, MA, RD, TS, PI, VER, VAL, DAR, IPM, RSKM, OPF, OPD e OT estarem todas no nível 3 de capacidade, conforme necessário para o nível 3 de maturidade, todas as duas outras OPP e OPM estão no nível 1

de capacidade. O quarto quadrante ilustra o perfil da organização para mover do nível 3 para o nível 4 de maturidade. Nesse caso, além das áreas REQM, PP, PMC, SAM, CM, PPQA, MA RD, TS, PI, VER, VAL, DAR, IPM, RSKM, OPF, OPD, OT, OPP e OPM estarem todas no nível de capacidade 3, conforme necessário para o nível 3 de maturidade, todas as duas outras áreas OID e CAR estão no nível 1 de capacidade. As áreas OPP e OPM no nível 3 de capacidade, causam que elementos das áreas que estavam no níveis 3 de capacidade estejam no nível 4 de capacidade.

A Figura 41 ilustra o perfil para mover do nível 4 para o nível 5 de maturidade.

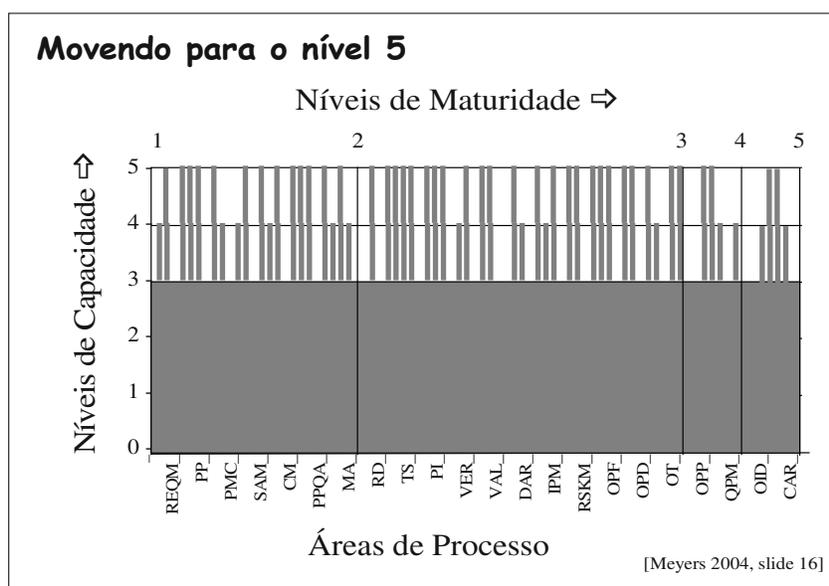


Figura 41 – Perfil de processo no nível 5 de maturidade do CMMI-SE/SW

Nesse caso, as áreas REQM, PP, PMC, SAM, CM, PPQA, MA RD, TS, PI, VER, VAL, DAR, IPM, RSKM, OPF, OPD, OT, OPP, OPM, OID e CAR estão todas no nível de capacidade 3, conforme necessário para o nível 5 de maturidade. As áreas OPP e OPM no nível 3 de capacidade, causam que elementos das áreas que estavam no níveis 3 de capacidade estejam no nível 4 de capacidade, e as áreas OID e CAR no nível 3 de capacidade, causam que alguns dos elementos das áreas que estavam no níveis 4 de capacidade estejam no nível 5 de capacidade.

Myers também sugere uma abordagem para utilização da representação contínua baseada em um foco nas áreas de processo de engenharia [Myers 2003]. A Figura 42 descreve a proposta de Myers e também reinterpreta a proposta de modelo de Rout. A única diferença entre as duas propostas é que Myers inclui as áreas de verificação e validação na primeira utilização, enquanto que Rout as inclui apenas na segunda.

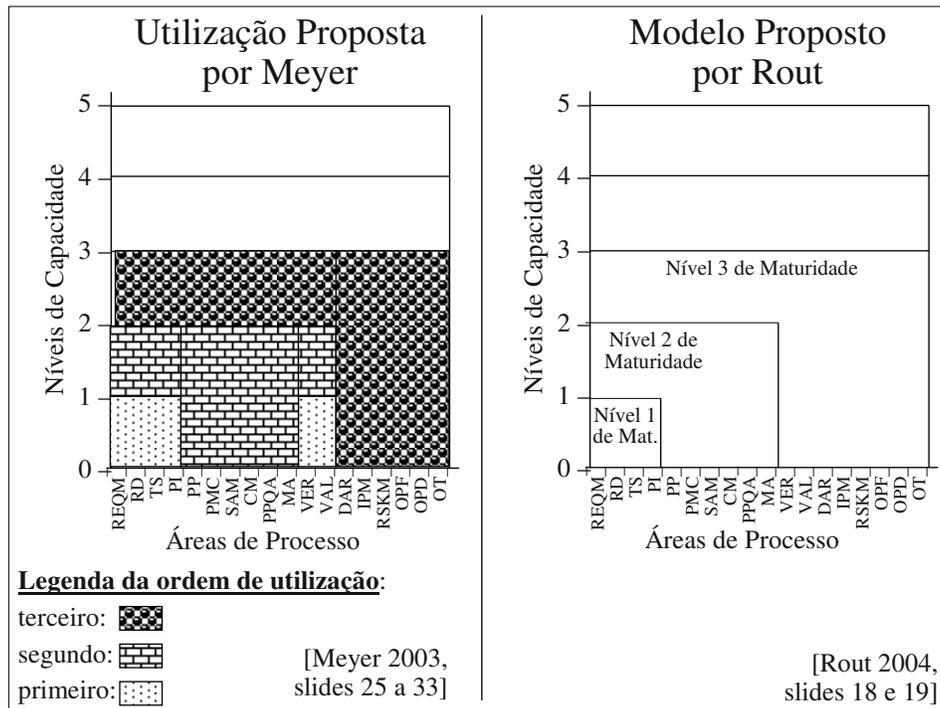


Figura 42 – Utilização proposta por Myers e modelo proposto por Rout

Kasse propõe uma abordagem para utilização das arquiteturas estagiada e contínua, ao mesmo tempo. Esta abordagem é chamada de abordagem “*Constagedeous*”<sup>8</sup> [Kasse 2004, p. 261-270]. Segundo Kasse, várias organizações escolhem a arquitetura contínua, algumas de forma consciente e outras sem mesmo perceber, de uma forma estagiada. Estas organizações tratam cerca de duas áreas de processo em um determinado momento antes de tratar outras áreas de processo para a melhoria. Para acompanhar o progresso, algumas áreas de processo de maior interesse podem ser avaliadas (uma avaliação segundo a arquitetura contínua) ou após uma melhoria em um conjunto de áreas de processo, as áreas de um determinado nível de maturidade podem ser avaliadas (uma avaliação segundo a arquitetura estagiada) [Kasse 2004, p. 269].

A seguir é reproduzido um parágrafo onde Kasse comenta a abordagem *constagedeous*. Nesse parágrafo ele utiliza os termos representação contínua e estagiada, e abordagem contínua.

“Organizações com menos experiência talvez queiram evitar o excesso de requisitos da representação estagiada para avaliar o progresso de todas as áreas pertencentes a um nível de maturidade e devem escolher uma abordagem evolucionária para realizar e validar melhorias, cerca de

<sup>8</sup> Constagedeous é uma palavra inventada por Kasse para significar a junção de “continuous” (contínuo) e “staged” (estagiado).

---

duas áreas de processo a cada vez, utilizando a abordagem *constagedeous*. É tudo uma questão de como olhar estas possibilidades. Melhoria de processo é uma atividade evolucionária, com pequenos passos a cada vez de forma incremental. Nenhuma organização consegue tratar todas as questões e problemas de uma vez. A organização conduz verificações intermediárias do progresso, independentemente da representação [estagiada ou contínua] que utiliza, mas pode utilizar a abordagem contínua para avaliar a capacidade de suas áreas de processo, ou pode realizar várias ações de melhoria e esperar para avaliar um nível de maturidade de várias áreas de processo em um determinado nível [de capacidade], conforme exigido em cada nível [de maturidade]. Se quiser focar as áreas de gerência de projeto, a organização ainda precisa elicitar requisitos; projetar e construir componentes, revisá-lo com pares, e realizar testes de unidade neles; e realizar integração, conduzir testes de sistema, e eventualmente produzir um produto que será entregue. Em outras palavras, a organização precisa realizar as atividades básicas de engenharia [de produto]. Buscar o nível 2 de maturidade do CMMI sem fazer a engenharia não tem nenhum sentido” [Kasse 2004, p. 269].

Na descrição das influências negativas do uso do modelo estagiado (do CMMI), faltou indicar que os níveis de maturidade são um exemplo, que funciona bem para um determinado contexto, e não é o melhor para qualquer organização em qualquer situação. Na descrição das influências negativas do uso do modelo contínuo (do CMMI), faltou indicar que falta a orientação da necessidade de apoio metodológico para selecionar perfis de capacidade de processo úteis e efetivos.

Os níveis de maturidade 4 e 5, naturalmente da arquitetura estagiada, podem ser considerados como uma representação contínua simulada. As duas áreas de processo do nível 4 orientam a escolha de algumas partes dos processos correspondentes às áreas de processo dos níveis 2 e 3, e para estas partes, aplicar os conceitos do nível 4. Ou seja, para o nível 4 de maturidade devem ser escolhidas algumas áreas de processo, e nestas áreas, algumas partes, para aplicar o nível 4 de maturidade. Para o nível 5, ocorre algo similar.

Na alternativa “parcialmente estagiado” [Ahern et al. 2001, p. 194-198], algumas áreas de processo, que sempre deverão ser consideradas, são selecionadas em estágios (por exemplo, planejamento de projeto, no nível 2). As outras áreas de processo são organizadas de forma semelhante ao contínuo (por exemplo acordo com fornecedor). Denominada informalmente de U\_CMMI (*Unified CMMI*, CMMI unificado) [Ahern et al. 2001, p. 199-200], propõe o modelo com apenas as práticas genéricas e as áreas de processos que impulsionam e apoiam estas práticas. Desta forma não teria a engenharia por exemplo. A terceira proposta é a “remoção das metas” [Ahern et al. 2001, p. 200-202].

Olson resume as vantagens e desvantagens de cada representação do CMMI conforme a Tabela 15 [Olson 2003]. Olson conclui identificando que organizações maduras devem utilizar várias estratégias de melhoria e ambas as representações (estagiada e contínua).

Tabela 15 – Vantagens e desvantagens segundo Olson [2003]

<b>Representação do CMMI</b>	<b>Vantagens</b>	<b>Desvantagens</b>
Estagiada:	Construída em uma estratégia	Investimento alto no início
	Áreas de processo construídas umas nas outras	Resultados mensurados podem demorar
	Grandes benefícios com longa duração	Pode ser mais difícil de ser vendido à alta gerência
	A maioria dos problemas da qualidade são planejados desta forma	Pode ser mais difícil de ser implementado
	Negócio do DoD	Avaliações mais caras
Contínua:	As áreas de processo selecionadas podem atender diretamente os objetivos estratégicos	Problemas sistêmicos da qualidade podem não ser tratados
	Pode atingir resultados mais rápidos	Talvez não tenha benefícios que durem por um longo tempo
	Pequeno investimento no início	Talvez implemente processos na ordem errada
	Mais fácil de ser vendido	Possível pensamento de curto prazo

Tim Olson argumenta que qual arquitetura (ou representação) deve ser utilizada, ou seja, estagiada ou contínua, não é a pergunta a ser feita. Primeiro devem ser identificadas as metas de qualidade, objetivos e estratégia da organização. Com isto deve ser escolhida uma estratégia de melhoria, que pode ser melhoria da qualidade, controle da qualidade ou planejamento da qualidade [Olson 2003]. Estas estratégias são baseadas na visão Juran em “*Juran on Leadership for Quality: An Executive Handbook*” [1989, apud Olson 2003, slide 21].

Por exemplo, uma organização com orçamento limitado, que precise economizar recursos e ter um retorno de investimento rápido deve escolher uma estratégia de melhoria da qualidade. Isto sugere a seleção da representação contínua com provavelmente a escolha de uma área de processo para orientar a melhoria. Por outro lado, uma organização que deseje ser a melhor de sua classe em um prazo longo, e deseje chegar nesta posição de forma ordenada, pode selecionar a estratégia de planejamento da qualidade. Isto sugere a seleção da representação estagiada [Olson 2003, slide 23].

### 2.4.3 Escolha de processos de modelos contínuos

A ISO/IEC 15504-4 [2004] descreve uma abordagem simples para o estabelecimento de capacidade alvo, baseado no conjunto de processos definidos na ISO/IEC 15504-5 [2006] como um Modelo para Avaliação de Processo (PAM). Esta abordagem é descrita a seguir com seus quatro passos e comentado em relação a um exemplo de um resultado que está ilustrado na Figura 43. Cada passo é descrito por um nome, ações e racionalidade.

No Passo 1 é selecionado um conjunto inicial de processos. A ação é selecionar os processos da categoria fundamental, excluindo qualquer processo não relevante para os requisitos especificados. A racionalidade é que os processos da categoria fundamental da ISO/IEC 15504-5 contribuem mais diretamente para os serviços e produtos entregues.

No Passo 2 são definidas as pontuações dos atributos de processo requeridos para cada processo da definição inicial. A ação é atribuir a pontuação F (“fully”) para todos os atributos de processo dos níveis 1, 2 e 3 de capacidade. A racionalidade é que esta abordagem garante que os processos selecionados sejam executados, gerenciados e padronizados.

Processos, conforme definidos na ISO/IEC 15504-5:	Níveis de Capacidade e Atributos de Processo, conforme definidos na ISO/IEC 15504-2:									
	Executado		Gerenciado		Estabelecido		Previsível		Em Otimização	
	PA 1.1	PA 2.1	PA 2.2	PA 3.1	PA 3.2	PA 4.1	PA 4.2	PA 5.1	PA 5.2	
ENG.1 Elicitação de Requisitos	F	L	L							
ENG.3 Projeto da Arquitetura de Sistema	F	F	F	F	F	L	L			
SUP.8 Gerência de Configuração	F	F	F	L	L					
MAN.5 Gerência de Riscos	F	F	F	F	F					
ACQ.2 Seleção de Fornecedor	F	F		L	L					

Pontuação, como definido na ISO/IEC 15504-2:			
	Não requerido		Completamente atingido
	Parcialmente atingido		Largamente atingido
	Não atingido		

figura adaptada de [ISO/IEC 15504-4 2004, p. 10, Figura 2]

Figura 43 – Exemplo de escolha de processos e níveis de capacidade com a 15504-4

No Passo 3 são revisadas e ajustadas as pontuações dos atributos de processo requeridos para cada processo da definição inicial. A ação é acrescentar pontuações para os níveis 4 ou 5, reduza

pontuações para os atributos do nível 3. A racionalidade é que acrescentando os atributos de processo dos níveis 4 e 5 para alguns processos pode ser justificado para reduzir os riscos relacionados a processo. A Figura 43 ilustra esse caso onde o perfil de processo alvo para o processo ENG.3 de projeto da arquitetura de sistema inclui os atributos de processo do nível 4 de capacidade. Algumas vezes, a remoção de atributos do nível 3 pode ser feita. A Figura 43 também ilustra esse caso onde o perfil de processo alvo para o processo ENG.1 de elicitação de requisitos consideram apenas os atributos dos níveis 1 e 2.

No Passo 4 são acrescentados outros processos, com as pontuações de atributo de processo requeridos para cada um desses processos acrescentados. A ação é acrescentar os processos de apoio e organizacionais do ciclo de vida. A racionalidade é que os processos de apoio e organizacionais do ciclo de vida são críticos para estabelecer níveis altos de capacidade dos processos da organização. Vários atributos de processo estão relacionados aos processos de apoio e organizacionais do ciclo de vida. Por exemplo, se o atributo de gerência de execução (PA2.1) for incluído para os processos primários, então o processo de gerência de projeto deve ser incluído. A capacidade alvo dos processos de apoio e organizacionais é dirigida pelo quanto eles apoiam os atributos de processo dos processos selecionados inicialmente. Outros processos organizacionais e de apoio também devem ser incluídos quando eles forem considerados relevantes para a determinação da capacidade ou para a melhoria contínua.

O iCMM também define uma abordagem para escolha de processos [já descrita na seção 2.3.2]. Em geral, o grupo de áreas de processo em um nível de maturidade é considerado como uma fundação essencial para o próximo incremento do progresso refletido no nível de maturidade seguinte.

O método FAME (*Fraunhofer IESE Assessment Method*, método de avaliação do Fraunhofer IESE) [Beitz et al. 1999] define uma abordagem para seleção dos processos mais relevantes para o negócio da organização, que serão avaliados e considerados para melhoria. Os processos são selecionados entre aqueles da ISO/IEC TR 15504. Para tanto são utilizados três visões:

- Modelagem das dependências entre produto e processo: baseados nas metas de qualidade do produto, os processos relacionados são identificados utilizando um repositórios destas dependências;
- Estudo dos processos influentes: Os processos relacionados são selecionados baseados nas metas de desempenho desejadas que são derivadas dos resultados do projeto SPICE Trials;

- Experiência baseada em heurísticas: Heurísticas simples são utilizadas para selecionar os processos mais relevantes baseado na coleção de experiências dos avaliadores entre os processos a serem melhorados e resultados de desempenho.

O método RAPID (*Rapid Assessment for Process Improvement for Software Development*, Avaliação Rápida para Melhoria de Processo para Desenvolvimento de Software) [Rout et al. 2000c] define um perfil de capacidade de processo fixo que é utilizado como referência para uma avaliação rápida. Os processos selecionados da ISO/IEC TR 15504-5 são os seguintes: CUS.3: Requisitos de Software, MAN.2: Gerência de Projeto, SUP.3: Garantia da Qualidade, MAN.4: Gerência de Riscos, ENG.1: Desenvolvimento de Software, SUP.2: Gerência de Configuração, SUP.8: Resolução de Problema e ORG.2.1: Estabelecimento de Processo. A racionalidade desta escolha é que os processos CUS.3, MAN.2, SUP.3, SUP.2 formam a base do que deve ser melhorado, e os processos MAN.4, SUP.8 e ORG2.1 são instrumentos para esta melhoria. Esse método foi utilizado em pelos menos 22 empresas na Austrália e a evolução da melhoria nessas empresas foi acompanhada [Cater-Steel et al. 2004].

OWPL (*Observatoire Wallon des Pratiques Logicielles, Walloon Observatory for Software Practice*, Observatório de Wallon para Práticas de Software) é uma metodologia para início de melhoria de processo em micro e pequenas empresas [Habra et al. 1999, Habra et al. 2002, Laporte et al. 2005] [Figura 44].

O modelo OWPL é composto por dez processos, cada um com de três a doze práticas, baseados nos modelos SW-CMM e ISO/IEC TR 15504: Gerenciamento de requisitos, planejamento de projeto, acompanhamento e monitoramento de projeto, desenvolvimento, documentação, teste, gerência de configuração, gerência de fornecimento, gerência da qualidade e capitalização de experiência. Esses processos formam um perfil de capacidade fixo para a avaliação e a melhoria.

O método OWPL é composto por três fases. Na primeira fase são feitas uma micro-avaliação da empresa, algumas ações de melhoria e uma nova micro-avaliação após a melhoria. Esta micro-avaliação utiliza um questionário, com um total de 18 questões, que cobrem uma descrição da empresa, comentários sobre o questionário, e seis áreas: garantia da qualidade, gerência de clientes, gerência de fornecimento, gerência de projeto, gerência de produto, e treinamento e gerência de recursos humanos. Na segunda fase são feitas uma avaliação da empresa com o modelo OWPL, algumas ações de melhoria e uma nova avaliação após a melhoria. Na terceira fase são feitas uma avaliação da empresa com o modelo ISO/IEC 15504-5 ou SW-CMM (ou um modelo do CMMI), algumas ações de melhoria e uma nova avaliação após a melhoria.

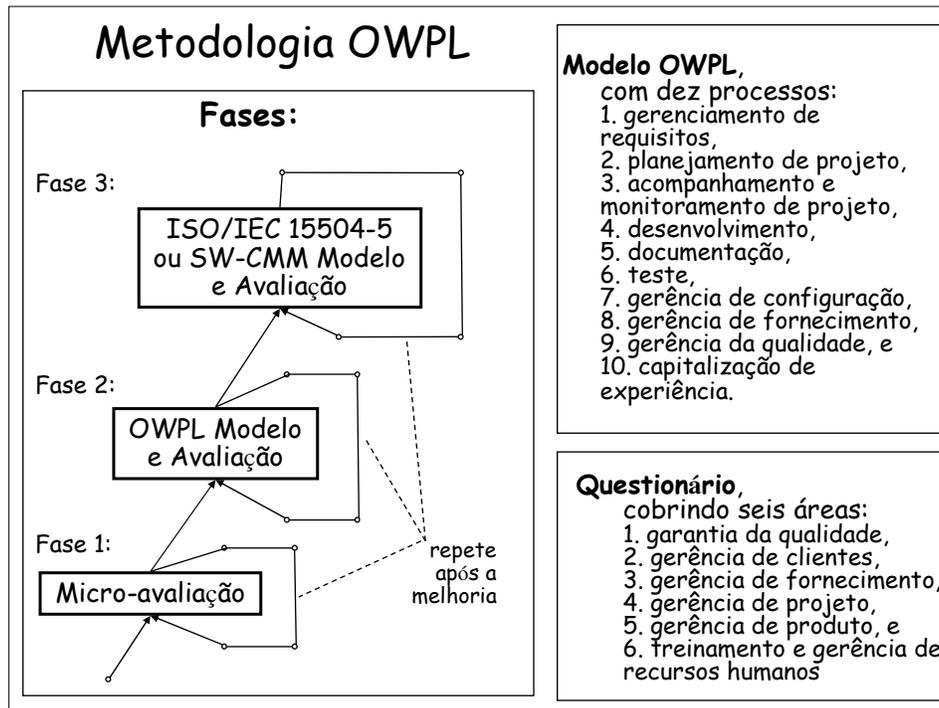


Figura 44 – Metodologia OWPL

De maio de 2003 a junho de 2004, uma equipe de pesquisadores do SEI e do AMRDEC-SED (*Aviation and Missele Research, Development and Engineering Center – Software Engineering Directorate*, Diretório de Engenharia de Software do Centro de Pesquisa, Desenvolvimento e Engenharia de Aviação e Mísseis) participaram de projetos de melhoria de processo em duas organizações pequenas para estudar a viabilidade técnica da aplicação de práticas do CMMI para melhorar o negócio destas organizações [Garcia et al. 2004]. Os elementos e resultados desses projetos estão consolidados no repositório *CMMI in Small Settings Toolkit* [Garcia et al. 2005]. A partir do resultado desses projetos, SEI está desenvolvendo uma abordagem para aplicação do CMMI em pequenos empreendimentos [Garcia 2005]. Um pequeno empreendimento é definido de forma aproximada como uma organização com menos de 100 pessoas (pequena organização), unidade organizacional com menos de 50 pessoas (pequena unidade organizacional) e projeto com menos de 20 pessoas (pequeno projeto).

A abordagem resultante desse projeto é composta por cinco fases: (i) reunião de início, (ii) sessão de análise da diferença, (iii) implementação do plano de ação, (iv) execução dos novos processos, e (v) encerramento da interação entre os consultores e o projeto. O projeto procurou minimizar a quantidade de consultoria na empresa com um total de dois dias de treinamento, dois dias de avaliação no local e um dia de consultoria por mês.

Como é utilizada a representação contínua do CMMI-SE/SW, uma atividade importante é a escolha, para cada organização de três áreas de processo para orientar a melhoria. Para tanto, é realizado um tutorial com uma visão geral do CMMI que cobre as áreas de processo dos níveis 2 e 3 de maturidade. Para cada uma destas áreas de processo, são apresentados o propósito, objetivos e práticas da área, conforme descrito no CMMI, e um relato dos impactos típicos da área no negócio de uma organização. É realizada então uma análise de cada área para captar as áreas de processo com maior impacto e maior necessidade para a organização, por meio da compilação da classificação feita por cada participante do tutorial em termos de alto, médio ou baixo impacto e necessidade.

As principais conclusões da viabilidade técnica da utilização do CMMI em pequenos empreendimentos foram:

- “CMMI provê um conjunto de melhores práticas das quais os pequenos empreendimentos podem se beneficiar;
- a representação contínua do CMMI permite que pequenos empreendimentos foquem nas melhorias que têm um melhor retorno de investimento para o empreendimento e aprendam sobre benefícios de outros elementos do modelo;
- o alinhamento da melhoria com os objetivos estratégicos é particularmente importante para pequenos empreendimentos, e nesse caso isto foi facilmente atingido;
- melhorias simples baseadas no CMMI podem ter um impacto significativo em pequenos empreendimentos;
- mudança de práticas para seguir o modelo não é necessário na maioria das vezes; encontrar práticas alternativas e ser criativo nos produtos de trabalho é geralmente mais relevante; e
- ambos o CMMI e SCAMPI A (o método de avaliação do CMMI) podem ser simplificados para atender pequenos empreendimentos” [Garcia 2005, slide 21].

## 2.5 Síntese do capítulo

Esse capítulo descreve resumidamente a área de pesquisa com a apresentação de uma revisão bibliográfica do vocabulário, conceitos, modelos, abordagens e tendências mais relevantes para esta pesquisa. As duas seções iniciais descrevem considerações adicionais sobre a apresentação da área e sobre a metodologia e processo de pesquisa. Vinte e três conceitos básicos, e seus relacionamentos, da área são identificados, com destaque para os conceitos de área de processo, nível de capacidade de processo e perfil de capacidade de processo.

Os principais relacionamentos entre os vinte e dois conceitos principais, identificados na Figura 4, que caracterizam a área de melhoria de processo de software são resumidos a seguir. Neste resumo, cada relacionamento entre dois conceitos é identificado como um texto sublinhado e cada conceito é identificado como um texto em **negrito**.

A área de **melhoria (ou engenharia) de processo de software (SPI/SPE)** envolve o uso de abordagem para SPI/SPE para a melhoria de processo, orientada a um modelo de capacidade de processo (MCP), alinhada ao contexto e objetivos estratégicos de uma organização. A **melhoria (ou engenharia) de processo de software (SPI/SPE)** utiliza também conceitos das áreas de **engenharia de software, modelo, engenharia de modelos e medição**. Um **processo** é documentado por representação de processo que é baseada em arquitetura de processo.

Uma **abordagem para SPI/SPE** é composta por atividades para o **estabelecimento de processo, avaliação de processo e melhoria de processo**. Uma **avaliação de processo** produz um **perfil resultante de avaliação**. Uma **melhoria de processo** é orientada por um **perfil alvo para a melhoria**, e complementada adicionalmente por outros tipos de modelo de referência, não PCM.

Um **modelo de capacidade de processo (PCM)** é construído com a utilização da **engenharia de PCM, com base no** conceito de **capacidade de processo**, que é uma previsão do desempenho de processo. A **engenharia de PCM** é responsável por frameworks de PCM, e define arquiteturas de PCM. Atualmente existem duas arquiteturas: **arquitetura estagiada** e **arquitetura contínua**. Um modelo com **arquitetura estagiada** é composto por níveis de maturidade e um modelo com **arquitetura contínua** é composto por níveis de capacidade de processo e áreas de processo. Um modelo com **arquitetura contínua** é utilizado por meio de perfil de capacidade de processo, que é composto por um conjunto de cada área de processo em um determinado nível de capacidade de processo. Um **nível de maturidade** é um exemplo de um perfil de capacidade de processo. Um **perfil resultante de avaliação** ou um **perfil alvo para a melhoria** é também um exemplo de perfil de capacidade de processo.

Vinte e seis modelos foram identificados como mais relevantes para a melhoria de processo e para esta pesquisa. Esses modelos de processos foram classificados em modelos de capacidade de processo e outros tipos de modelos de referência. Os modelos de capacidade de processo, que incluem aqueles denominados originalmente de modelos de maturidade, identificados foram: SEI SW-CMM, ISO/IEC 15504-5, FAA iCMM, SEI CMMI-SE/SW representação estagiada, SEI CMMI-SE/SW

representação contínua, PMI OPM3, ITsqc eSCM-SP, ITGI COBIT, MPS-BR MR-MPS, SM<sup>MM</sup>, KMMM, PMMM, UMM, TMM, OOSPICE, SPICE4SPACE, Automotive SPICE, 15504MPE e S9K. O modelo ISO/IEC 15504-5 foi identificado como representante mais significativo dos modelos do framework ISO/IEC 15504. O modelo SEI CMM, apesar de já estar aposentado e substituído pelos modelos do CMMI, foi identificado devido a sua importância como o modelo que estabeleceu a área de melhoria de processo. Os modelos do CMMI estão representados pelo modelo SEI CMMI-SE/SW e as duas representações desse modelo, e de qualquer outro modelo do CMMI, são consideradas como dois modelos para esta pesquisa. Os modelos de outros tipos, que não de capacidade de processo, identificados foram: ISO 9001, EFQM, PNQ, ISO/IEC 12207, PMBOK, RUP e SWEBOK.

Esses vinte e seis modelos são considerados na pesquisa em quatro níveis de relevância. No nível mais relevância são considerados os modelos ISO/IEC 15504-5, FAA iCMM, SEI CMMI-SE/SW representação estagiada, SEI CMMI-SE/SW representação contínua e MPS-BR MR-MPS. No segundo nível de relevância são considerados os modelos ISO 9001 e PMBOK como representantes. No terceiro nível de relevância são considerados os modelos SEI CMM, PMI OPM3, ITsqc eSCM-SP, ITGI COBIT, ISO 9001, EFQM, PNQ, ISO/IEC 12207, PMBOK, RUP e SWEBOK. No quarto e último nível de relevância são considerados os modelos OOSPICE, SPICE4SPACE, Automotive SPICE, 15504MPE, S9K, SM<sup>MM</sup>, KMMM, PMMM, UMM e TMM.

A Figura 45 relaciona quinze modelos dos três níveis mais relevantes, agrupa os modelos nos três tipos, e continua o relacionamento entre esses tipos e o processo de uma organização, já ilustrados na Figura 17.

Várias abordagens para melhoria de processo são também identificadas e relacionadas, com destaque para as abordagens PDCA, IDEAL, ciclo de melhoria da 15504 e AMP1.

Esse capítulo é concluído com um balanço de tendências de modelos e abordagens para melhoria de processo mais relevantes para a pesquisa, incluindo relacionamentos e integração de modelos, comparações e combinações das arquiteturas estagiada e contínua, e escolha de processos de modelos contínuos. As oportunidades para evolução da melhoria de processo de software envolvem uma combinação das arquiteturas estagiada e contínua.

Parte dessas tendências podem ser classificadas pela orientação de cada proposta em relação a duas dimensões. Uma dimensão é o quanto a posição destaca uma arquitetura ou outra (dimensão de arquitetura: estagiada ou contínua). Outra dimensão é o quanto a posição destaca a criação de modelo ou a utilização de modelo (dimensão de foco: modelo ou utilização). A Figura 46 indica essas posições sobre modelo e utilização das arquiteturas estagiada e contínua.

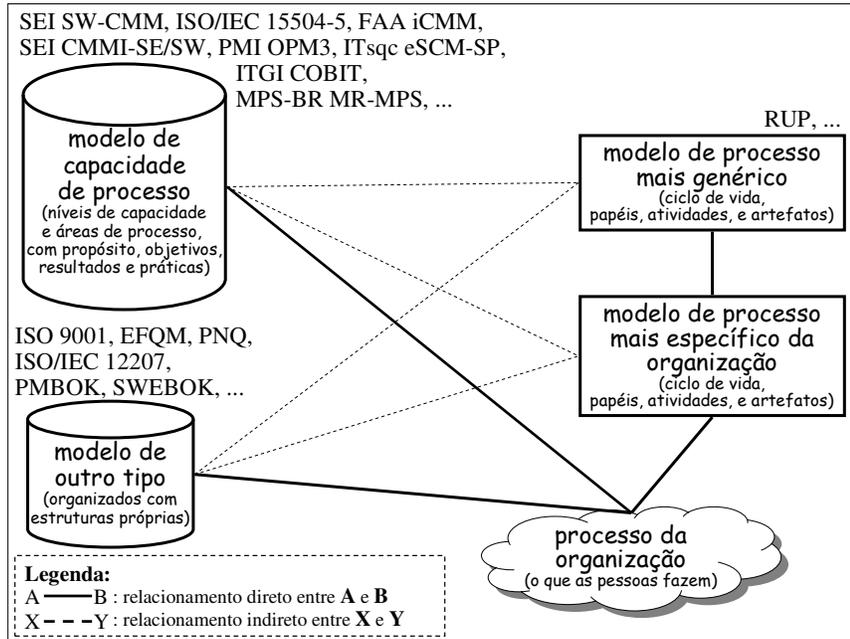


Figura 45 – Quinze modelos, seus tipos e relacionamentos com o processo

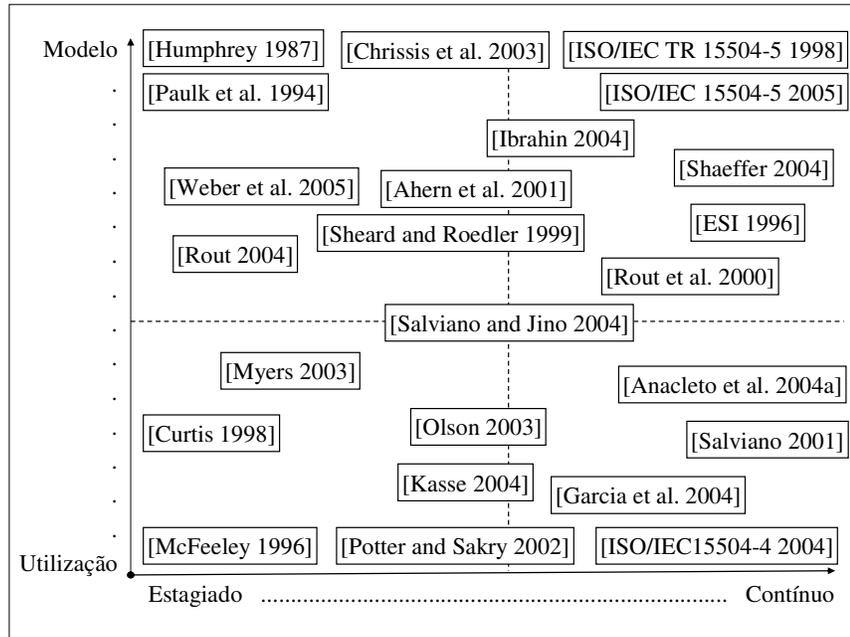


Figura 46 – Posições sobre modelo e utilização das arquiteturas estagiada e contínua

## Capítulo 3

### 3 Gerações de arquiteturas de modelos de capacidade de processo

O objetivo deste capítulo é apresentar reflexões sobre os relacionamentos entre as arquiteturas estagiada e contínua para modelos de capacidade de processo e buscar uma melhor caracterização das diferenças e similaridades entre as duas arquiteturas, para subsidiar uma evolução da melhoria de processo. Essa melhor caracterização é feita com a identificação de gerações de arquiteturas de modelos de capacidade de processo. O capítulo é iniciado com uma descrição do desenvolvimento e utilização de uma abordagem para melhoria de processo, incluindo um método para escolhas de processos a serem melhorados. O desenvolvimento e utilização dessa abordagem foram o início desta pesquisa e foram os antecedentes para a identificação das gerações de arquiteturas.

#### 3.1 Antecedentes para as gerações de arquiteturas

Esta pesquisa foi iniciada com o desenvolvimento e utilização da abordagem AMP1 e o método MEP1. A Figura 47 ilustra as fases da abordagem AMP1 (Abordagem para Melhoria de Processo).

A abordagem AMP1 [Salviano et al. 1999] é baseada no método IDEAL [McFeeley 1996] e no ciclo de melhoria da ISO/IEC TR 15504-7 [1998], utiliza áreas de processo de qualquer modelo, ou de combinações de modelos de capacidade de processo, incluindo os modelos ISO/IEC TR 15504-5, SW-CMM, CMMI-SE/SW e MR-MPS. AMP1 orienta uma melhoria alinhada com o contexto e objetivos estratégicos de uma organização e resultados e experiências de outras organizações, baseado na visão que “não existem duas organizações iguais e não existem duas organizações totalmente diferentes” [Weinberg 1993]. AMP1 orienta um ciclo de melhoria com seis fases, iniciadas quando existe uma decisão e comprometimento para a melhoria. As seis fases seqüenciais são: inicia trabalhos e define metas, avalia práticas correntes, planeja ações de melhoria, implementa ações de melhoria, verifica resultados e aprende, e institucionaliza a melhoria. O resultado principal é a melhoria da organização.

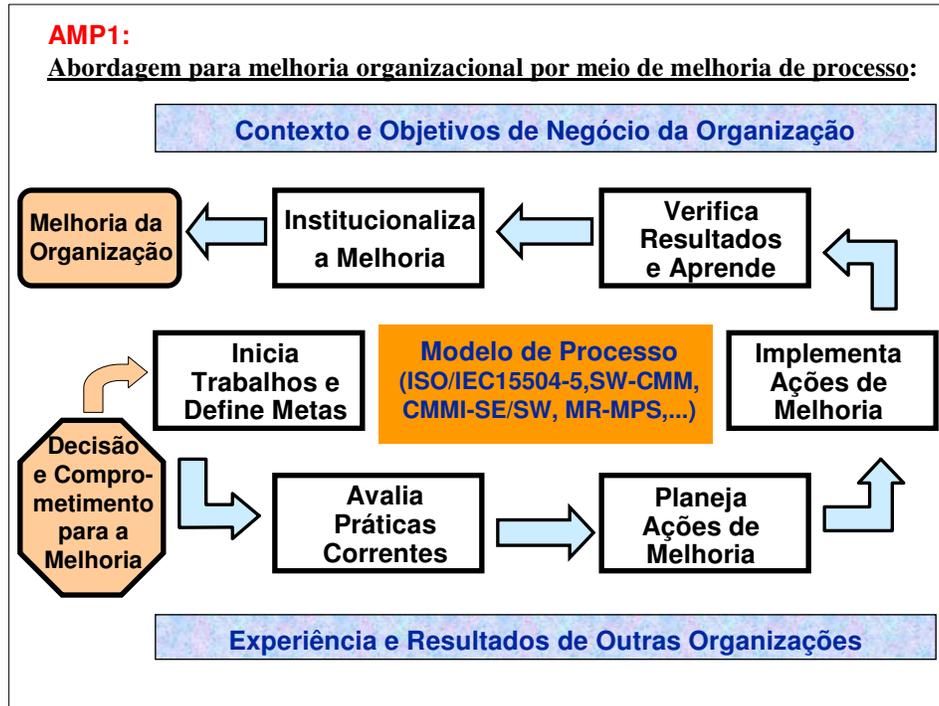


Figura 47 – Abordagem AMP1 para Melhoria de Processo

A abordagem AMP1 foi desenvolvida e utilizada, ao invés da utilização do método IDEAL ou do ciclo de melhoria da ISO/IEC TR 15504-7 porque existia a percepção que embora os dois métodos proviam uma orientação adequada para a melhoria, o detalhamento não eram adequados para uma utilização em micro, pequenas ou mesmo médias organizações com baixa capacidade de processo. Também, por ser o início de uma pesquisa, era importante ter todo o domínio sobre uma abordagem.

Partes das atividades da fase 1 de AMP1 (inicia trabalhos e define metas), podem ser realizadas com o método MEP1 (Método para Escolha de Processos) [Salviano 2001], para orientar a escolha dos processos a serem melhorados de forma a realizar uma melhoria alinhada ao contexto e objetivos estratégicos da organização, conforme ilustrado na Figura 48. A Figura 49 ilustra as nove atividades do método MEP1, organizadas em três fases.

A atividade 0 do método MEP1 é a preparação para as atividades seguintes. A atividade 1 é o levantamento dos objetivos estratégicos da organização. A atividade 2 é o posicionamento de um conjunto pré-selecionados de processos em uma quadro de importância para a unidade organizacional e risco da execução atual para a unidade organizacional. A atividade 3 é uma sessão de respostas a um questionário com opiniões sobre vários aspectos da organização.

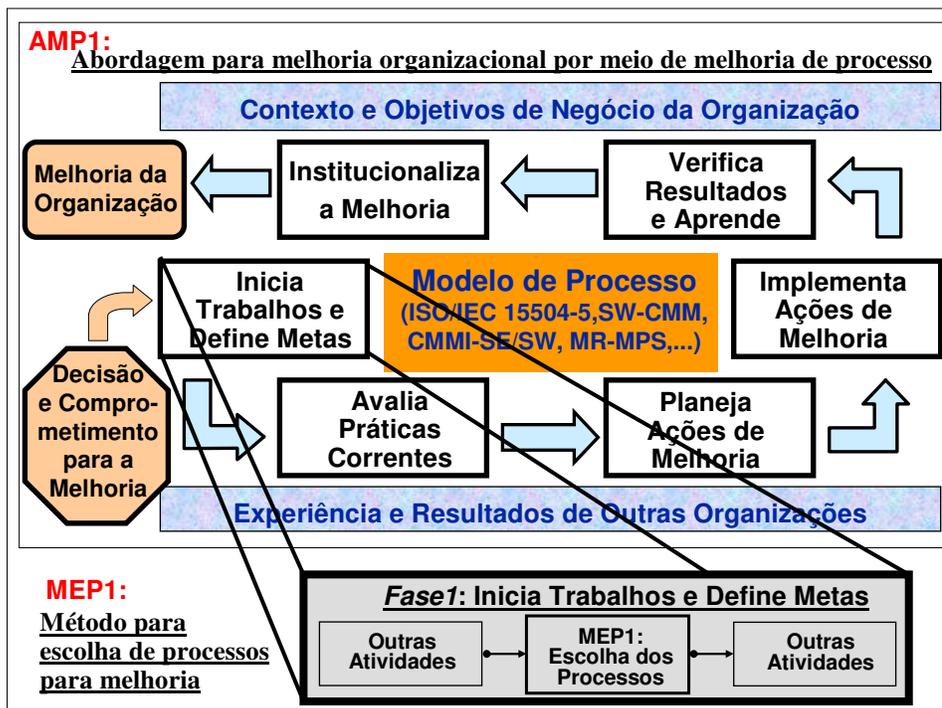


Figura 48 – Abordagem AMP1 e método MEP1

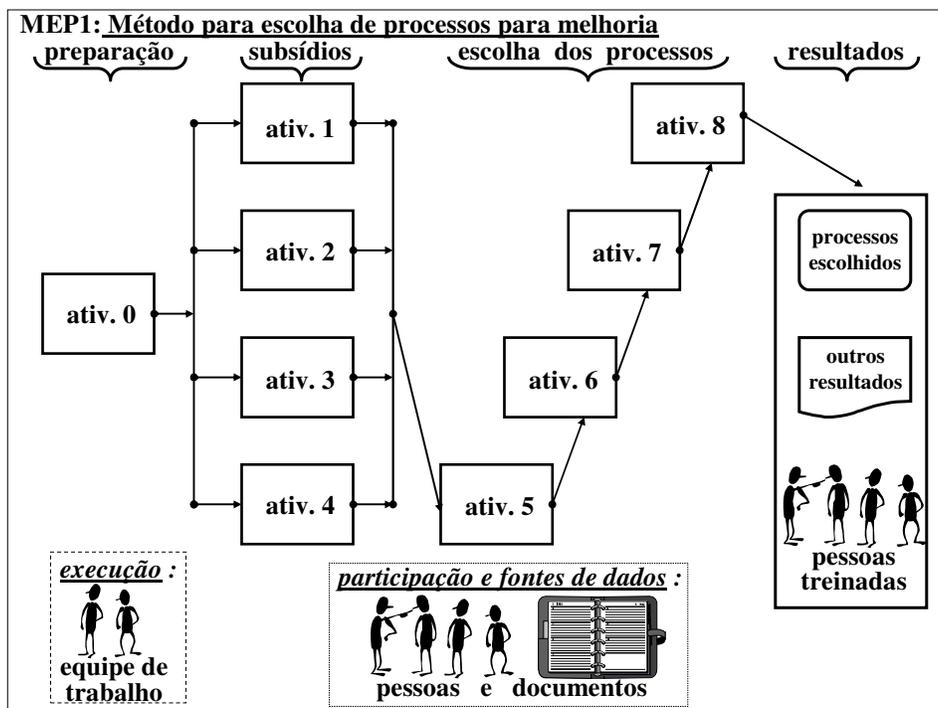


Figura 49 – Fases, atividades e resultados do método MEP1

A atividade 4 do método MEP1 é a consideração de experiências de outras organizações, com a seleção dos processos mais utilizados em melhoria de processo nessas outras organizações. A

atividade 5 é o mapeamento dos subsídios, identificados nas atividades 1, 2, 3 e 4, nos processos do modelo de referência. A atividade 6 inicia a fase de escolha dos processos com a definição preliminar dos processos. A atividade 7 é a análise e revisão da escolha. A atividade 8 é a revisão da escolha com o patrocinador.

AMP1 e MEP1 foram utilizados em projetos de melhoria de processo, incluindo aqueles descritos em Salviano et al. [1999], Nicoletti e Salviano [2003], Salviano e Souza [2000], e Silva et al. [2003]. O desenvolvimento e utilização de AMP1 e MEP1 iniciaram esta pesquisa com o reconhecimento do impacto dos níveis de maturidade do SW-CMM, dos níveis de maturidade do estagiado como um uso específico do contínuo, dos níveis de capacidade de processo como a referência conceitual básica da melhoria de processo e os perfis de capacidade de processo como o meio de utilização dos modelos contínuos, e também dos modelos estagiados.

### **3.2 Pressupostos para as gerações de arquiteturas**

O Modelo SW-CMM é o principal responsável pelo estabelecimento da área de Melhoria de Processo de Software. Os resultados dos programas de melhoria de processo que utilizaram o modelo SW-CMM como referência para as ações de melhoria do processo de desenvolvimento de software e para a avaliação desse processo estabeleceram esta área tanto como área de pesquisa e desenvolvimento quanto como área de prática. O impacto foi causado tanto no estado da arte como no estado da prática de engenharia de software.

Os elementos principais do modelo SW-CMM são os quatro níveis de maturidade (2, 3, 4 e 5) e o relacionamento sequencial e cumulativo entre eles. Cada nível de maturidade contém um conjunto específico de áreas chave de processo, que são utilizadas como referência para a melhoria e avaliação do processo de desenvolvimento de software de qualquer unidade organizacional intensiva em software, independentemente de qualquer característica específica desta unidade, seja quanto à estratégia de negócio, tamanho, localização geográfica, natureza, etc.

A característica de oferta de apenas uma solução para qualquer problema de uma área tem sido utilizada no estabelecimento dessa área. O exemplo mais clássico é o do automóvel preto Ford T. Ford não inventou o automóvel mas viabilizou o automóvel como meio de consumo, com a produção a custo acessível de um único modelo e de todo o ambiente necessário para sua utilização, incluindo estradas e postos de abastecimento. A analogia com os níveis de maturidade do SW-CMM é imediata.

O debate entre as vantagens e desvantagens das arquiteturas estagiada e contínua, tem sido dominada por visões favoráveis à estagiada, como, por exemplo, Curtis [1998], favoráveis à contínua, como, por exemplo, ESI [1996], e uso do modelo estagiado com visões do contínuo, como, por exemplo, Chrissis et al. [2003] e Kasse [2004]. Todas elas tratam as duas arquiteturas como duas alternativas para melhoria de processo.

A arquitetura contínua não é uma alternativa ou concorrente da arquitetura estagiada, mas sim uma evolução, com a abstração em níveis de capacidade. A arquitetura estagiada é apenas um uso específico dessa abstração. Arquitetura contínua é como o sistema métrico, enquanto que a arquitetura estagiada é como escala normal de crescimento de adolescentes, que define faixas consideradas normais de tamanho em relação à idade, expressas em termos de metros e centímetros. Os quatro níveis de maturidade podem ser vistos como um exemplo de utilização da arquitetura contínua. Esta utilização é para um contexto mais específico, que é projetos de desenvolvimento de software sob encomenda.

Como a arquitetura continua (C) define duas dimensões ( $C^2$ ), nas quais deve ser definido um modelo (M) que corresponde na verdade a um nível de maturidade que caracteriza a arquitetura estagiada (E), pode ser proposta a expressão  $E = MC^2$  como representação informal, pois não é uma equação matemática, da relação da atual arquitetura estagiada com a arquitetura contínua.

Os seis níveis de capacidade de processo, conforme definidos na ISO/IEC 15504-2 [2003] e utilizado, com algumas adaptações, pelos modelos do CMMI [Chrissis et al. 2003] formam atualmente a referência conceitual básica da melhoria de processo em geral e da melhoria de processo de software em particular. Essa referência conceitual está documentada na Norma ISO/IEC 15504-2, com o denominação de estrutura de medição de processo e é composta por seis níveis de capacidade de processo. Segundo a ISO/IEC 15504-2, um processo pode ser caracterizado por um de seis níveis, seqüenciais e cumulativos, de capacidade, identificados por um número e um nome, conforme Figura 50.

Cada um dos atributos de processo medem aspectos que, junto com os outros atributos do mesmo nível de capacidade, representam as características do nível. Com isto, a 15504 define um framework para medição da capacidade de um processo. A capacidade de um processo é definida numa escala ordinária de seis pontos que permite a avaliação da capacidade desde o fundo da escala, Incompleto, até o seu topo, Em Otimização. A escala representa a capacidade de crescimento do processo implementado, desde não estar atingindo o propósito do processo até estar atingindo os

objetivos atuais e projetados do negócio. Desta forma, a escala define uma rota para a melhoria de cada processo.

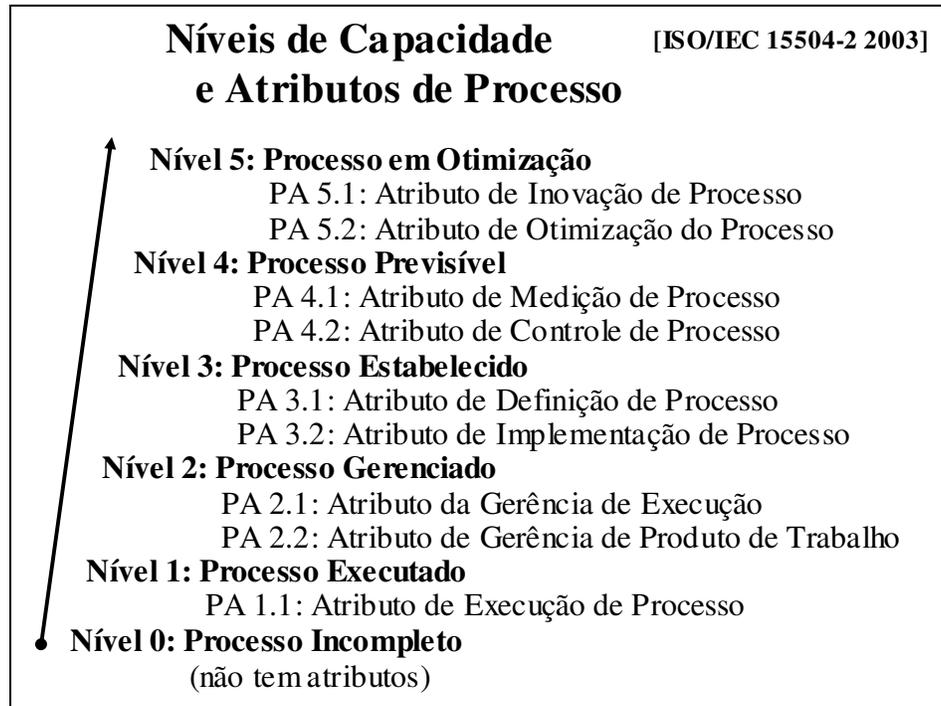


Figura 50 – Níveis de Capacidade [ISO/IEC 15504-2 2003]

O conceito de Perfil de Capacidade de Processo (PCP) é o unificador entre as arquiteturas estagiada e contínuo, a base para entender as gerações de modelos e a base da metodologia proposta nesta tese.

A questão das arquiteturas estagiada e contínua pode ser melhor caracterizada como três gerações de arquiteturas: estagiada fixa, contínua fechada e contínua aberta<sup>9</sup>. A Tabela 16 relaciona as três gerações de arquiteturas de modelos e identifica suas principais características. As características identificadas são relacionadas ao framework ou modelo principal da geração, o ano de lançamento desse framework ou modelo, o nome atual da arquitetura, outros modelos que podem ser classificados na geração, os principais elementos fixos, que são os pontos de estabilidade, os principais elementos variáveis, que são os pontos de flexibilidade, algumas abordagens exemplo e comentários e implicações da geração para a melhoria de processo.

<sup>9</sup> A identificação das arquiteturas de frameworks de modelos de capacidade de processo como gerações foi inspirada na identificação de gerações de uso de e-commerce descrita em Vu [2001].

Tabela 16 – Gerações de Arquitetura de Modelos de Capacidade de Processo

	<b>Primeira geração</b>	<b>Segunda geração</b>	<b>Terceira geração</b>
Framework ou modelo principal	Modelo SW-CMM v. 1.1	Framework ISO/IEC TR 15504	Framework ISO/IEC 15504
Ano de lançamento	1993	1998	2003
Nome atual	Estagiado	Contínuo	Contínuo
Outros modelos	CMMI-SW Est. v1.1 MR-MPS	ISO/IEC 15504-5 iCMM, CMMI-SW Con.	Utilizações de PRO2PI
Nome proposto	<i>Estagiado fixo</i>	<i>Contínuo fechado</i>	<i>Contínuo aberto</i>
Principais elementos fixos (estabilidade)	Hierarquia de Perfis de Capacidade de Processo (PCPs)	Níveis de capacidade, Áreas de processo (AP) e requisitos para PCPs	Níveis de capacidade, e requisitos para APs e PCPs
Principais elementos variáveis (flexibilidade)	Interpretação dos PCPs	Escolha e interpretação de PCPs	Escolha e definição de APs, e escolha e interpretação de PCPs
Abordagens	MPS, IDEAL	MEP1, Toolkit SEI	PRO2PI-CYCLE
Comentários e implicações para a melhoria	Sucesso, fundamental para estabelecer a área de MPS	Evolução, separação entre essência (APs e NCs) e uso (PCPs)	Evolução, essência (NCs) e uso (APs e PCPs), nova área: POHE

### 3.3 Primeira geração: Arquitetura estagiada fixa

Esta primeira geração é caracterizada pelo modelo SW-CMM. Os cinco níveis de maturidade são fixos, com os níveis 2 a 5 compostos por conjuntos fixos de áreas chave de processo. Não existe a separação entre framework e modelos, apenas o modelo.

Uma utilização de um modelo estagiado fixo deve utilizá-lo exatamente como ele é, podendo apenas interpretar as orientações para o contexto da utilização.

Esta geração é caracterizada por uma separação rígida entre construção de modelos (na verdade construção do modelo) e de sua forma de utilização (por meio de níveis de maturidade), e a utilização do modelo. O primeiro cabe a entidades, cabendo às organizações apenas o segundo.

Em termos dos três conceitos (área de processo, níveis de maturidade de processo e perfis de capacidade de processo), os modelos desta geração são compostos por uma sequência cumulativa de PCPs, onde cada um é composto por uma combinação fixa de áreas de processo em um determinado

nível de capacidade de processo. Os níveis 4 e 5 são na verdade uma simulação da arquitetura contínua por meio da arquitetura estagiada.

Os modelos versões 1.1 do CMMI, na representação estagiada, mantêm as características da geração de modelos estagiados fixos, mas já apresentam o conceito de framework e o mapeamento como um exemplo do uso de modelos contínuos é mais claro. Outros exemplos de modelos estagiados fixos são o modelo MR-MPS do MPS.BR [Weber et al. 2004a] e o OPM3 [PMI 2003].

### **3.4 Segunda geração: Arquitetura contínua fechada**

A segunda geração definiu a separação de um nível de maturidade como a composição de dois elementos: área de processo e nível de capacidade de processo. Nela, as áreas de processo e os níveis de capacidade de processo são fixos e as suas combinações são variáveis.

Esta geração é caracterizada por uma separação menos rígida entre construção de modelos e de sua forma de utilização (níveis), e a utilização do modelo. O primeiro ainda cabe a entidades, mas o segundo pode caber tanto a entidades quanto às organizações e naturalmente o terceiro continua cabendo às organizações.

Exemplos de frameworks são a ISO/IEC TR 15504 e o framework do CMMI. Modelos são o ISO/IEC TR 15504-5 e as representações contínuas dos modelos do CMMI. O modelo CMMI-SE/SW versão 1.1 é um exemplo de um modelo contínuo fechado, com suas vinte e duas áreas de processo e a não possibilidade de definição de outras áreas de processo.

Apesar de alguns modelos permitirem a inclusão de novas áreas de processo, como, por exemplo, o modelo ISO/IEC 15504-5, a existência de um conjunto de áreas de processo já definidas é a característica mais forte. A ISO/IEC 15504-5, por exemplo, define quarenta e oito áreas de processo.

A utilização de um modelo passa pela escolha de um PCP a partir dos níveis de capacidade de processo e áreas de processo, seguindo as regras de composição. Deveria haver uma metodologia para apoiar esta escolha. A metodologia AMP1 [Salviano 2001] e *CMMI in Small Settings Toolkit* [Garcia et al. 2005] são exemplos disto. Um modelo da primeira geração, pode ser definido como um exemplo de utilização de um modelo desta segunda geração. Cada nível de maturidade é um perfil de capacidade de processo.

### **3.5 Terceira geração: Arquitetura contínua aberta**

A terceira geração fixou apenas os níveis de capacidade. Com isto podem ser definidos conjuntos de áreas de processo e então definidos combinações de áreas de processo e níveis de capacidade de processo em PCPs.

Esta geração é caracterizada por uma separação ainda menos rígida entre construção de modelos e de sua forma de utilização (níveis), e a utilização do modelo. O primeiro e o segundo podem caber tanto a entidades quanto às organizações e naturalmente o terceiro continua cabendo às organizações.

A utilização desta terceira geração passa por uma etapa anterior às da utilização da segunda geração, pois inclui a escolha ou definição das áreas de processo. Com isto aumentam a flexibilidade e as oportunidades de melhor alinhamento da melhoria com a organização, porém, em contrapartida, aumenta também o risco de escolhas inadequadas, o que leva ao aumento da necessidade de metodologias para sua utilização.

A proposta desta tese é estender o framework ISO/IEC 15504, como um representante desta terceira geração, para apoiar a sua utilização. Nesta visão é defendida a idéia de que, em geral, a melhoria de processo deve ser baseada em PCPs dinâmicos e específicos, ou em outra linguagem, “níveis de maturidade” dinâmicos e específicos. Com isto iremos diluir a separação rigorosa entre definição de modelos (hoje feita mais por institutos) e a utilização de modelos. Uma organização ao fazer melhoria de processo, também define modelos, por meio de perfis de capacidade de processo.

### **3.6 Síntese do capítulo**

Esse capítulo descreveu o início desta pesquisa, com o desenvolvimento e utilização da abordagem AMP1 e o método MEP1, quatro observações sobre o relacionamento entre as arquiteturas contínua e estagiada e uma proposta de caracterização das arquiteturas por meio de três gerações, como substituição à caracterização corrente como arquitetura contínua e estagiada. Estas três gerações de arquiteturas são nomeadas respectivamente de estagiada fixa, contínua fechada e contínua aberta.

# Capítulo 4

## 4 Engenharia de processo dirigida por perfis de capacidade

O objetivo deste capítulo é apresentar reflexões sobre as oportunidades para uma evolução da área de melhoria de processo de software, para uma melhoria mais viável e mais adequada aos objetivos, estratégia e contexto de negócio das organizações. Esse objetivo é complementado com uma proposta para esta evolução e considerações sobre os desdobramentos da proposta. Esta reflexão e evolução estão fundamentadas na utilização da terceira geração de arquitetura de modelos de capacidade, descrita no capítulo anterior, que oferece uma maior flexibilidade e é a arquitetura contínua aberta.

### 4.1 Fundamentos para uma engenharia de processo

A Melhoria de Processo de Software acrescenta uma dimensão à Engenharia de Software, que é a construção, por cada organização, dos processos para uma engenharia de software. Talvez uma das causas da dificuldade do atual estado da arte da engenharia de software em ser efetivamente utilizado na prática, esteja no uso da abordagem pesquisa-depois-transfere [Potts 1993]. A melhoria de processo de software pode ser vista como uma abordagem para a construção da engenharia de software, e com isto, não tem o ônus de definir a engenharia de software. Com isto uma nova dimensão é incluída ao atual estado da arte da engenharia de software: como orientar a construção dos processos. Essa construção é orientada pelos níveis de capacidade de processo.

A Melhoria de Processo de Software deve ser evoluída com sua a integração à engenharia de modelos de capacidade de processo. O conhecimento da melhoria de processo de software está atualmente dividido em duas áreas. Uma relacionada com a construção de modelos de capacidade de software, que é utilizada atualmente nos projetos CMMI, ISO/IEC 15504 e MPS.BR, por exemplo. Outra é a utilização desses modelos como referência para a melhoria.

A abrangência da Melhoria de Processo de Software é ampliada para qualquer área de tecnologia, intensiva em atividade criativa humana. Existe uma tendência atual de ampliação da

abrangência da melhoria de processo de software para melhoria de qualquer processo tecnológico que seja intensivo em atividade criativa humana. A evolução do SW-CMM, que era específico para software, para o CMMI, para sistemas, e da ISO/IEC TR 15504-5, que também era específico para software, para a ISO/IEC 15504, que é genérica, evidencia esta tendência.

A terceira geração de arquitetura de modelos de capacidade de processo provê uma base adequada para essa evolução da engenharia de processo

## 4.2 Definição de engenharia de processo

Um perfil de capacidade de processo é um modelo que representa um processo segundo o aspecto de capacidade de processo. O processo de uma organização deve ser representado por um perfil de capacidade de processo, que é uma abstração do processo, segundo o aspecto capacidade de processo. O par consistente formado por perfil de capacidade de processo e processo, é ilustrado na Figura 51.

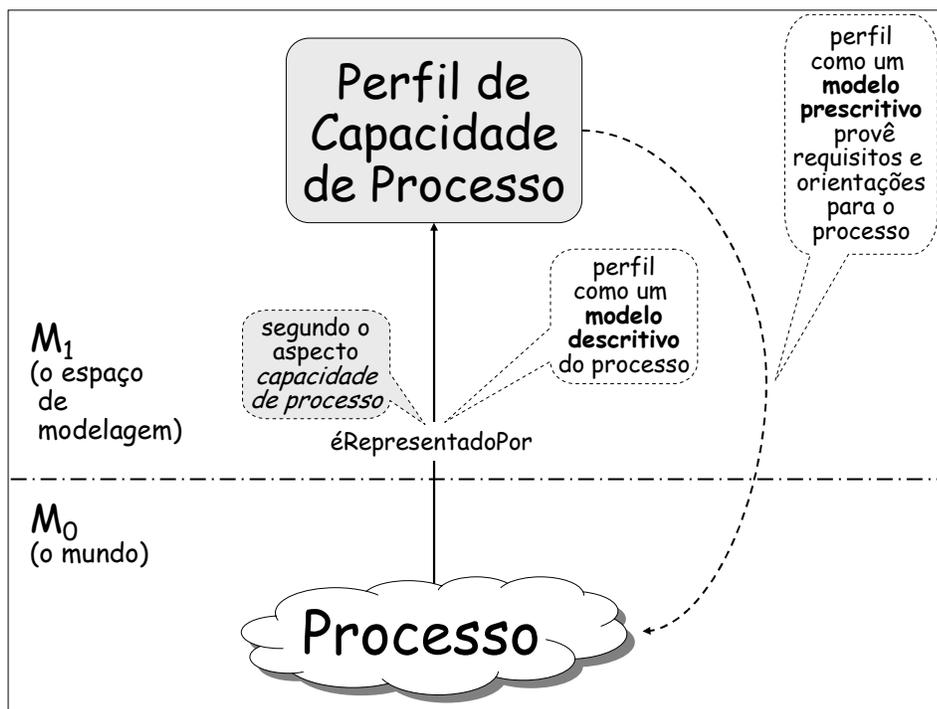


Figura 51 – Par consistente perfil de capacidade de processo e processo

Na Figura 51 um processo é uma parte do mundo ( $M_0$ ) e é representado, segundo o aspecto capacidade de processo, pelo modelo perfil de capacidade de processo no espaço de modelagem ( $M_1$ ). Um processo pode também ser representado por outros modelos, segundo outros aspectos, como, por,

exemplo, por um modelo com atores, atividades e artefatos, segundo o aspecto de descrição de processo. O perfil de capacidade de processo como representação do processo é um modelo descritivo. Esta representação pode ser obtida por uma avaliação da capacidade de processo. O perfil de capacidade de processo como requisitos e orientações para a melhoria de um processo é um modelo prescritivo. Este modelo representa o perfil alvo de uma melhoria de processo .

O processo atual pode ser representado por um perfil de capacidade de processo como um modo descritivo e o processo futuro pode ser representado por outro perfil de capacidade de processo como um modelo prescritivo. Em ambos os casos existe uma relação entre processo e perfil de capacidade de processo na qual o processo satisfaz completamente o perfil de capacidade de processo e o processo contém apenas o que é requisitado pelo perfil de capacidade de processo. Desta forma o princípio da ‘substitucionalidade’ limitada [Bézivin2003b] se aplica.

Outra forma de entender a relação entre o perfil e o processo é com o seguinte questionamento: se o perfil de capacidade de processo representado por um nível de maturidade do modelo CMMI-SE/SW tivesse sido definido para o processo de uma determinada unidade organizacional, em um determinado momento, qual seria este nível de maturidade de forma a representar o processo atual ou o processo alvo para uma melhoria alinhada ao contexto e objetivos estratégicos da organização.

A melhoria de processo deve ser evoluída para uma engenharia que trate esse par consistente no centro dessa engenharia. É proposto então, como uma evolução da atual melhoria de processo de software baseada em modelos de maturidade, uma Engenharia de Processo Dirigida por Perfil de Capacidade de Processo (*Process Capability Profile Driven Process Engineering - PCDE*) aplicada a software.

Seguindo a tendência de expansão da abrangência da melhoria de processo de software para melhoria de processo de software e de sistemas, evidenciada por exemplo pelo CMMI e ISO/IEC 15504, a engenharia de processo de software também deve ter essa expansão. Para isso é proposta a expressão “de software e de qualquer outro trabalho humano intensivo em conhecimento” na qual é ressaltada a abrangência dessa engenharia para software, que tem sido a área na qual é mais utilizada, e caracterizado de forma mais precisa a abrangência maior da engenharia, que é qualquer trabalho humano intensivo em conhecimento, incluindo trabalho de software. A escolha dessa terminologia é baseada no termo “trabalhador do conhecimento” (*knowledge worker*) definido originalmente por Peter Drucker no seu livro *Landmarks of Tomorrow* publicado em 1959 e consolidado, por exemplo, em Druker [1992]. Segundo Druker, “um trabalhador do conhecimento é aquele cujo trabalho envolve o desenvolvimento ou utilização de conhecimento, incluindo aqueles da tecnologia da informação,

como, por exemplo, programadores, analistas de sistemas, escritores técnicos, profissionais acadêmicos, pesquisadores e outros, e também incluindo pessoas fora da tecnologia da informação, como, por exemplo, advogados, professores, cientistas e estudantes” [searchCRM 2006].

Uma definição completa de engenharia de processo PCDE é apresentada no quadro a seguir.

**{(Engenharia de processo) (de Software e de qualquer outro Trabalho Humano Intensivo em Conhecimento) Dirigida por (Perfis de Capacidade de Processo)}**  
ou de forma mais curta  
**{(Engenharia de processo) Dirigida por (Perfis de Capacidade de Processo)}**  
é definida como:

- (1) a aplicação de engenharia à definição, utilização, manutenção, estabelecimento, medição, mudança, melhoria e co-evolução de pares consistentes de perfil de capacidade de processo e processo para uma melhoria de processo dirigida por perfis de capacidade de processo, em processos de software e de qualquer outro trabalho humano intensivo em conhecimento, como meio para atingir excelência organizacional alinhada ao contexto e objetivos estratégicos;
- (2) a aplicação de engenharia ao desenvolvimento e manutenção de modelos de capacidade de processo para contextos ou domínios estratégicos mais específicos; e
- (3) o estudo das abordagens como em (1) e (2).

Esta definição é baseada em uma combinação de definições de engenharia [Shaw 1990], engenharia de software [Abran et al. 2004], engenharia de processo de software [El Emam 2001], melhoria de processo de software [Zahran 1998, Caputo 1998] e engenharia dirigida por modelos [Bézivin 2003a].

A Figura 52 descreve um diagrama de Venn da relação de PCDE com outras engenharias.

A engenharia PCDE cobre a atual melhoria (ou engenharia) de processo de software e grande parte da atual engenharia de modelos de capacidade de processo. Apenas as atividades de consideração dos modelos das primeira e segunda gerações, como algo único, e não como um caso particular da terceira geração, não são cobertas na engenharia PCDE. O apoio para estabelecimento de perfis de capacidade de processo e a relação entre perfil e processo, não são cobertos pelas SPI/SPE e PCME.

O termo ‘dirigido’ é utilizado no sentido similar ao termo ‘orientado’, como em Engenharia de Software Orientado a Objetos (*Object-oriented Software Engineering*); ao mesmo termo ‘dirigido’,

como em Engenharia Dirigida por Modelos (*Model-Driven Engineering*) e ao termo ‘baseado’, como em Engenharia de Software Baseada em Componentes (*Component-Based Software Engineering*). A escolha foi feita pelo termo ‘dirigido’ pelo uso já estabelecido em engenharia, arquitetura e desenvolvimento dirigido por modelos. O termo baseado já é utilizado para melhoria de processo de software com o sentido de apenas utilizar modelos.

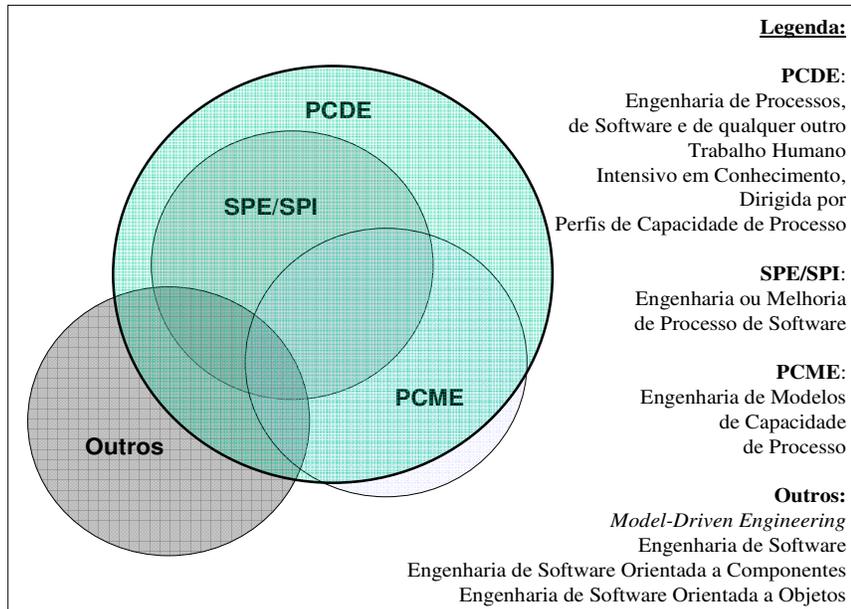


Figura 52 - Diagrama de Venn da PCDE e outras engenharias

### 4.3 Desdobramentos da engenharia de processo

Com PCDE as cinco áreas propostas por Kinnula [1999] são ampliadas para incluir uma área adicional, estabelecimento de perfil de capacidade de processo, conforme ilustrado na Figura 53.

A área de estabelecimento de processo envolve o projeto, implementação, refinamento e teste de mudanças em um processo para melhorar a sua capacidade em atingir as metas definidas do processo. A área de utilização de processo envolve o estabelecimento de instâncias de um processo. A área de manutenção de processo envolve a manutenção do processo. A área de gerência da engenharia de processo envolve o gerenciamento das atividades de engenharia de processo e o desenvolvimento e manutenção do processo. A área de estabelecimento de perfil de capacidade de processo envolve a definição e utilização de perfis de capacidade de processo. A área de melhoria de processo envolve a melhoria do processo. A área de avaliação de processo envolve a avaliação do processo. A área de gerência de ativos de processo envolve a gerência dos ativos do processo.

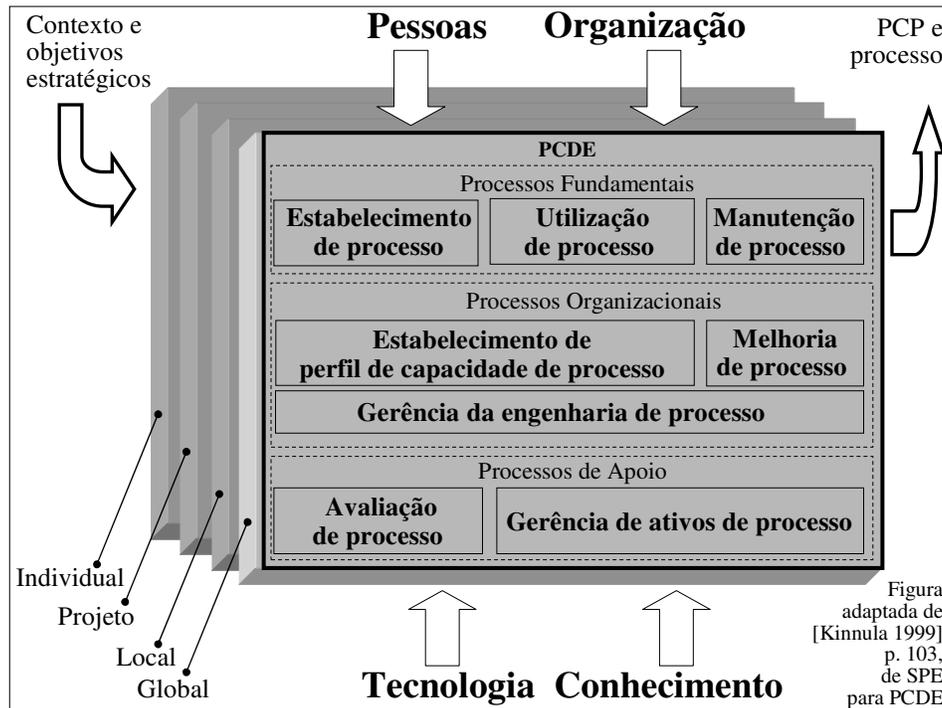


Figura 53 – Áreas e contexto da engenharia PCDE

O estabelecimento de perfil de capacidade de processo pode ser visto como uma área de processo. Essa área de processo está ilustrada na Figura 54.

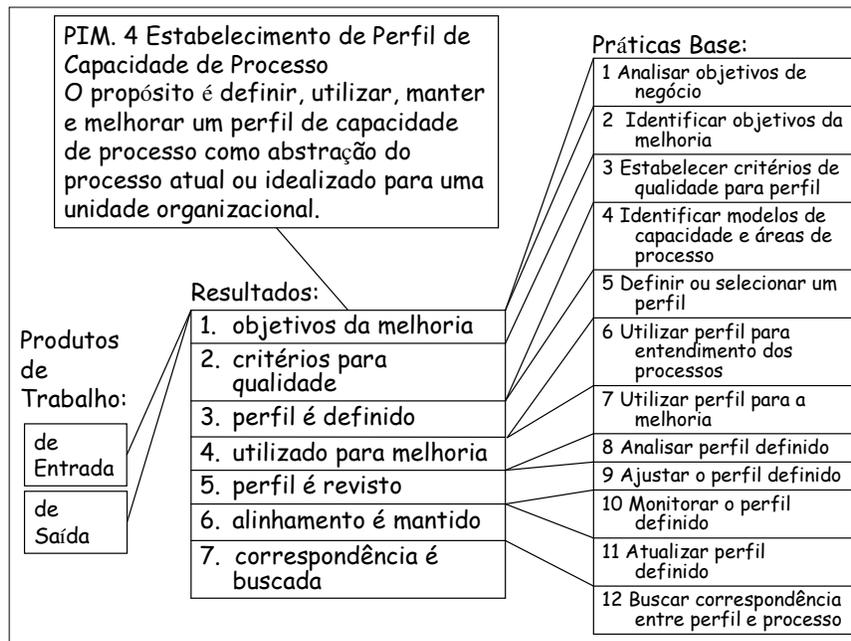


Figura 54 – (Área de) Processo de Estabelecimento de PCP

Essa área de processo está descrita na Tabela 17 segundo o formato utilizado na ISO/IEC 15504-5 [2005]. Como esta área de processo pode ser considerada como um quarto processo do grupo de processos de melhoria de processo (identificado pela sigla PIM) esse processo é identificado como PIM.4.

Tabela 17 – (Área de) Processo de Estabelecimento de PCP como 15504-5

<b>Identificação</b>	<b>PIM.4</b>
Nome	Estabelecimento de Perfil de Capacidade de Processo
Propósito	O propósito (da Área) de Processo de Estabelecimento de Perfil de Capacidade de Processo é definir, utilizar, manter e melhorar um perfil de capacidade de processo como abstração do processo atual ou idealizado de uma unidade organizacional.
Resultados	Como resultado de uma implementação com sucesso (da área) de processo de Estabelecimento de Perfil de Capacidade de Processo:
	1.os objetivos da melhoria são identificados, de forma alinhada aos objetivos estratégicos;
	2.critérios para qualidade do perfil são estabelecidos;
	3.um perfil de capacidade de processo é definido;
	4.o perfil é utilizado para orientar a melhoria;
	5.o perfil é revisto quando necessário, segundo os critérios definidos;
	6.o alinhamento entre o perfil e os objetivos estratégicos é mantido; e
	7. a correspondência entre o perfil e o processo é buscada.
Práticas base	<b>PIM.4.BP.1 Analisar objetivos estratégicos.</b> Identificar e analisar os objetivos, estratégia, contexto e/ou qualquer outro aspecto relevante de negócio da unidade organizacional e da organização, para subsidiar e orientar a definição dos objetivos de melhoria. [resultado 1]
	<b>PIM.4.BP.2 Identificar objetivos da melhoria.</b> Identificar os objetivos da melhoria, incluindo objetivos mais específicos para o próximo ciclo de melhoria e objetivos mais gerais para o programa de melhoria, sempre alinhado aos objetivos, estratégia, contexto e/ou qualquer outro aspecto relevante de negócio identificados. [resultado 1]
	<b>PIM.4.BP.3 Estabelecer critérios de qualidade para perfil.</b> Estabelecer critérios de qualidade para avaliar e melhorar um perfil de capacidade de processo. [resultado 2]
	<b>PIM.4.BP.4 Identificar modelos de capacidade e áreas de processo.</b> Identificar modelos de capacidade de processo que sejam relevantes para a organização, e selecionar áreas de processo desses modelos e, caso necessário, definir novas áreas de processo, que sejam relevantes para o processo atual e sua melhoria. [resultado 3]
	<b>PIM.4.BP.5 Definir ou selecionar um perfil.</b> Definir, selecionar e/ou adaptar um perfil de capacidade de processo com áreas de processo selecionadas e/ou definidas, de forma que esse perfil esteja alinhado aos objetivos de melhoria. [resultado 3]
	<b>PIM.4.BP.6 Utilizar perfil para entendimento dos processos.</b> Utilizar o perfil de capacidade de processo definido para orientar o entendimento dos processos correntes da unidade organizacional. [resultado 4]

	<b>PIM.4.BP.7 Utilizar perfil para a melhoria.</b> Utilizar o perfil de capacidade de processo definido e os resultados do entendimento dos processos correntes, para orientar as ações de melhoria dos processos da unidade organizacional. [resultado 4]	
	<b>PIM.4.BP.8 Analisar o perfil definido.</b> Analisar o perfil de capacidade de processo definido segundo os critérios de qualidade estabelecidos, quando necessário, como, por exemplo, em marcos definidos do ciclo de melhoria, em eventos ou periodicamente. [resultado 5]	
	<b>PIM.4.BP.9 Ajustar o perfil definido.</b> Ajustar o perfil de capacidade de processo definido segundo resultados de análise do perfil. [resultado 5]	
	<b>PIM.4.BP.10 Monitorar o perfil definido.</b> Monitorar o perfil de capacidade de processo definido em relação aos objetivos estratégicos para identificar possíveis desvios no alinhamento entre o perfil e os objetivos estratégicos. [resultado 6]	
	<b>PIM.4.BP.11 Atualizar perfil definido.</b> Atualizar o perfil de capacidade de processo definido sempre que o monitoramento identificar desvios significativos no alinhamento entre o perfil e os objetivos estratégicos, de forma que este alinhamento seja mantido. [resultado 6]	
	<b>PIM.4.BP.12 Buscar correspondência entre perfil e processo.</b> Buscar, por meio de ajustes no perfil e/ou ações de melhoria, a correspondência entre o perfil de capacidade de processo definido e o processo da unidade organizacional. [resultado 7]	
Produtos de trabalho	Entrada:	Saída:
	08-29 plano de melhoria [resultados 1, 2, 6]	
	15-02 metas de negócio [resultados 1, 2, 6]	
	15-03 declaração do valores chave	
	15-04 declaração da missão	
	15-13 relatório com resultado de avaliação	15-13 relatório com resultado de avaliação
	15-16 oportunidades de melhoria	15-16 oportunidades de melhoria
	16-01 repositório de resultados de avaliações	16-01 repositório de resultados de avaliações
	15-25 perfil de capacidade de processo [resultados: 4, 5, 6, 7]	15-25 perfil de capacidade de processo [resultados: 3, 5, 6, 7]

A Figura 54 ilustra, e a Tabela 17 descreve, a área de processo (ou processo, segundo a ISO/IEC 15504-5) de estabelecimento de perfil de capacidade de processo, segundo o formato utilizado na ISO/IEC 15504-5 [2005] e como um processo adicional à ISO/IEC 15504-5, no caso um quarto processo do grupo de melhoria de processo. Por isto a identificação de PIM.4. Esta área de processo é definida com sete resultados, doze práticas base e utiliza, produz ou atualiza oito produtos de trabalho. Destes oito produtos de trabalho, sete estão descritos na ISO/IEC 15504-5 e o oitavo, no caso o

produto de trabalho identificado como 15-25 e nomeado ‘perfil de capacidade de processo’, é definido nesse trabalho.

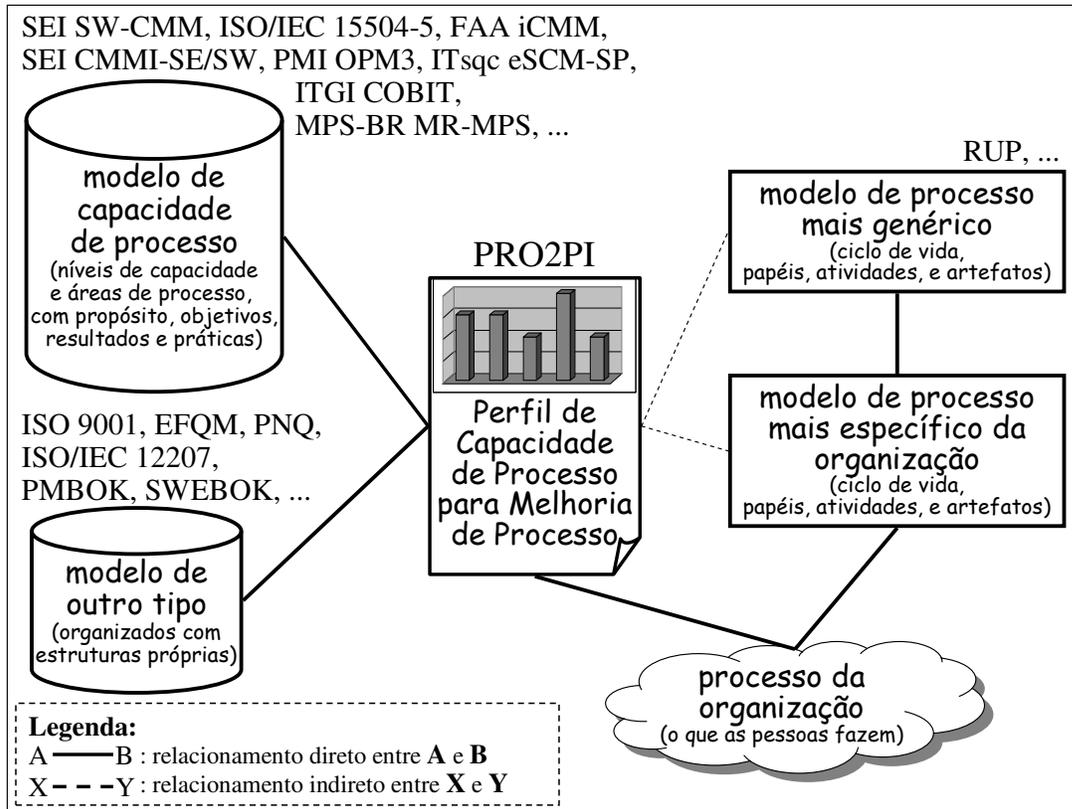


Figura 55 – Perfil no relacionamento de modelos com o processo

Os quinze modelos mais utilizados nesta tese, o seu agrupamento nos três tipos de modelos (modelo de capacidade de processo, outros tipos de modelos e modelo de processo genérico), e os seus relacionamentos com o processo de uma organização, todos já ilustrados na Figura 45, podem ser redesenhados, tendo um perfil de capacidade de processo como elemento central dos relacionamentos, conforme ilustrado na Figura 55.

Com a engenharia de processo PCDE, um perfil de capacidade de processo atua como elemento central no relacionamento entre modelos de capacidade de processo e outros tipos de modelos, com um relacionamento direto com o processo da organização. Esse processo pode estar sendo representado por um modelo do processo da organização, que por sua vez pode ser uma adaptação de um modelo de processo genérico. A Figura 55 ilustra esses relacionamentos, com os modelos SW-CMM, CMMI-SE/SW, ISO/IEC 15504-5, iCMM, eSCM-SP, MR-MPS, OPM3, COBIT, ISO/IEC 12207, ISO 9001, PMBOK, SWEBOK, EFQM, PNQ e RUP.

Para concluir esse desdobramento da engenharia PCDE, os conceitos básicos e seus relacionamentos descritos na Figura 4 do Capítulo 1 para a atual SPI/SPE, são revistos, para refletir a engenharia PCDE conforme ilustrado na Figura 56.

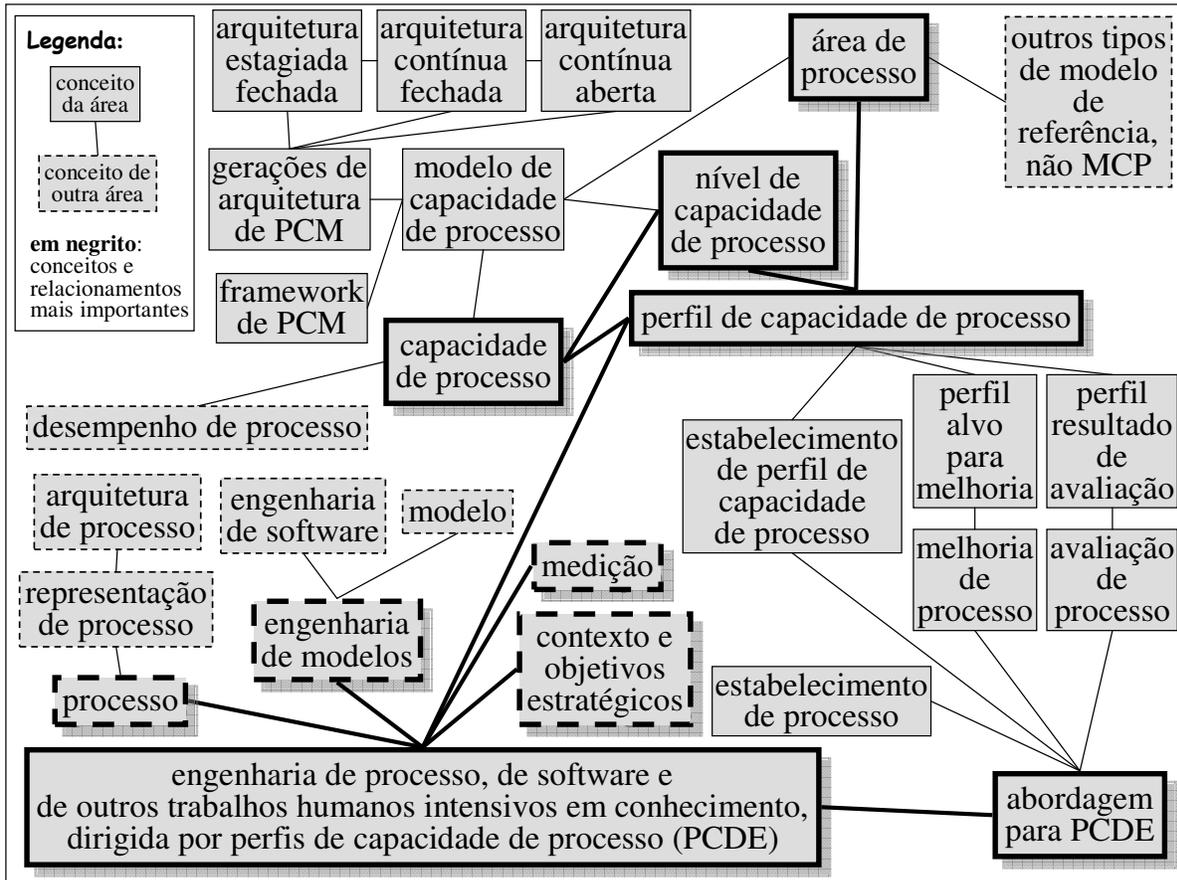


Figura 56 – Conceitos básicos para engenharia PCDE

A seguir os conceitos identificados na Figura 56 estão descritos pelos relacionamentos entre eles. Os conceitos são identificados em **negrito** e os relacionamentos como textos sublinhados.

A **engenharia de processo PCDE** orienta a definição e utilização de perfil de capacidade de processo. A **engenharia de processo PCDE** é uma engenharia de modelos para melhoria de processo alinhada ao contexto e objetivos estratégicos com apoio de medição, por meio de abordagem para PCDE. Um **processo** é documentado por representação de processo que é baseada em arquitetura de processo. Um **perfil de capacidade de processo** é fundamentado na capacidade de processo que é uma previsão do desempenho de processo.

Uma **abordagem para PCDE** é composta por atividades para estabelecimento de perfil de capacidade de processo, estabelecimento de processo, avaliação de processo e melhoria de



## 4.4 Síntese do capítulo

Esse capítulo apresenta uma reflexão sobre oportunidades de evolução da área de melhoria de processo de software para uma melhoria mais viável e mais adequada ao contexto e objetivos estratégicos das organizações. Esta reflexão é complementada com uma proposta para esta evolução, como uma {(Engenharia de processo) (de Software e de qualquer outro Trabalho Humano Intensivo em Conhecimento) Dirigida por (Perfis de Capacidade de Processo)} (PCDE). Esta proposta é sintetizada por uma definição de PCDE e uma indicação de seu relacionamento com outras engenharias. Esta reflexão e evolução estão fundamentadas na utilização da terceira geração de arquitetura de modelos de capacidade, descrita no capítulo anterior. Como desdobramento são apresentadas, para a engenharia PCDE, uma nova versão das áreas componentes, uma nova área de processo para a ISO/IEC 15504-5, e novas versões para os conceitos básicos e relacionamentos que caracterizam esta engenharia.

# Capítulo 5

## 5 Abordagem PRO2PI para engenharia de processo

O objetivo deste capítulo é apresentar uma abordagem exemplo para a engenharia de processo PCDE que oriente a experimentação, validação e evolução de PCDE por meio da abordagem exemplo. A abordagem exemplo é uma abordagem para uma melhoria de processo dirigida por perfis de capacidade de processo que é denominada de abordagem PRO2PI (Perfil de Capacidade de Processo para Melhoria de Processo, em inglês, *Process Capability Profile to Process Improvement*)<sup>10</sup>. O termo PRO2PI é também utilizado para designar o perfil de capacidade de processo definido e utilizado para dirigir a melhoria de processo segundo a abordagem PRO2PI.

Esse capítulo apresenta um conjunto de pressupostos e requisitos para a abordagem PRO2PI, apresenta uma visão geral da utilização de PCDE com PRO2PI, identifica quatro elementos que descrevem a abordagem e define cada um elemento desse conjunto. Esses elementos são propriedades de um PRO2PI, um modelo de PRO2PI, medições de PRO2PI e um ciclo de melhoria com PRO2PI.

### 5.1 Pressupostos, requisitos, apresentação e visão geral

Essa seção apresenta um conjunto de pressupostos, apresenta uma visão geral da utilização de PCDE com PRO2PI, identifica um conjunto de requisitos para a abordagem PRO2PI e identifica um conjunto de elementos para satisfazer esses requisitos.

Os pressupostos e requisitos para o desenvolvimento e entendimento da abordagem PRO2PI estão relacionados a seguir.

- a) A abordagem deve ser um exemplo da engenharia de processo dirigida por perfis de capacidade de processo (PCDE).
- b) Uma organização tende a utilizar elementos de mais de um modelo de capacidade de processo como referência para a melhoria, devido à diversidade de estratégias e características entre as

---

<sup>10</sup> PRO2PI: *PRO*cess capability *PRO*file to *PRO*cess *Improvement*, onde o 2 representa ao mesmo tempo a repetição de PRO e também o termo *to*. PRO2PI deve ser pronunciado em inglês, de forma semelhante a “pró tru pai”.

organizações e à diversidade de modelos com diferentes focos. Essa diversidade de modelos e de estratégias explicam a tendência atual de ser utilizado mais de um modelo como referência para a melhoria. A ISO/IEC 15504 antecipou esta tendência e descreve um framework para definição de modelos e métodos de avaliação.

- c) A diversidade de modelos com a especialização de cada modelo é uma característica esperada após o estabelecimento da área de melhoria de processo de software que foi baseada em um único modelo, com pouca flexibilidade.
- d) É necessário apoio metodológico para a utilização dessa diversidade e especificidade de modelos. Sempre que uma flexibilidade é oferecida, é necessário o desenvolvimento de conhecimento para utilização desta flexibilidade. Toda flexibilidade, para ser bem aproveitada, demanda o desenvolvimento de conhecimento para sua utilização com eficiência. Este apoio metodológico deve ser provido pela abordagem proposta.
- e) A abordagem deve ser desenvolvida com uma metodologia de pesquisa semelhante à indústria-como-laboratório. Por ser uma área recente, a abordagem deve ser desenvolvida por meio de interações constantes com a prática e portanto com uma abordagem semelhante à indústria-como-laboratório.

Uma visão geral da utilização da engenharia PCDE com a abordagem PRO2PI é apresentada a seguir com o apoio de seis diagramas esquemáticos seqüenciais e cumulativos. A Figura 58 apresenta o primeiro diagrama esquemático. Esse diagrama é referente à utilização corrente da melhoria de processo.

As ações de melhoria de processo são geralmente orientadas principalmente por um nível de maturidade de processo de um modelo estagiado, como, por exemplo, o modelo CMMI-SE/SW. Essas ações são geralmente chamadas de “implementação do nível <n> de maturidade”, onde <n> é por exemplo 2 ou 3. Essa implementação do nível de maturidade deve considerar também o contexto e objetivos estratégicos da organização e da unidade organizacional. Elementos de outros modelos, como, por exemplo, dos modelos da ISO/IEC 15504-5, iCMM, ISO 9001, PMBOK e RUP, também têm sido utilizados como referências adicionais. A realização das ações de melhoria de processo evoluem o processo da unidade organizacional. Essa evolução do processo causa novos resultados de desempenho organizacional. O processo evoluído pode ser examinado por uma avaliação de capacidade de processo que gera como resultado o quanto o processo atende a um determinado nível de maturidade.

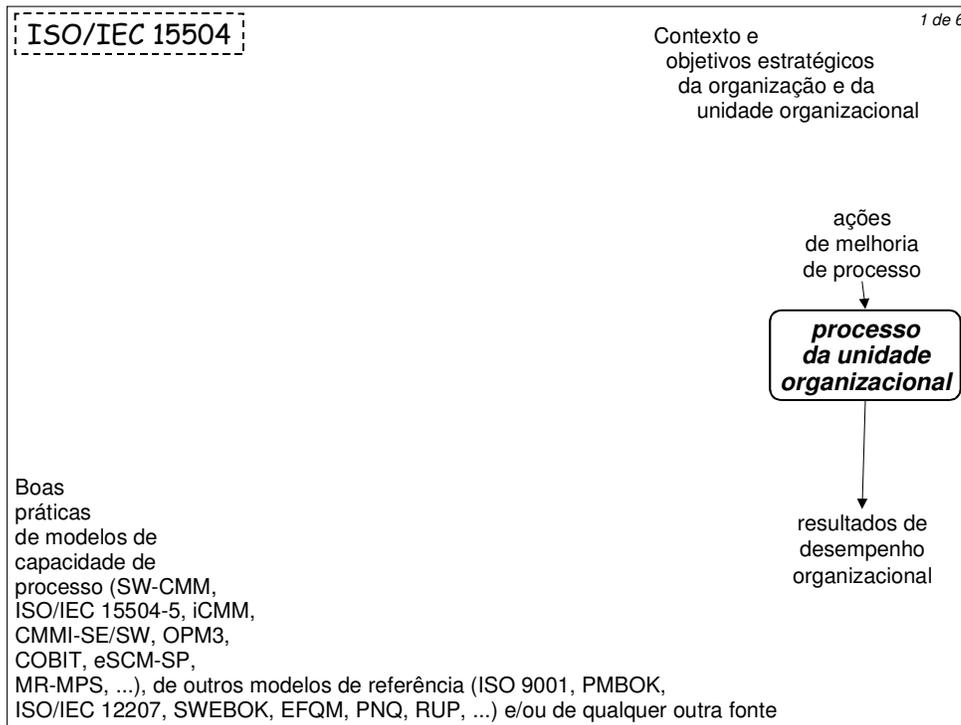


Figura 58 – Diagrama da utilização corrente da melhoria de processo

O processo também pode ser evoluído por ações orientadas por um perfil de capacidade de processo definido de um modelo contínuo, como, por exemplo, os modelos CMMI SE/SW (na representação contínua), ISO/IEC 15504-5 e iCMM. Nesse caso o processo evoluído pode ser examinado também por uma avaliação de capacidade de processo, neste caso em relação ao perfil definido. Ações de melhoria orientadas pelo modelo ISO 9001 também podem ser utilizadas e neste caso o exame do processo evoluído é feito por uma avaliação de certificação de processo em relação à ISO 9001. Também podem ser utilizados elementos de modelos como o PMBOK e RUP para orientar as ações de melhoria mas nesses casos, não são feitas avaliações de processo para examinar o processo evoluído. Concluindo, também podem ser utilizadas orientações de qualquer outro modelo ou mesmo serem realizadas ações de melhoria de processo sem nenhuma orientação definida. Essas ações de melhoria e avaliações de processo podem ser feitas em qualquer combinação das situações descritas nesse parágrafo. As maiores limitações estão relacionadas à não existência de uma visão sistêmica desta combinação. Não existe uma representação, sob o aspecto de capacidade de processo, do processo da unidade organizacional. Apenas representações de elementos do processo. Partes dessas representações são redundantes entre si e outras partes podem ser até mesmo inconsistentes entre si.

Uma visão geral da utilização da abordagem PRO2PI é apresentada a seguir por meio de quatro funções. O diagrama com a função de definição, ou atualização, de PRO2PI é apresentado na Figura 59.

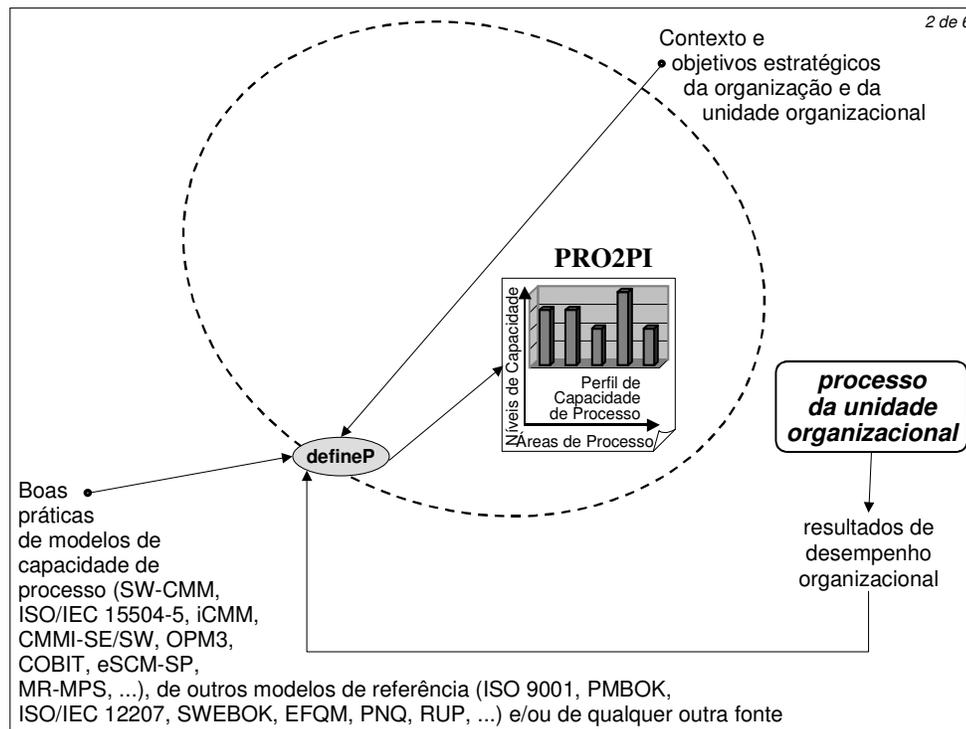


Figura 59 – Diagrama da definição de PRO2PI

Uma organização seleciona elementos de um ou mais modelos de referências e define um perfil de capacidade de processo que representa os elementos selecionados desses modelos e de qualquer ou fonte. Um ciclo de melhoria de processo é realizado buscando evoluir os processos para atender a esse perfil. Esse perfil é alinhado com o contexto e objetivos estratégicos da organização e da unidade organizacional. O perfil pode conter boas práticas de referência selecionadas dos modelos mais genéricos existentes e de outras fontes, de forma a representar orientações relevantes à organização. Esse perfil pode ser alterado a qualquer momento em função de novas percepções, alterações do contexto ou dos objetivos estratégicos e dos resultados da utilização da versão corrente do perfil. Esse uso é representado pela função “defineP” (define, ou atualiza, perfil de capacidade de processo) da Figura 59.

O diagrama com a função de utilização de PRO2PI é apresentado Figura 60 como um acréscimo em relação à Figura 59.

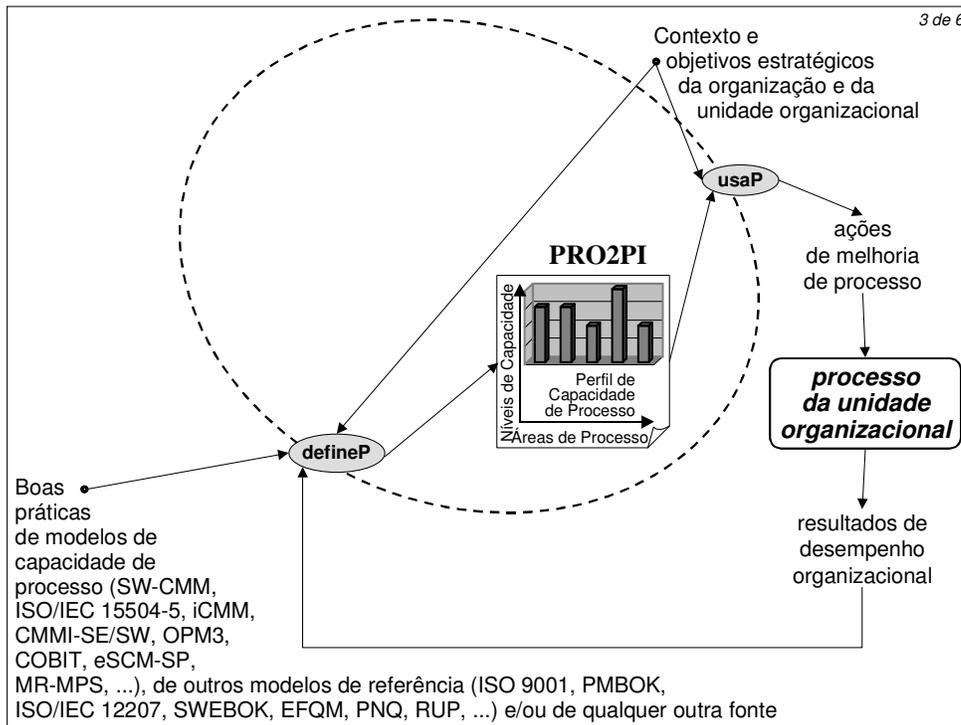


Figura 60 - Diagrama da utilização (e definição) de PRO2PI

O uso desse perfil para orientar as ações de melhoria é representado pela função “usaP” (usa perfil de capacidade de processo) da Figura 59. Neste caso as ações de melhoria devem ser suficientes para que o processo resultante atenda a todos os requisitos representados no perfil de capacidade de processo e apenas a estes requisitos.

O diagrama com a função de avaliação de capacidade de processo em relação a um PRO2PI é apresentado na Figura 61 como um acréscimo em relação à Figura 60.

O processo da unidade organizacional pode ser examinado com uma avaliação de capacidade de processo em relação ao perfil de capacidade de processo. Esse exame é representado na Figura 61 pela função “avaliaPr” (avalia capacidade de processo em relação a um PRO2PI). Os resultados de capacidade de processo gerados por essa avaliação podem ser utilizados como referências adicionais para a definição, ou alteração, do perfil de capacidade de processo.

O diagrama com a função de definição de modelos de capacidade de processo mais específicos com a abordagem PRO2PI é apresentado na Figura 63.

A partir do contexto de negócio de um segmento, como, por exemplo, o segmento bancário, ou de um domínio, como, por exemplo, engenharia de requisitos, pode ser definido um modelo mais específico. Esse modelo pode ser composto por áreas de processo ou perfis de capacidade de processo mais específicos para um segmento ou domínio.

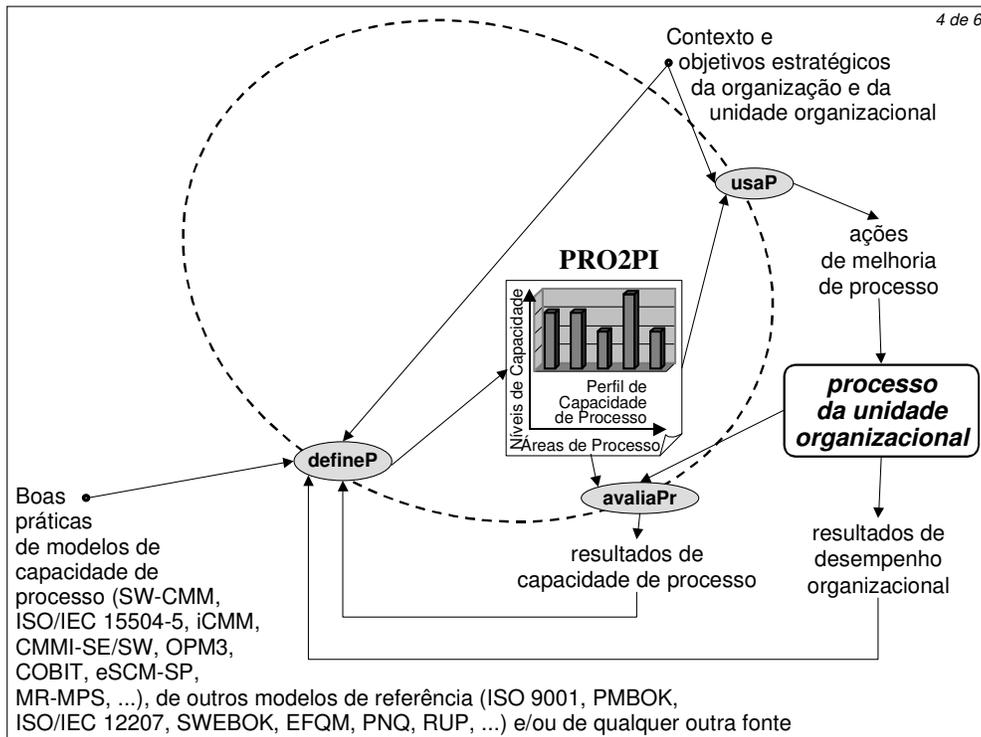


Figura 61 - Diagrama da avaliação de processo (e definição e utilização de PRO2PI)

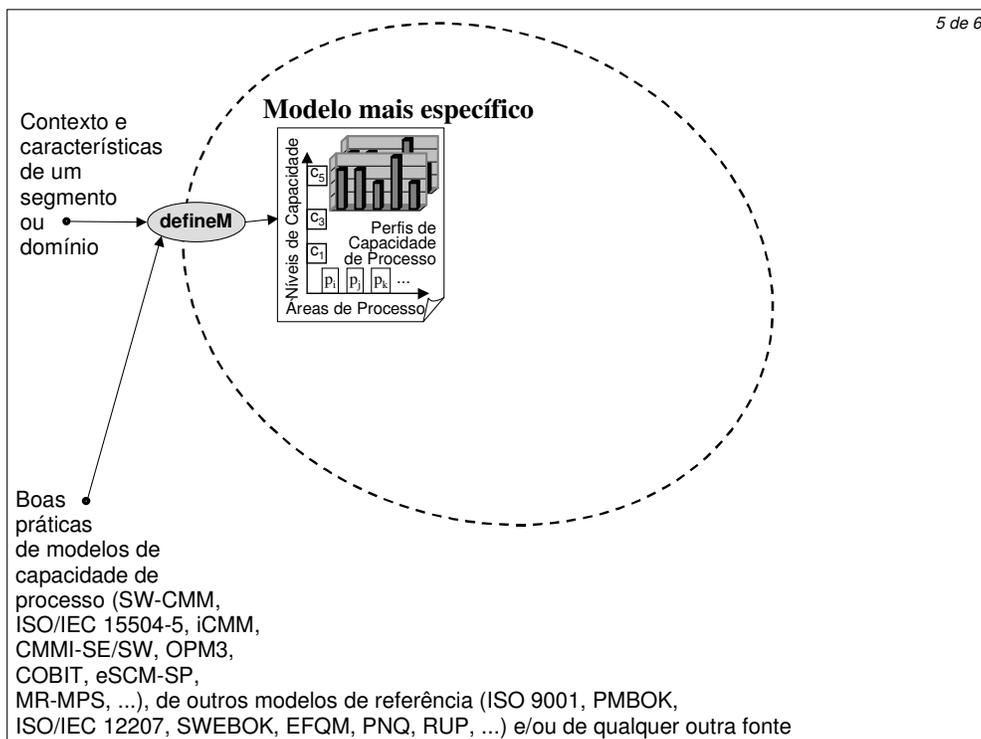


Figura 62 – Diagrama da definição de modelos mais específicos

Esse modelo mais específico pode conter boas práticas selecionadas dos modelos de capacidade de processo, modelos de outros tipos ou de qualquer outra fonte de forma a representar orientações relevantes no segmento ou domínio. Esse uso é representado pela função “defineM” (define, ou atualiza, modelo mais específico de capacidade de processo) da Figura 63.

Um modelo mais específico pode ser que qualquer uma das três gerações de arquiteturas. Caso seja um modelo estagiado fixo, ele será composto por uma hierarquia de perfis de capacidade de processo, como, por exemplo, a hierarquia do modelo PMMM para o domínio de gerência de projeto e do modelo KMMM para o domínio de gerência de conhecimento. Caso seja um modelo contínuo fechado, ele será composto por um conjunto de áreas de processo, como, por exemplo, as áreas de processo do modelo Automotive SPICE para o segmento de fornecedores de software para automóveis e as do modelo OOSPICE para o domínio de desenvolvimento de software orientado a componentes. Esses modelos podem ter sugestões de hierarquia de perfis de capacidade de processo. Caso seja um modelo contínuo aberto, ele será composto de especializações dos níveis de capacidade de processo.

A Figura 63 representa um diagrama esquemático completo da utilização da abordagem PRO2PI para desenvolvimento de modelos mais específicos e para realização de ações de melhoria dirigida por perfil de capacidade de processo. Essa utilização é realizada por meio das quatro funções: defineM, defineP, usaP e avaliaPr. Não existe uma ordem pré-definida de utilização destas funções. Elas podem ser utilizadas várias vezes cada uma independente da execução das outras funções.

Na abordagem PRO2PI existe uma busca contínua em direção ao alinhamento do perfil de capacidade de processo como uma representação, segundo o aspecto de capacidade de processo, do processo da unidade organizacional. Essa relação está descrita no capítulo, no qual está ilustrada na Figura 51.

Conforme representado na Figura 63 uma melhoria de processo com a abordagem PRO2PI é realizada com a definição e utilização de um PRO2PI que pode conter boas práticas selecionadas dos modelos de capacidade de processo, modelos de outros tipos, modelos mais específicos de um segmento ou domínio ou de qualquer outra fonte.

Essa utilização da abordagem PRO2PI pode ser considerada como análoga à utilização da engenharia de requisitos no desenvolvimento de software, que podemos denominar de desenvolvimento de software dirigido por requisitos. A Figura 64 apresenta um diagrama da utilização da engenharia de requisitos para desenvolvimento de software de forma análoga ao diagrama de utilização de PRO2PI descrito na Figura 63.

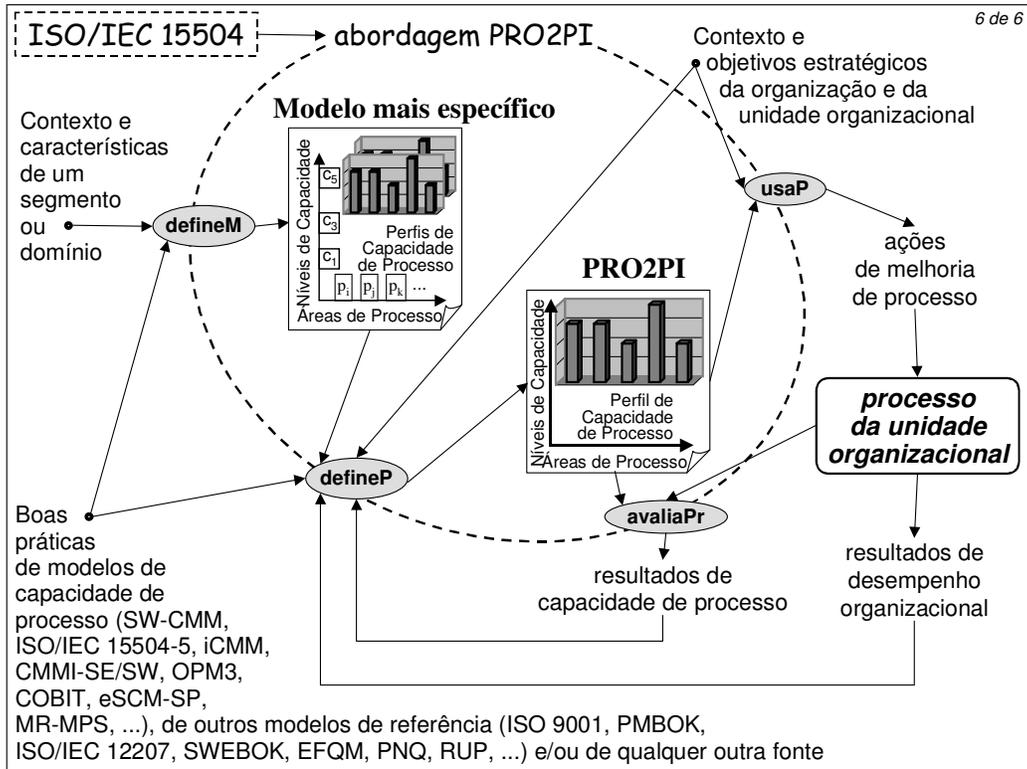


Figura 63 – Diagrama da abordagem PRO2PI para modelos e melhoria de processo

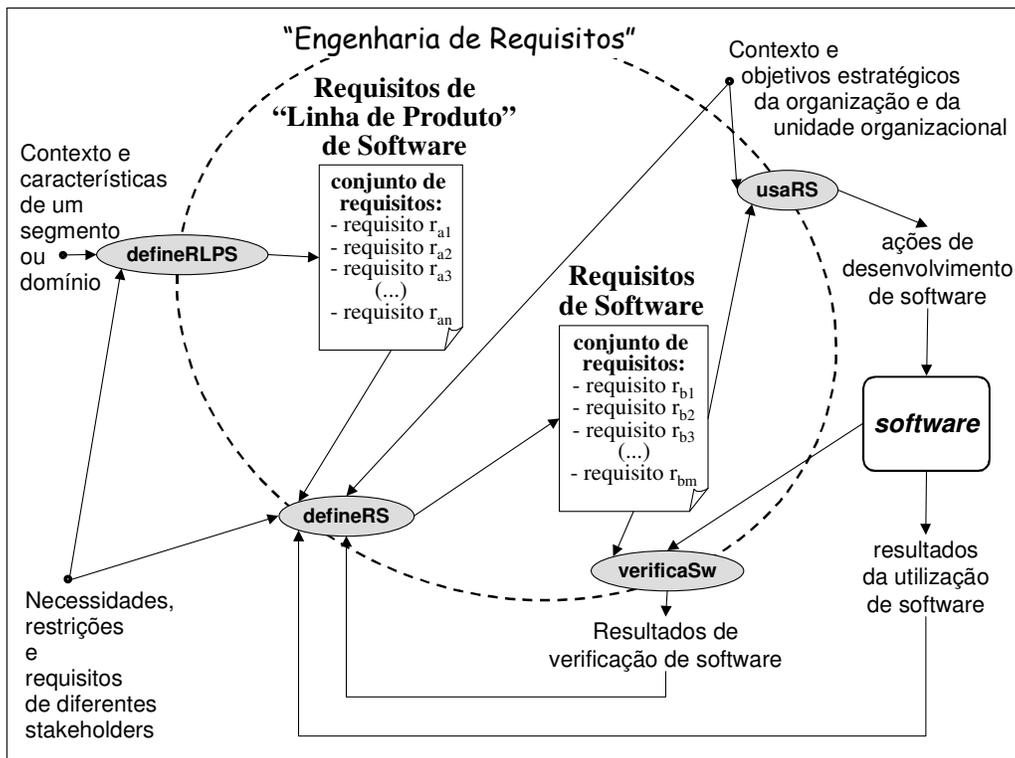


Figura 64 – Diagrama, análogo ao de PRO2PI, do desenvolvimento de software

Uma forma hipotética de desenvolvimento de software seria fazer esse desenvolvimento baseado em conjuntos de necessidades, restrições e requisitos de diferentes stakeholders, sem sistematizar esses conjuntos em um único conjunto de requisitos. Essa forma de desenvolvimento de software seria análoga à forma de melhoria de processo representada na Figura 58. Essa forma é hipotética porque não é utilizada, ou não deveria ser utilizada, em praticamente nenhum desenvolvimento de software.

O diagrama da Figura 64 representa uma forma de desenvolvimento de software que é mais utilizada atualmente, para a qual os conjuntos de necessidades, restrições e requisitos dos diferentes *stakeholders* são analisados e representados como um sistema em um único conjunto de requisitos para o software. Desta forma o software tem tudo que os requisitos requisitam e somente o que os requisitos requisitam. Nesta forma de desenvolvimento, a função “defineR” (define requisitos) seria utilizada para definir, ou atualizar, o conjunto de requisitos, a função “usaRS” (usa requisitos de software) seria utilizada para orientar as ações de implementação do software de forma que isto seja dirigida pelos requisitos e a função “testaSv” (testa software) seria utilizada para buscar problemas no software e com isto sugerir o quanto o software desenvolvido atende aos requisitos.

A função “defineRLPS” (define requisitos de “linha de produto” de software) representa o desenvolvimento de requisitos para uma “linha de produto” de software, de tal forma que esses requisitos sejam um sistema, sendo que os requisitos foram baseados em diferentes conjuntos de requisitos, de diferentes *stakeholders*. Esta forma de desenvolvimento de requisitos é análoga à forma de desenvolvimento de modelos mais específicos representada na Figura 63.

O diagrama da Figura 64 representa a integração da forma de desenvolvimento de requisitos para um framework de produtos de software e da forma de desenvolvimento de software dirigida a requisitos de software, de tal forma que esses requisitos sejam um sistema, sendo que os requisitos foram baseados em diferentes conjuntos de requisitos, de diferentes stakeholders, e em requisitos de um framework de software. Esta forma de desenvolvimento de requisitos é análoga à forma de desenvolvimento de modelos e realização de melhoria de processo representada na Figura 63.

A abordagem PRO2PI pode ser descrito por meio de quatro componentes:

- Um conjunto de oito propriedades que um perfil de capacidade de processo deve ter, em um grau suficiente, para ser um PRO2PI;
- Um modelo que permite a representação de uma abstração segundo o aspecto capacidade de processo do processo atual ou do processo alvo de uma organização;

- Um conjunto inicial de medições relacionadas a PRO2PI, incluindo uma medição da complexidade de um PRO2PI e da viabilidade de um PRO2PI; e
- Um ciclo de melhoria com PRO2PI.

## 5.2 PRO2PI-PROP: Propriedades de PRO2PI

Para ser útil e efetivo como orientação para a melhoria de uma organização, um PCP deve ter, em um grau suficiente, pelo menos as propriedades de ser relevante, oportuno, viável, sistêmico, representativo<sup>11</sup>, rastreável, específico e dinâmico. Este PCP pode então ser chamado de PRO2PI. Dessa forma PRO2PI é um PCP que tem, em um grau suficiente, pelo menos as propriedades de ser relevante, oportuno, viável, representativo e específico para uma determinada unidade organizacional, em um determinado momento e com uma determinada previsão de investimento em ações de melhoria, ser sistêmico e dinâmico, e ser rastreável a modelos relevantes, conforme ilustrado na Figura 65.

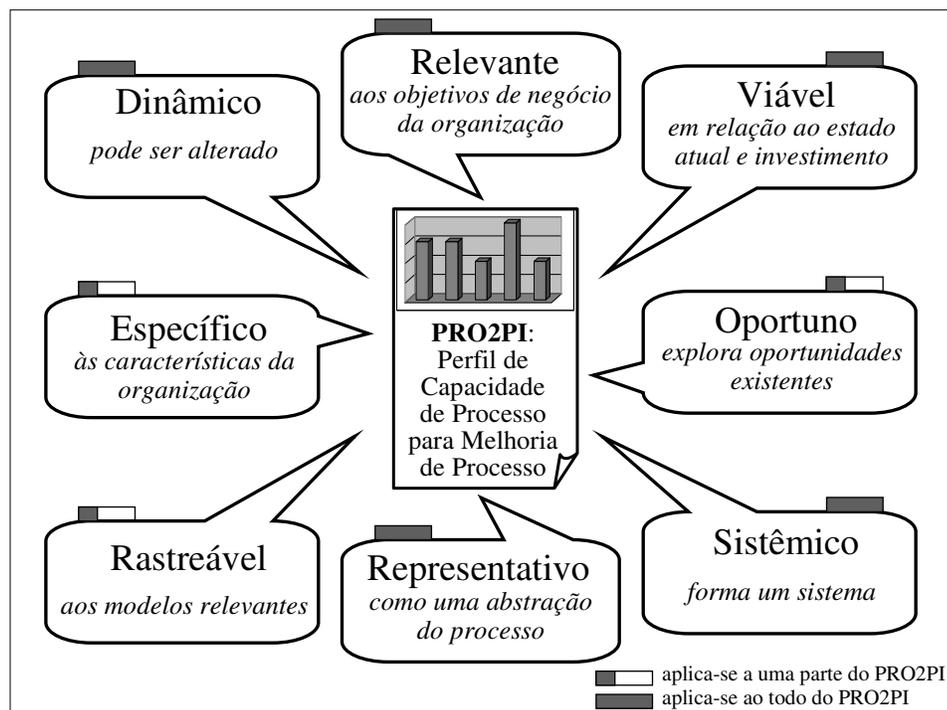


Figura 65 – Propriedades de PRO2PI

<sup>11</sup> Essa propriedade era nomeada de “abstração”, com o mesmo significado, em versões anteriores do trabalho [por exemplo Salviano et al. 2004 e Salviano e Jino 2004]

Conforme ilustrado na Figura 65, cinco propriedades (relevante, viável, sistêmico, representativo e dinâmico) aplicam-se ao perfil como um todo. Por exemplo, todo o perfil deve ser relevante aos objetivos estratégicos da organização. As outras três propriedades (oportuno, rastreável e específico) podem ser aplicadas apenas a uma parte do perfil ou mesmo não ser aplicada a determinado perfil. Por exemplo, caso apropriado, uma parte do perfil pode aproveitar alguma oportunidade para a melhoria.

A seguir cada uma das oito propriedades são descritas e exemplificadas.

### **Relevante**

Um PCP deve representar um estado do processo da organização que esteja alinhado com o contexto de negócio da organização, e com isto represente melhorias importantes para a organização. Estas melhorias são geralmente descritas em termos de qualidade, prazos de desenvolvimento, custos e satisfação dos clientes e trabalhadores. Várias abordagens podem ser utilizadas para apoiar a definição de PCPs relevantes, entre elas: BSC (*Balanced Scorecard Framework*) [Kaplan e Norton 1996] e SFO (*Strategy-Focused Organization*) [Kaplan e Norton 2001] e Abordagem Baseada em Problemas e Metas [Potter e Sakry 2002].

Por exemplo, uma organização que precise garantir o desenvolvimento de sistemas de software que sejam entregues ao cliente sem erros importantes, pode incluir no PCP a área de processo de teste no nível 4 de capacidade, juntamente com as áreas de processo de desenvolvimento no nível 2 e a área de processo de requisitos no nível 3.

A relevância de uma área de processo e um nível de capacidade deve também considerar fatores contextuais, técnicos, legados e oportunidades. Fatores contextuais são aqueles fatores sobre os quais a organização tem pouco ou nenhum controle. Por exemplo, requisitos impostos por agências reguladoras do governo. Fatores técnicos de seleção são aqueles relacionados à adequação do modelo dada a natureza específica da organização e do contexto de uso. Por exemplo, se os processos do modelo são consistentes com as necessidades da organização. Fatores legados são aqueles relacionados ao histórico de melhoria de processo da organização. Por exemplo, experiências anteriores com sucesso, utilizando o modelo da 12207 [ISO/IEC 15504-5 2004].

### **Oportuno**

Um PCP deve considerar, e se apropriado incorporar, oportunidades disponíveis. Por exemplo, mesmo que a área de teste não esteja entre as mais prioritárias, mas existam recursos disponíveis a custo baixo, pode ser interessante incluir devido à oportunidade.

### **Viável**

Um PCP deve representar um estado de processo que requeira uma quantidade de esforço e de recursos para seu atendimento, a partir do estado atual, que seja viável de ser disponibilizado. Para isto uma análise deve ser feita. Um aspecto importante é a utilização de uma medição para esta quantidade. Esta medição pode ser baseada na unidade de melhoria definida em PRO2PI-MEAS.

### **Sistêmico**

Um PCP deve representar um estado do processo que seja um sistema. Desta forma o processo representado por esse estado funcionará na organização, gerando os resultados necessários. Como um sistema, esse estado será ao mesmo tempo um situação auto sustentável e uma etapa para a melhoria. Para tanto, as seguintes características devem existir:

- Como cada área de processo estará em um nível de capacidade, as características do nível já representam um sistema para o processo,
- Como existem áreas de processo que podem suportar o atendimento de elementos de um nível de capacidade para uma área de processo, existem duas opções: estas áreas de processo podem ser incluídas no PCP, em um nível de capacidade adequado, ou elementos destas áreas podem ser utilizadas para implementação da área de processo. A primeira opção deve ser utilizada quando for apropriado um reforço destas áreas e a segunda quando não for necessário esse reforço.
- Os níveis de capacidade entre áreas de processo devem ser compatíveis entre si, sem grandes diferenças que inviabilizem o funcionamento do conjunto.

É muito importante considerar o relacionamento entre as áreas de processo e entre os níveis de capacidade de processo e as áreas de processo para definir um PCP sistêmico. Para isto devem ser considerados os relacionamentos já descritos nos modelos e em artigos técnicos. Os modelos do CMMI por exemplo, identificam para cada área de processo, outras áreas de processos relacionadas e como é esse relacionamento. Os modelos do CMMI e da ISO/IEC 15504 agrupam áreas de processo em categorias e/ou grupos, o que indica algum tipo de relacionamento entre áreas da mesma categoria e/ou mesmo grupo. Trabalhos desenvolvidos por Mäkinen e Varkoi [2003] e Adamonis et al. [2005], por exemplo, também identificam relacionamentos. Bons exemplos de PCPs sistêmicos são aqueles que representam os níveis de maturidade dos modelos estagiados do SW-CMM e do CMMI.

### **Representativo**

Um PCP deve representar uma abstração do processo, segundo o aspecto de capacidade de processo. Esta abstração deve ser completa, ou seja, representar todo o processo, nos termos e elementos de PCP. Todas as características representadas no PCP têm que estar sendo realizadas no processo e o processo realiza apenas o que está representado no PCP.

**Rastreável**

Como os modelos genéricos mais utilizados são consolidações de melhores práticas utilizadas por várias organizações, pode ser importante manter uma rastreabilidade do PCP com elementos relevantes, como, por exemplo, áreas de processo e níveis de maturidade, dos modelos relevantes para a organização. Mapeamento entre esses modelos, que estão disponíveis, podem ser também utilizados para manter esse rastreamento com múltiplos modelos.

Desta forma, caso seja necessário demonstrar satisfação de parte do estado do processo com esses modelos, isto poderá ser feito com esta rastreabilidade. É importante que esta rastreabilidade seja bi-direcional, ou seja, do PCP para elementos dos modelos e dos elementos do modelo para o PCP.

**Específico**

Como é importante manter um relacionamento um para um entre um PCP e o estado do processo, e cada organização tem sua especificidade, é fundamental que o PCP tenha elementos específicos para a organização. Por exemplo, caso uma organização tenha como diferencial de negócio o conhecimento de um determinado domínio, incluindo o conhecimento de sistemas legados desse domínio, uma área de processo “manter conhecimento do domínio” pode ser definida e incluída no PCP.

**Dinâmico**

Como os vários fatores que influenciam o que deveria ser o estado do processo podem mudar, é importante que os PCPs possam ser ajustados. Também como o conhecimento sobre esses mesmos fatores tende a aumentar, os PCPs podem ser ajustados para refletir esse melhor conhecimento. Portanto eles devem ser dinâmicos.

As seguintes operações podem ser realizadas como ajuste de um PCP, considerando PCPr como o PCP resultante da operação, PCPa1 e PCPa2 como os PCPs fontes da operação e CLdelta como um incremento em termos de níveis de capacidade:

- Incremento vertical:  $PCPr = PCPa1 + CLdelta$ ;
- Decremento vertical:  $PCPr = PCPa1 - CLdelta$ ;
- Composição horizontal:  $PCPr = PCPa1 + PCPa2$ ; e
- Decomposição horizontal:  $PCPr = PCPa1 - PCPa2$ .

Essa propriedade pode ser exemplificada com a relação entre o estabelecimento de um PRO2PI e um nível de maturidade do SW-CMM ou da representação estagiada do CMMI-SE/SW. Do ponto de vista de PRO2PI, um nível de maturidade é um exemplo de um PCP sistêmico, fixo e genérico (ou específico para uma unidade organizacional típica desenvolvedora de projetos baseada em editais).

Uma unidade organizacional realizando uma melhoria de processo buscando um nível de maturidade pode utilizar PRO2PI para:

- a) expandir o nível de maturidade, incluindo novos requisitos, representados por áreas de processos, que orientem implementações adicionais julgadas apropriadas para a melhoria de processo, com a operação de composição horizontal, onde PCPa1 é o nível de maturidade e PCPa2 é formado pelas áreas de processo a serem incluídas;
- b) expandir o nível de maturidade, semelhantemente ao caso anterior, mas com o incremento de um ou mais níveis de capacidade para as áreas de processo, com a operação de incremento vertical, onde PCPa1 é o nível de maturidade e CLdelta o incremento de nível de capacidade;
- c) reduzir o nível de maturidade, excluindo requisitos, representados por áreas de processos, que orientem implementações em menores quantidades julgadas mais viáveis para a melhoria de processo, com a operação de decomposição horizontal, onde PCPa1 é o nível de maturidade e PCPa2 é formado pelas áreas de processo a serem excluídas;
- d) reduzir o nível de maturidade, semelhante ao caso anterior, mas com o decremento de um ou mais níveis de capacidade para as áreas de processo, com a operação de decremento vertical, onde PCPa1 é o nível de maturidade e CLdelta o decremento de nível de capacidade;
- e) combinações das opções anteriores.

### **5.3 PRO2PI-MODEL: Modelo de PRO2PI**

Esta seção apresenta um modelo de PRO2PI, denominado de PRO2PI-MODEL (*PRO2PI Model*). O modelo PRO2PI-MODEL atende os seguintes requisitos:

- unifica os elementos e estruturas dos modelos de capacidade de processo considerados mais relevantes, especialmente dos modelos iCMM v2.0, CMMI-SE/SW v1.1, ISO/IEC 15504-5:2006 e MR-MPS v1.0, e com isto permite a representação de praticamente qualquer elemento desses modelos em um PRO2PI;
- busca representar em um PRO2PI os elementos de outros modelos de capacidade de processo, incluindo PMI OPM3, ITsqc eSCM-SP, ITGI COBIT, SM<sup>MM</sup>, KMMM, PMMM, UMM, TMM, OOSPICE, SPICE4SPACE, Automotive SPICE, 15504MPE e S9K;
- permite a representação em um PRO2PI também de elementos selecionados de outros modelos de referência, que não sejam modelos de capacidade de processo, especialmente dos modelos

PMBOK e ISO 9001, e incluindo ISO/IEC 12207 Amd2, IEEE SWEBOK 2004, EFQM, FPNQ PNQ e RUP.

- é especificado em uma referência conceitual adequada na linha de MDE; e
- minimiza a quantidade de tipos de elementos.

Utilizando a notação BNF (*Backus Naur Form*) [Estier 1998], um perfil de capacidade de processo pode ser representado segundo a seguinte notação:

```
<PCP> ::= “[“ <PCP_Element> { “,” <PCP_Element> } “]”
<PCP_Element> ::= <ProcAreaId> “:” CL <CapLevelId> |
                “{“ <ProcAreaList> “}” “:” CL <CapLevelId>
<ProcAreaList> ::= <ProcAreaId> { “,” <ProcAreaId> }
```

Com <ProcAreaId> sendo uma identificação de uma área de processo e <CapLevelId> a identificação de um nível de capacidade de processo.

O perfil de capacidade de processo indicado pelo nível 2 de maturidade do modelo CMMI-SE/SW, por exemplo, pode ser representado como:

```
[ { REQM, PP, PMC, CM, PPQA, MA, SAM } : CL 2 ]
```

O perfil de capacidade de processo ilustrado Figura 18 pode ser representado como:

```
[ { GerProj, ElicReq } : CL3, ConsSw : CL 2, TesteSw : CL 4, SupCli : CL 2 ]
```

Uma hierarquia de PCPs pode ser representada como uma seqüência com cada elemento em uma de duas possíveis formas. Em uma forma o elemento tem uma identificação relativa do PCP e o próprio PCP, de tal forma que um PCPa2 hierarquicamente superior a outro PCPa1 contenha o PCPa1. Em outra forma o elemento tem uma identificação relativa do PCP e o termo “Delta” seguido do acréscimo em relação ao PCP imediatamente inferior hierarquicamente. Esse acréscimo é também representado com a mesma notação de um PCP. Uma hierarquia pode então ser representada conforme a seguinte notação:

```
<H-PCP> ::= “<<“ <H-PCP_Element> { “,” <H-PCP_Element> } “>>”
<PCP_Element> ::= “<“ <PCP-Id> “>” “:” <PCP> |
                “<“ <PCP-Id> “>” “:” “Delta” <PCP>
```

Com <PCP-Id> sendo uma identificação de um PCP

Os quatro níveis de maturidade do modelo CMMI-SE/SW, por exemplo, podem ser representados como a seguinte hierarquia com quatro PCPs:

```

<<
<Nível 5> : Delta [ { CAR, OID } : CL 3 ] ;
<Nível 4> : Delta [ { OPP, QPM } : CL 3 ] ;
<Nível 3> : [ { REQM, PP, PMC, CM, PPQA, MA, SAM,
              RD, TS, PI, VER, VAL, DAR, IPM, RSKM, OPF, OPD , OT } : CL 3 ] ;
<Nível 2> : [ { REQM, PP, PMC, CM, PPQA, MA, SAM } : CL 2 ]
>>

```

A Figura 66 ilustra a caracterização de PRO2PI-MODEL como um modelo, que representa um processo sob o aspecto de capacidade de processo, e outros três modelos, respectivamente ETVX, UML e Java, que representam outros aspectos de processo e software.

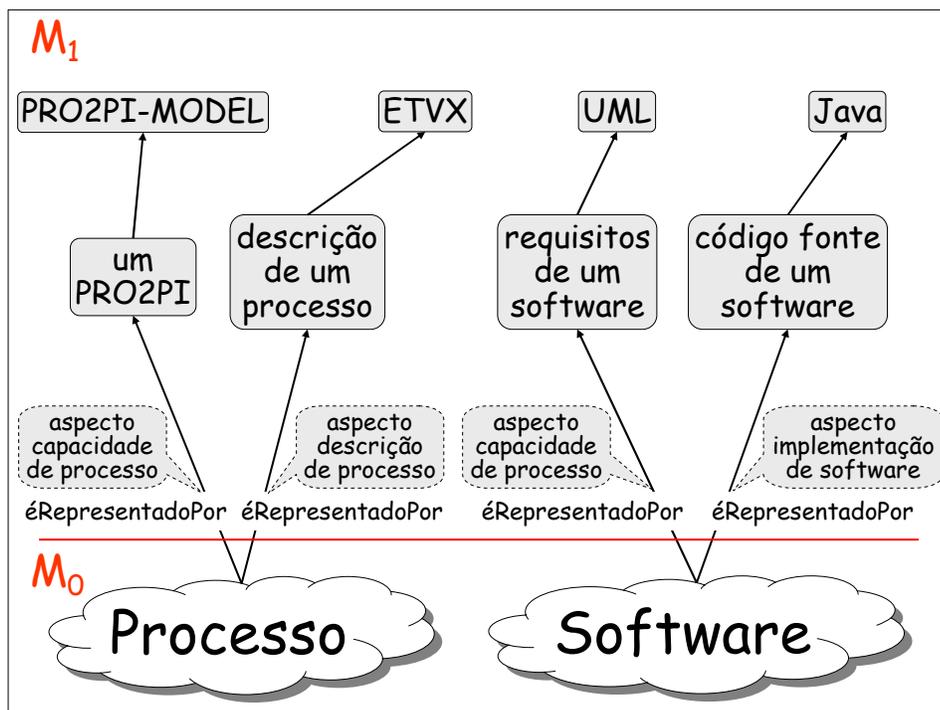


Figura 66 – PRO2PI-MODEL e outros modelos de processo ou software

Um PRO2PI é uma abstração, segundo o aspecto de capacidade de processo, de um processo e pode ser utilizado como referência para a melhoria do processo atual para esse estado em uma organização. Como cada PRO2PI é um modelo de capacidade de processo de um processo, um modelo de PRO2PI, ou mais precisamente, um modelo da linguagem de modelagem de PRO2PI, pode ser

considerado como um metamodelo de um processo. Segundo a hierarquia de quatro níveis de modelagem adotado pela OMG, temos:

- M0: exemplos de processo
- M1: Perfis de Capacidade de Processo como modelos de referência de processos,
- M2: Modelos de Perfis de Capacidade de Processo como metamodelos de processos
- M3: Modelo de metamodelos de processos como modelo dos modelos de Perfis de Capacidade de Processo.

A Figura 67 ilustra PRO2PI-MODEL como um metamodelo M2 da hierarquia da OMG com hierarquias análogas para processo e software.

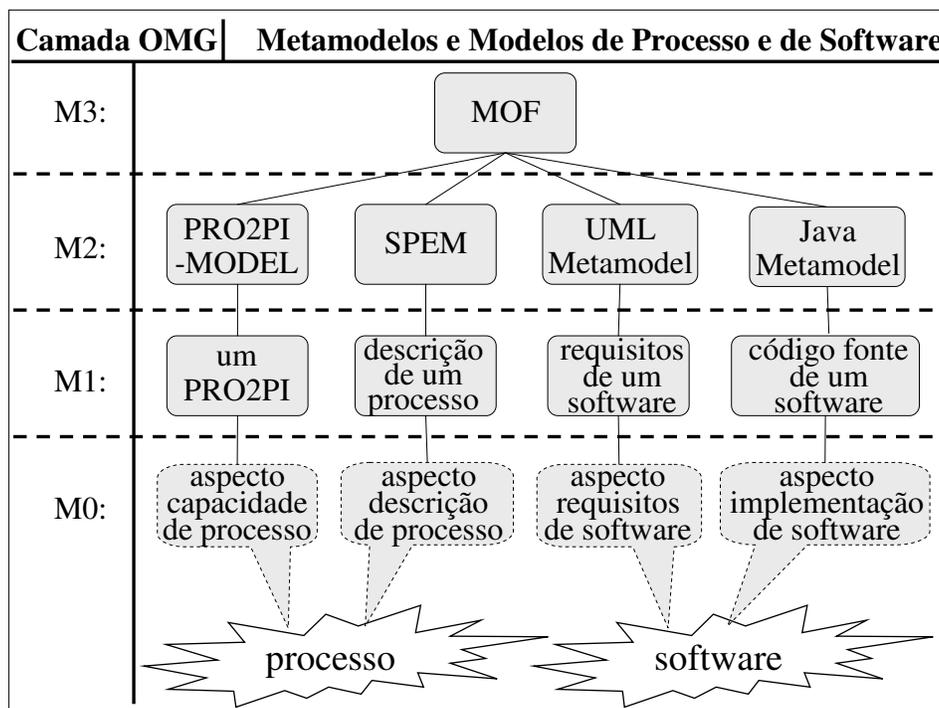


Figura 67 - PRO2PI-MODEL e outros metamodelos e modelos de processo e software

PRO2PI-MODEL é um modelo de um PRO2PI, e com isto um metamodelo de processo segundo o aspecto de capacidade de processo. De forma análoga, SPeM [OMG 2002b] é um modelo de uma descrição de processo, e com isto um metamodelo de processo segundo o aspecto descrição de processo. SPeM está representado conforme o metamodelo MOF [OMG 2002a] e PRO2PI-MODEL também pode ser representado conforme o metamodelo MOF.

Esse par de hierarquias de PRO2PI e descrição de processo é por sua vez análogo ao par de hierarquias de requisitos de software e implementação de software. Os requisitos de um software formam um modelo do software segundo o aspecto requisitos de software e o código fonte de um

software é um modelo do mesmo software segundo o aspecto implementação do software. O metamodelo UML [OMG 2003a] é um exemplo de metamodelo para requisitos de software e o Java Metamodel [OMG 2002c] é um exemplo de metamodelo para implementação de software. Os dois metamodelos são representados conforme MOF.

A especificação de PRO2PI-MODEL pode ser inspirada na especificação do Software Process Engineering Metamodel (SPEM) do OMG [OMG 2002b]. PRO2PI-MODEL complementa SPEM. Um modelo em SPEM representa a descrição de um processo de desenvolvimento de software que, por sua vez, representa a descrição de como o processo deve ser executado. Esta descrição é feita em termos de papéis, atividades e produtos de trabalho, organizados em processos que, por sua vez, são organizados em fases em uma dimensão e disciplinas em outra dimensão. Ou seja, a abstração de SPEM do processo é focada na descrição das atividades a serem executadas.

Um modelo em PRO2PI-MODEL representa a descrição dos requisitos de capacidade de um processo. Esse processo pode ser, por exemplo, o processo de desenvolvimento de software. Um modelo em SPEM representa a descrição de um processo. Esse processo pode ser, por exemplo, o mesmo processo de desenvolvimento de software.

A limitação do SPEM é pelo fato do foco na descrição do processo não ser suficiente para orientar a gerência, descrição e execução de aspectos do processo que sejam sistêmicos, cumulativos e viáveis de serem realizados por organizações. Esta orientação pode ser baseada em um conjunto de associações de áreas de processo e níveis de capacidade. SPEM é um metamodelo de processos de desenvolvimento de software, que é parte do processo, com modelos baseados na abstração da descrição do processo, e PRO2PI é um metamodelo de processos, baseado na abstração de perfis de capacidade de processo.

#### **Diagrama de classes de PRO2PI-MODEL:**

A seguir é apresentada a especificação das classes do modelo PRO2PI-MODEL na notação OMG UML (*Unified Modeling Language*) versão 1.5 [OMG 2003a].

Esse modelo é utilizado para descrever Perfis de Capacidade de Processo (PCP), compostos por associações de Áreas de Processo e Níveis de Capacidade de Processo. Na visão de PRO2PI, um PCP representa os requisitos e orientações para o processo de uma unidade organizacional resultante de um esforço de melhoria de processo e para esse esforço de melhoria. Quando um PCP é definido e utilizado segundo a metodologia PRO2PI, ele é denominado de PRO2PI.

PRO2PI-MODEL é baseada na unificação e extensões de dois frameworks para modelos de capacidade de processo: o framework da ISO/IEC 15504 [ISO/IEC 15504-2 2003] e o do CMMI.

Elementos de outros modelos, como, por exemplo, ISO 9001, SWEBOK, PMBOK, OPM3 e SW-CMM, podem também ser modelados e integrados com PRO2PI-MODEL. Com isto é possível definir PCPs baseados em elementos de múltiplos modelos.

O projeto de PRO2PI-MODEL busca também minimizar a quantidade de elementos. Elementos genéricos foram identificados de forma a representar, por meio de especializações, os elementos de PRO2PI.

Segundo PRO2PI-MODEL, todo modelo de capacidade de processo é composto por práticas de referência para melhoria de processo. Conceitualmente, PRO2PI-MODEL pode então ser resumido por uma classe abstrata, que representa esta prática de referência (*Cl\_PracticeReference*). Esta classe abstrata é especializada em três subclasses, também abstratas, que representam três níveis hierárquicos de práticas de referências (*Cl\_PracticesSystem*, *Cl\_PracticesGroup* e *Cl\_PracticeElement*), conforme representado no diagrama simplificado na Figura 68.

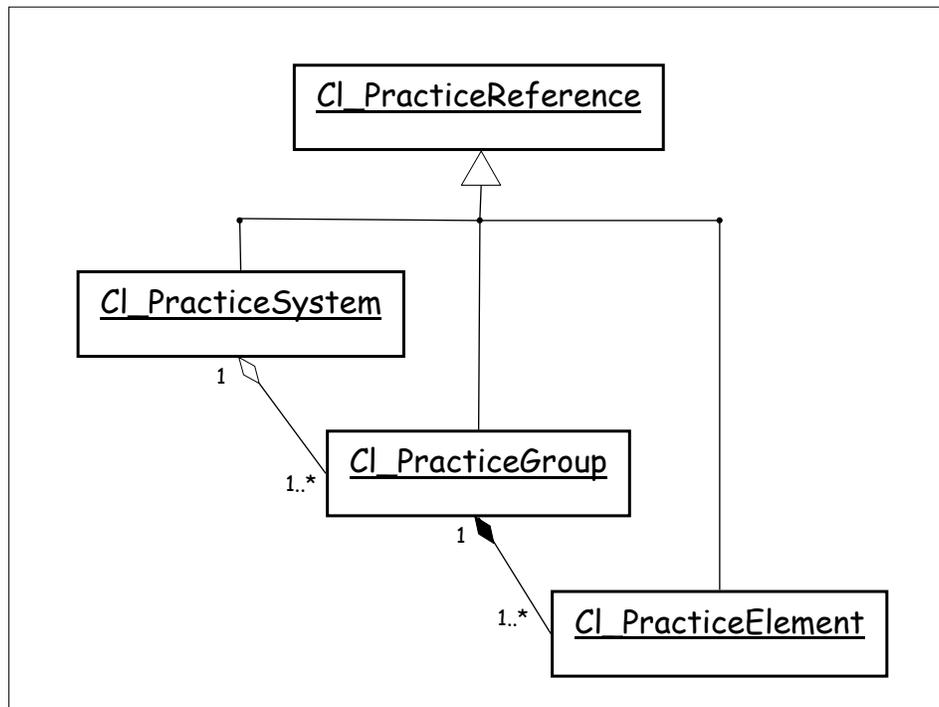


Figura 68 – Diagrama simplificado de PRO2PI-MODEL

A classe *Cl\_PracticeElement* (Elemento de Prática) representa os vários tipos de práticas elementares, como, por exemplo, a prática “estimar tamanho do software a ser desenvolvido”. A classe *Cl\_PracticesGroup* (Grupo de Práticas) representa os dois tipos de agrupamento de práticas, que são, nível de capacidade de processo e área de processo, como, por exemplo, a área de processo

“planejamento de projeto”. A classe *Cl\_PracticesSystem* (Sistema de Práticas) representa um conjunto de grupo de práticas, como, por exemplo, o nível 2 de maturidade.

A Figura 69 contém o diagrama de classes de PRO2PI-MODEL, com as dezesseis classes e seus relacionamentos. Como nome de cada classe é utilizado o termo em inglês com o prefixo “CI” (*Class*).

Conceitualmente, PRO2PI-MODEL pode ser resumido por três classes abstratas (*Cl\_PracticesSystem*, *Cl\_PracticesGroup* e *Cl\_PracticeElement*) que representam três níveis decrescentes de granularidade de práticas de referência para a melhoria de processo. Por isto essas três classes são generalizadas na classe abstrata *Cl\_ReferencePractice*. A classe *Cl\_PracticeElement* é especializada pelas classes *Cl\_Objective*, *Cl\_Outcome*, *Cl\_BasePractice*, *Cl\_Artefact* e *Cl\_Resource*. A classe *Cl\_PracticesGroup* é composta por um conjunto de *Cl\_Objective* e suas composições. A classe *Cl\_PracticesGroup* é especializada pelas classes *Cl\_ProcessArea* e *Cl\_CapabilityLevel*. A classe *Cl\_ProcessAreasGroup* é também uma especialização de *Cl\_ReferencePractice* e é composta por *Cl\_ProcessAreas* ou outros *Cl\_ProcessAreasGroups*.

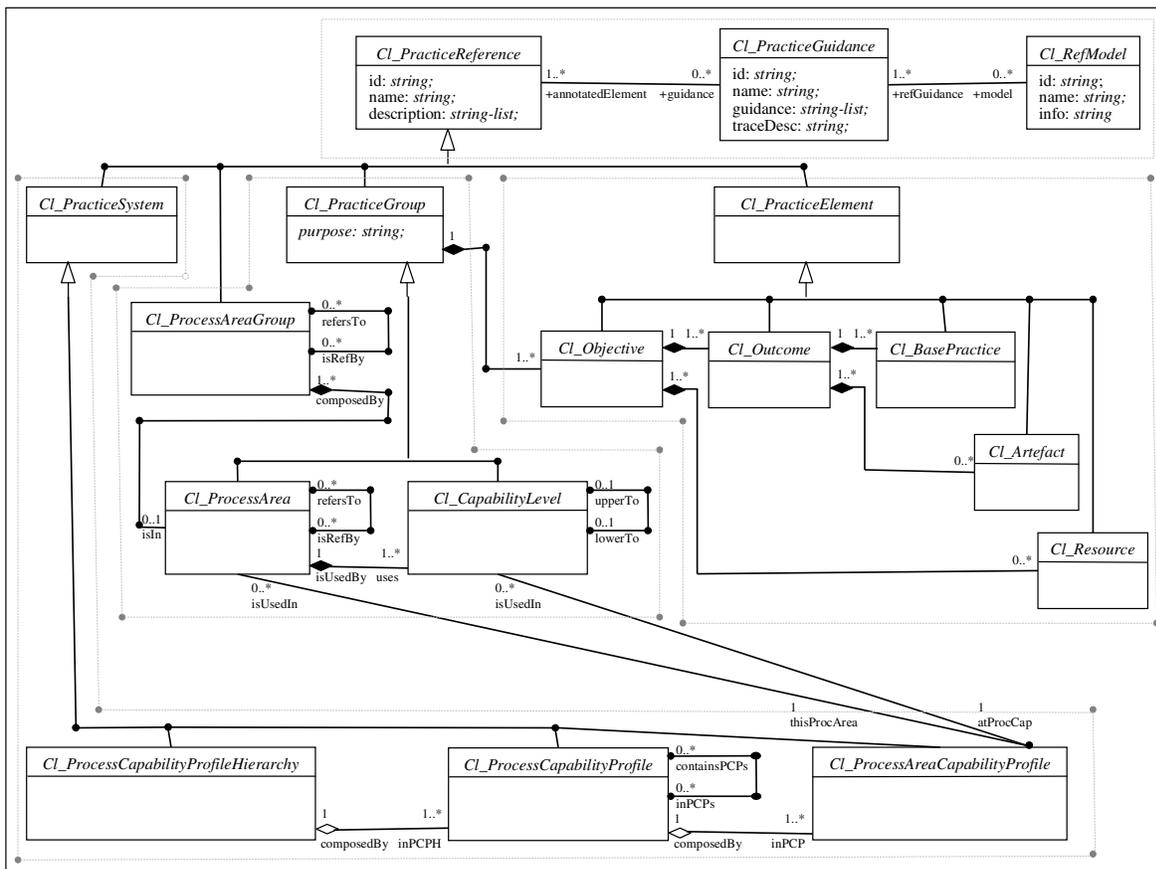


Figura 69 - Diagrama de classes de PRO2PI-MODEL em UML

A classe `Cl_PracticesSystem` é especializada em `Cl_ProcessCapabilityProfilesHierarchy`, `Cl_ProcessCapabilityProfile` e `Cl_ProcessAreaCapabilityProfile` que representam três níveis decrescentes de granularidades de perfis de capacidade de processo. A classe `Cl_ProcessAreaCapabilityProfile` é uma associação de uma `Cl_ProcessArea` com um `Cl_CapabilityLevel`. A classe `Cl_ProcessCapabilityProfile` é composta por `Cl_ProcessAreaCapabilityProfiles`. A classe `Cl_ProcessCapabilityProfilesHierarchy` é composta por `Cl_ProcessCapabilityProfiles`. A Tabela 18 relaciona o nome, em inglês e português, de cada uma das dezessete classes de PRO2PI-MODEL e o Apêndice A.1 apresenta a especificação dessas classes.

Tabela 18 – Nomes das classes de PRO2PI-MODEL em inglês e português

Nome em inglês	Nome em Português
<code>Cl_ReferencePractice</code>	Prática de Referência
<code>Cl_PracticeGuidance</code>	Guia de Prática
<code>Cl_SourceModel</code>	Modelo Fonte
<code>Cl_PracticeElement</code>	Elemento de Prática
<code>Cl_Objective</code>	Objetivo
<code>Cl_Outcome</code>	Resultado
<code>Cl_BasePractice</code>	Prática Base
<code>Cl_Artifact</code>	Artefato
<code>Cl_Resource</code>	Recurso
<code>Cl_PracticesGroup</code>	Grupo de Práticas
<code>Cl_ProcessArea</code>	Área de Processo
<code>Cl_ProcessAreasGroup</code>	Grupo de Áreas de Processo
<code>Cl_CapabilityLevel</code>	Nível de Capacidade
<code>Cl_PracticesSystem</code>	Sistema de Práticas
<code>Cl_ProcessAreaCapabilityProfile</code>	Perfil de Capacidade de Área de Processo
<code>Cl_ProcessCapabilityProfile</code>	Perfil de Capacidade de Processo
<code>Cl_ProcessCapabilityProfilesHierarchy</code>	Hierarquia de Perfis de Capacidade de Processo

A Figura 70 relaciona os principais elementos e os principais relacionamento entre os elementos de PRO2PI-MODEL.

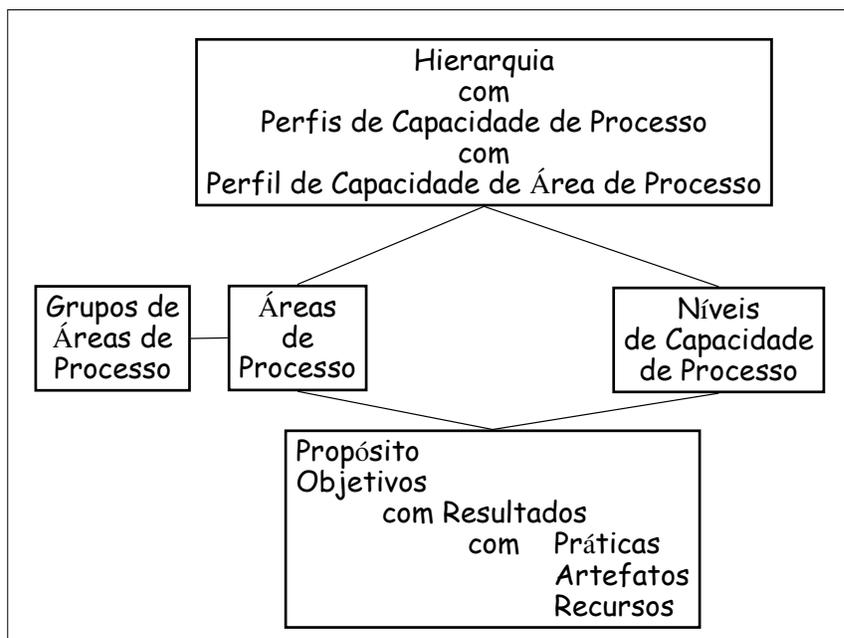


Figura 70 – Principais elementos de PRO2PI-MODEL e seus relacionamentos

A Figura 71 relaciona os principais elementos da estrutura dos modelos CMMI-SE/SW v1.1 e ISO/IEC 15504-5:2006 e indica como esses elementos são considerados na estrutura de PRO2PI-MODEL.

CMMI-SE/SW v1.1	PRO2PI-MODEL	ISO/IEC 15504-5 :2006
	(Grupo de Práticas)	
Área de Processo	Área de Processo	Processo
Propósito	Propósito	Propósito
Objetivo	Objetivo	"Propósito"
"Prática Específica"	Resultado	Resultado
Prática Específica	Prática Base	Prática Base
Produto Típico	Artefato	Produto de Trabalho
" "	Recurso	" "
	(Grupo de Práticas)	
Nível de Capacidade	Nível de Capacidade	Nível de Capacidade
Objetivo Genérico	Propósito	Propósito
"Prática Genérica"	Objetivo	Atributo de Processo
Prática Genérica	Resultado	Resultado
Sub-prática	Prática Base	Prática Genérica
" "	Artefato	Produto Genérico
" "	Recurso	Recurso

Figura 71 – Elementos da estrutura do CMMI e 15504-5 em PRO2PI-MODEL

## 5.4 PRO2PI-MEAS: Medições para PRO2PI

PRO2PI-MEAS (PRO2PI Measurements, Medições de PRO2PI) é conjunto de medições relacionadas a PRO2PI. A viabilidade de um PRO2PI pode ser medida com um modelo de informação de medição como uma função da complexidade do PRO2PI e da efetividade do investimento a ser feito na melhoria. Esta tese define uma medição da complexidade de um PRO2PI, da efetividade de um investimento para melhoria e da viabilidade de um PRO2PI. PRO2PI-MEAS utiliza o modelo de informação de medição definido na Norma ISO/IEC 15939 [2002] e utilizado no PSM [Aguiar 2002, McGarry et al. 2002] e nas áreas de processo de medição e análise dos modelos CMMI-SE/SW e ISO/IEC 15504-5. Para tanto são definidos:

- Uma medida derivada da complexidade de uma melhoria de processo representada por um PRO2PI (*Derived Measure Process Improvement Complexity DM-PIC*) em relação à situação atual de uma unidade organizacional, em termos de Unidades de Melhoria de Processo (*Unit for Process Improvement Complexity U-PIC*),
- Uma medida derivada da efetividade de investimento para melhoria de processo em uma unidade organizacional (*Derived Measure for Investment Effectiveness for Process Improvement DM-IEPI*), em termos de Unidades de Eficiência de Investimento para Melhoria de Processo (*Unit for Investment Effectiveness for Process Improvement U-IEPI*), e
- o produto de informação denominado Viabilidade de PRO2PI, em termos de uma interpretação de DM-PIC e DM-IEPI.

A Figura 72 ilustra o modelo de medição, segundo a estrutura definida na Norma ISO/IEC 15939.

O modelo de análise da viabilidade é uma função da complexidade de um PRO2PI e da efetividade do investimento de uma Unidade Organizacional, e tem valor normalizado da viabilidade igual a 1, quando a complexidade da melhoria modelada por PRO2PI é compatível com a capacidade da unidade organizacional. O valor de viabilidade é maior que 1, quando a melhoria é maior que a capacidade, e menor que 1, quando a melhoria é menor que a capacidade. Um PRO2PI é considerável viável para uma Unidade Organizacional quando o valor normalizado da viabilidade estiver no intervalo [0.8 – 1.2], ou seja, dentro da faixa de 20% para cima ou para baixo. A especificação da medição de complexidade de uma melhoria de processo está descrita no Apêndice A.2.

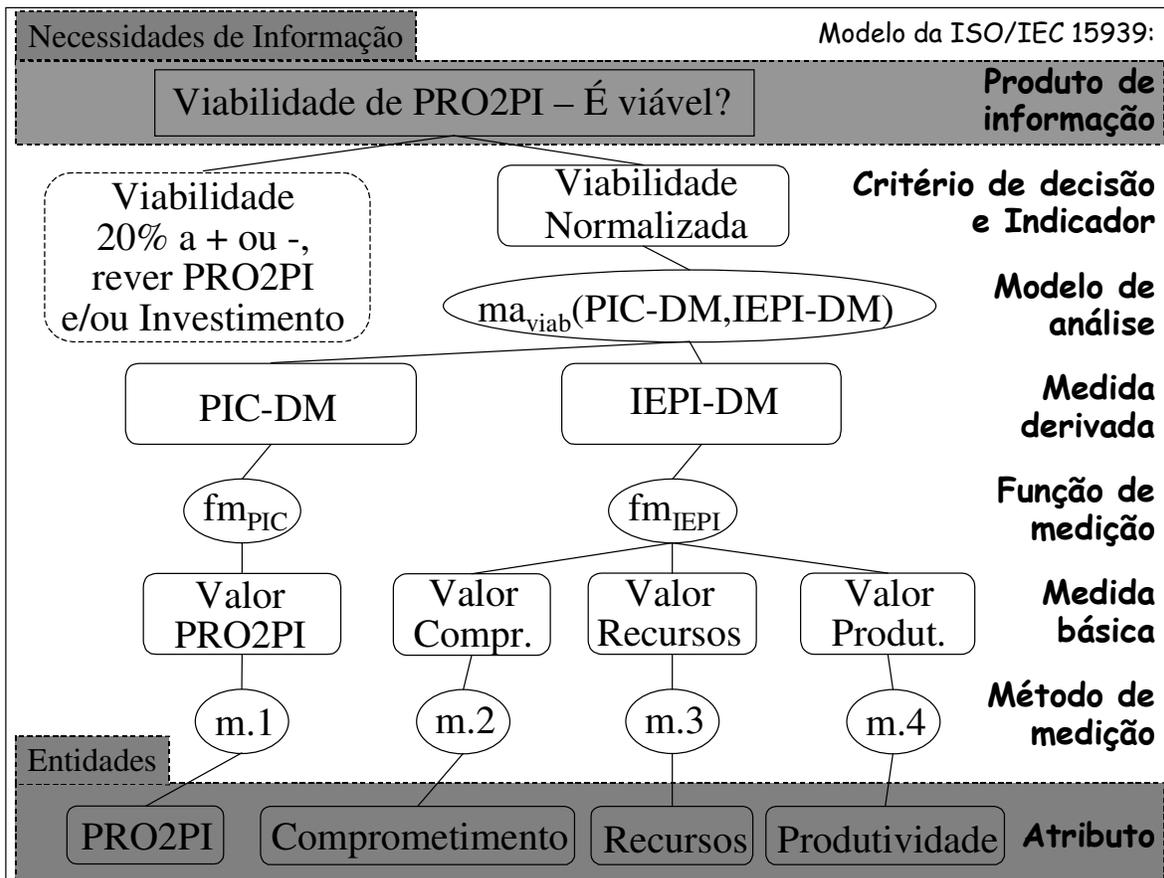


Figura 72 - Produto de informação Viabilidade de PRO2PI

A efetividade de um investimento em melhoria por uma organização pode ser calculada como uma função do comprometimento da organização com a melhoria, da quantidade e qualidade de investimento a ser feita, incluindo valores financeiros, recursos humanos e duração, e da produtividade da organização em ações de melhoria. A influência do comprometimento para a melhoria de processo é identificada por vários autores, incluindo Humphrey [1989] e Kasse e McQuaid [2000], estudada em detalhes por Abrahamsson [2002]. Formas de medições desse comprometimento são estudadas em Scheible [2004] e Scheible e Bastos [2005]. Esta medição não é desenvolvida neste trabalho. Haverá a atribuição de um valor baseado em estimativa.

Conforme descrito na Figura 72, a viabilidade de um PRO2PI é determinada pelo resultado de um modelo de análise da complexidade e da efetividade de investimento.

## **5.5 PRO2PI-CYCLE: Processo para ciclo de melhoria com PRO2PI**

A realização de melhoria de processo baseada em modelos é realizada com a utilização de abordagens. Entre elas podem ser destacadas a abordagem IDEAL, ciclo de melhoria da ISO/IEC 15504 e a abordagem AMP1. IDEAL foi definida para arquitetura estagiada fixa, e ciclo 15504-4 e AMP1 são definidas para arquitetura estagiada fechada.

O processo para ciclo de melhoria com PRO2PI utiliza como referência as três abordagens citadas. Um ciclo de melhoria com PRO2PI pode ser entendido como composto por fases correspondentes às fases das abordagens citadas com o acréscimo de uma atividade para definição e utilização de PRO2PI que pode ser utilizada em qualquer fase. Desta forma são enfatizadas e detalhadas as modificações definidas neste trabalho.

O trabalho é iniciado com a decisão e comprometimento da organização em realizar a melhoria. As atividades desse ciclo são alinhadas com o contexto e os objetivos estratégicos da organização e podem utilizar experiências e os resultados de outras organizações. Esse alinhamento é essencial para a melhoria porque o objetivo é a melhoria do negócio da organização, por meio da melhoria do processo. A referência principal para a melhoria é perfil de capacidade de processo, cuja definição e utilização é encapsulada em ‘define e utiliza PRO2PI’.

A Figura 73 ilustra as seis fases e outros elementos do processo PRO2PI-CYCLE.

O objetivo principal das atividades de definição e utilização de PRO2PI é definir e utilizar PRO2PI. O produto de entrada e de saída é o perfil de capacidade de processo. As atividades, que podem ser consideradas como uma implementação das práticas base definidas para o processo de estabelecimento de processo, que está descrito na Tabela 17. Os objetivos de PRO2PI-CYCLE incluem:

- identificar e analisar os objetivos, estratégia, contexto e/ou qualquer outro aspecto relevante de negócio da unidade organizacional e da organização, para subsidiar e orientar a definição dos objetivos de melhoria;
- identificar os objetivos da melhoria, incluindo objetivos mais específicos para o próximo ciclo de melhoria e objetivos mais gerais para o programa de melhoria, sempre alinhados aos objetivos, estratégia, contexto e/ou qualquer outro aspecto relevante de negócio identificados;
- estabelecer critérios de qualidade para avaliar e melhorar um perfil de capacidade de processo;

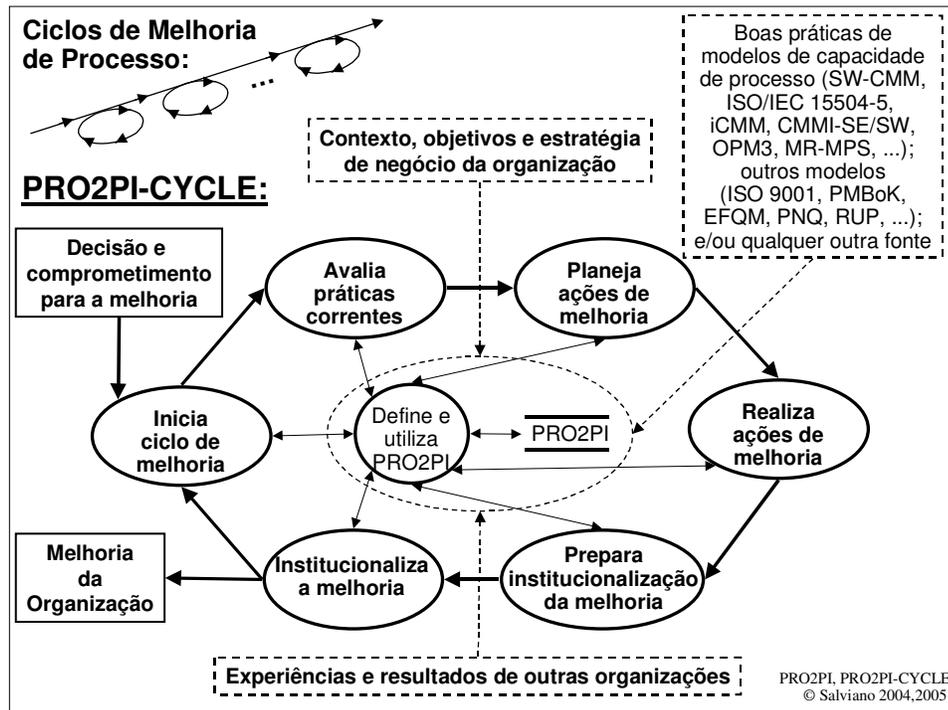


Figura 73 - PRO2PI-CYCLE: Processo para ciclo de melhoria com PRO2PI

- identificar modelos de capacidade de processo que sejam relevantes para a organização, e selecionar áreas de processo desses modelos e, caso necessário, definir novas áreas de processo, que sejam relevantes para o processo atual e sua melhoria;
- definir, selecionar e/ou adaptar um perfil de capacidade de processo com áreas de processo selecionadas e/ou definidas, de forma que esse perfil esteja alinhado aos objetivos de melhoria;
- utilizar o perfil de capacidade de processo definido para orientar o entendimento dos processos correntes da unidade organizacional;
- utilizar o perfil de capacidade de processo definido e os resultados do entendimento dos processos correntes, para orientar as ações de melhoria dos processos da unidade organizacional;
- analisar o perfil de capacidade de processo definido segundo os critérios de qualidade estabelecidos, quando necessário, como, por exemplo, em marcos definidos do ciclo de melhoria, em eventos ou periodicamente.
- ajustar o perfil de capacidade de processo definido segundo resultados de análise do perfil;
- monitorar o perfil de capacidade de processo definido em relação aos objetivos estratégicos para identificar possíveis desvios no alinhamento entre o perfil e os objetivos estratégicos;

- atualizar o perfil de capacidade de processo definido sempre que o monitoramento identificar desvios significativos no alinhamento entre o perfil e os objetivos estratégicos, de forma que esse alinhamento seja mantido; e
- buscar, por meio de ajustes no perfil e/ou ações de melhoria, a correspondência entre o perfil de capacidade de processo definido e o processo da unidade organizacional

PRO2PI-CYCLE é um ciclo com seis fases seqüenciais. A Figura 74 ilustra o relacionamento entre as fases de PRO2PI-CYCLE e as fases das abordagens IDEAL, ciclo de melhoria da 15504 e AMP1.

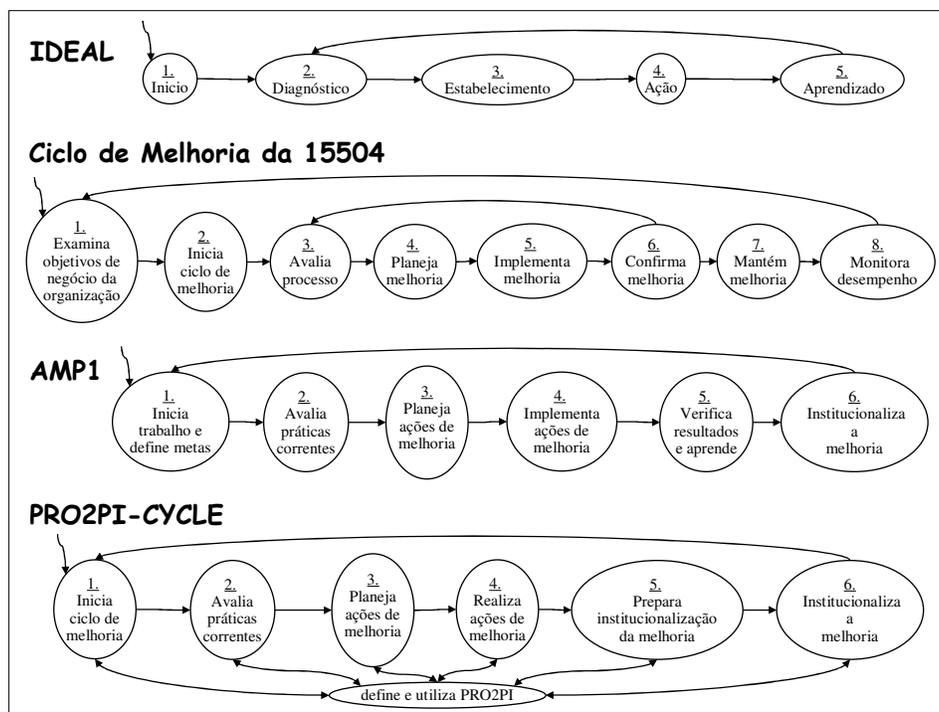


Figura 74 – Fases de PRO2PI-CYCLE e de IDEAL, ciclo 15504 e AMP1

Nas seis fases de PRO2PI-CYCLE, três das quatro funções que caracterizam a abordagem PRO2PI, que são “defineP”, “avaliaPr” e “usaP” e estão representadas na Figura 63, são utilizadas com frequências e intensidades diferentes. Na fase de “inicia ciclo de melhoria” a função “defineP” é mais utilizada. Na fase de “avalia práticas correntes” a função “avaliaPr” é mais utilizada. Nas fases de “planeja ações de melhoria” e “realiza ações de melhoria” a função “usaP” é mais utilizada.

A Figura 75 apresenta PRO2PI-CYCLE como um processo na notação ETVX.

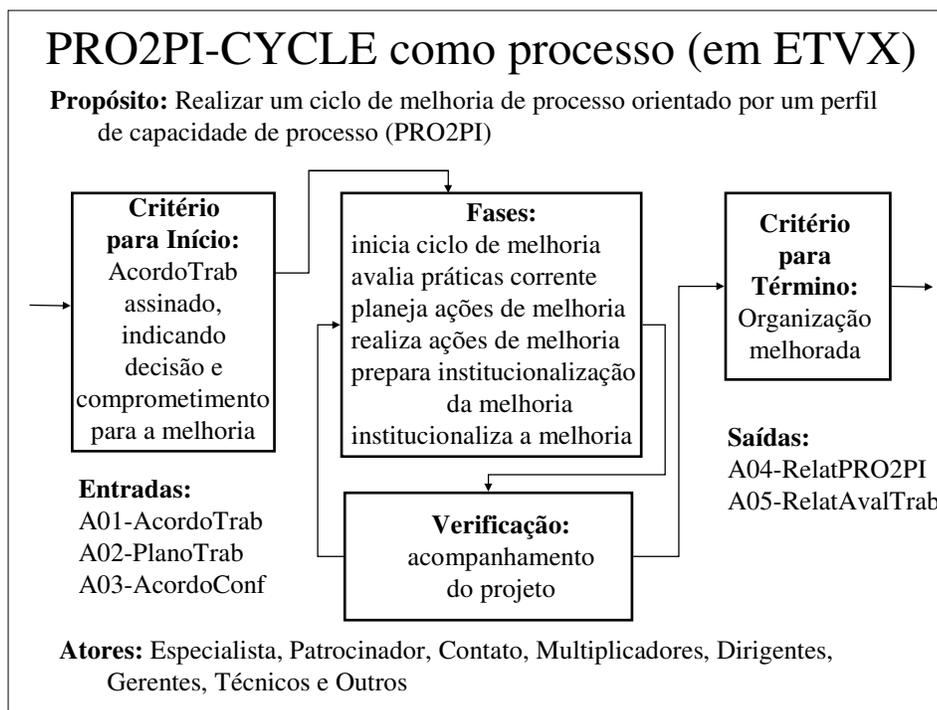


Figura 75 – PRO2PI-CYCLE como um processo na notação ETVX

A seguir, para cada fase estão descritos: o objetivo, os produtos de entrada, os produtos de saída e as atividades principais.

### Fase 1: Inicia ciclo de melhoria

**Propósito:** Examinar o contexto e objetivos estratégicos da organização para iniciar o ciclo de melhoria, identificar a unidade organizacional, identificar objetivos da melhoria e definir perfil de capacidade de processo.

**Produtos de entrada:** A01-AcordoTrab; A02-PlanoTrab e A03-AcordoConf.

**Produto de saída:** A02-PlanoTrab,

#### Atividades:

- Levantamento de informações gerais sobre a empresa, incluindo o contexto e objetivos estratégicos, metodologias existentes, tecnologias e projetos existentes;
- Definição das Metas de Melhoria;
- Definição do Perfil de Capacidade de Processo;
- Revisão e detalhamento do Programa de Melhoria de Processo; e
- Definição do Plano Preliminar da Avaliação das Práticas Correntes.

**Fase 2: Avalia práticas correntes**

**Propósito:** Conhecer as práticas correntes da organização relacionadas ao perfil de capacidade de processo selecionado.

**Produto de entrada:** A02-PlanoTrab

**Produto de saída:** A04-ResultAval

**Atividades:**

- Revisão do Plano da Avaliação das Práticas Correntes;
- Treinamento da equipe de avaliação sobre o processo da avaliação;
- Adaptação das práticas de referência para a avaliação;
- Levantamento e validação de dados para a avaliação dos projetos e processos selecionados;
- Derivação e elaboração dos resultados da avaliação;
- Apresentação dos resultados da avaliação.
- Revisão do perfil de capacidade de processo.

**Fase 3: Planeja ações de melhoria**

**Propósito:** Analisar o resultado da avaliação, planejar as ações de melhoria e preparar a organização para execução das ações de melhoria.

**Produto de entrada:** A04-ResultAval

**Produtos de saída:** A02-PlanoTrab;e A05-PlanoMelh

**Atividades :**

- Treinamento de pessoas selecionadas sobre o processo de planejamento da melhoria e elementos para sua implantação;
- Revisão e discussão dos resultados da avaliação e das metas de melhoria;
- Definição da estratégia para implantação da melhoria;
- Definição das ações de melhoria;
- Elaboração do Plano de Melhoria de Processo;
- Definição das responsabilidades e necessidades de compromissos;
- Elaboração do Plano de Preparação da Organização;
- Revisão do Plano de Melhoria de Processo por pessoas selecionadas;
- Estabelecimento das responsabilidades para execução do Plano de Melhoria de processo, incluindo as funções de gerente, grupo de processo e grupos de implementação de processo; e

- Estabelecimento dos compromissos para execução do Plano.
- Atividades para consolidação de planejamento
- Revisão das atividades e resultados da sub-fase anterior;
- Atualização do Plano;
- Treinamento de pessoas selecionadas sobre elementos para implantação do Plano; e
- Apresentação para início da implantação das ações de melhoria.

#### **Fase 4: Realiza ações de melhoria**

**Propósito:** realizar as ações de melhoria descritas no Plano.

**Produtos de entrada:** A02-PlanoTrab e A05-PlanoMelh.

**Produto de saída:** conjunto de descrição dos processos definidos.

#### **Atividades:**

- Realizar atividades descritas no Plano
- Monitorar as atividades conforme o Plano
- Revisar o plano e realizar ações corretivas, quando necessário.

Como exemplo de um plano, a seguir estão descritos quatro etapas na qual a implementação pode ser dividida:

**Etapa 1:** Nesta etapa são elaborados e/ou melhorados alguns aspectos de alguns processos. Esse aspectos são aqueles que foram definidos, durante a fase de planejamento, como de maior prioridade e que possam gerar resultados concretos no curto prazo. Como resultado principal são geradas versões dos processos.

**Etapa 2:** Nesta etapa, a versão dos processos gerados na Etapa 1 são aplicadas em um projeto piloto. Para esta aplicação os processos podem ser detalhados e com o acompanhamento que é feito desta aplicação, podem até ser revisados. Como resultado principal são gerados relatórios sobre os resultados desta aplicação e as versões detalhadas e revistas dos processos.

**Etapa 3:** Nesta etapa são elaborados e/ou melhorados outros aspectos dos processos. Esses aspectos são aqueles definidos na fase de planejamento que não foram tratados na etapa 1. Como resultado principal são geradas versões dos processos.

**Etapa 4:** Nesta etapa, a versão dos processos geradas na Etapa 3 são aplicadas em um segundo projeto piloto. Para esta aplicação os processos podem ser detalhados e com o acompanhamento que é feito desta aplicação, podem até ser revisados. Como resultado principal são gerados relatórios sobre os resultados desta aplicação e as versões detalhadas e revistas dos processos.

**Fase 5: Prepara institucionalização da melhoria**

**Propósito:** Verificar o resultados da implementação das ações de melhoria e preparar a institucionalização da melhoria.

O produto principal resultante desta fase é um relatório com a verificação, planejamento de ajustes e principais lições aprendidas.

**Atividades:**

- Revisão do resultado da melhoria, e
- Elaboração do Plano de institucionalização.

**Fase 6: Institucionaliza a melhoria**

**Propósito:** Institucionalizar os resultados positivos da melhoria conforme o Plano de Institucionalização.

**Atividades:**

- Realizar atividades descritas no Plano de Institucionalização
- Monitorar as atividades conforme o Plano de Institucionalização
- Revisar o plano de Institucionalização e realizar ações corretivas, quando necessário.

## 5.6 Síntese do capítulo

Esse capítulo descreveu a abordagem PRO2PI (Perfil de Capacidade de Processo para Melhoria de Processo, em inglês, *Process Capability Profile to Process Improvement*) para uma melhoria de processo dirigida por perfis de capacidade de processo. Esta abordagem é um exemplo de abordagem para a engenharia de processo PCDE descrita no capítulo anterior. A descrição de PRO2PI é feita com uma visão geral da utilização de PRO2PI e os quatro elementos que compõem a abordagem PRO2PI. Esses quatro elementos são:

- Um conjunto de oito propriedades (relevante, oportuno, viável, sistêmico, representativo, rastreável, específico e dinâmico) que um PCP deve ter, em um grau suficiente, para ser um PRO2PI;
- Um modelo que (i) unifica os elementos das estruturas dos modelos de capacidade da maturidade, especialmente dos modelos iCMM v2.0, CMMI-SE/SW v1.1, ISO/IEC 15504-5 e MR-MPS v1.0, e com isto permite a representação de todos elementos desses modelos em um PRO2PI, (ii) permite a representação em um PRO2PI de elementos selecionados de outros modelos de

referência, especialmente PMBOK e ISO 9001, (iii) é representado com um diagrama de classes e pode ser representado como um metamodelo no meta-metamodelo MOF, e, (iv) sintetiza todos os elementos desses modelos em uma única classe `CI_ReferencePractice` (prática de referência), que é especializada em três subclasses `CI_PracticeElement`, `CI_PracticesGroup` e `CI_PracticesSystem` (respectivamente elemento de prática, grupo de práticas e sistema de práticas);

- Um conjunto inicial de medições relacionadas a PRO2PI, incluindo uma medição da complexidade de um PRO2PI e da viabilidade de um PRO2PI; e
- Um ciclo de melhoria com PRO2PI composto por fases correspondentes às fases dos ciclos de melhoria convencionais, principalmente, IDEAL, Ciclo 15504 e AMP1, acrescido de uma atividade para definição e utilização de PRO2PI que pode ser utilizada em qualquer fase.

## Capítulo 6

# 6 Método PRO2PI-WORK para estabelecimento de PRO2PI

O objetivo deste capítulo é descrever um método para o estabelecimento de um PRO2PI que possa ser utilizado nas fases iniciais de PRO2PI-CYCLE e seja viável para micro e pequenas empresas.

### 6.1 Apresentação

O método é denominado PRO2PI-WORK (*PRO2PI Establishment Workshop Method*, Método para Oficina de Estabelecimento de PRO2PI). Esse método realiza de forma compactada as atividades das três primeiras etapas do ciclo de melhoria PRO2PI-CYCLE, mais as atividades de definição e utilização de PRO2PI. PRO2PI-WORK é composto por quatro fases sequenciais: preparação do trabalho, escolha de PRO2PI, orientações para PRO2PI e conclusão do trabalho.

Considerando que (i) as abordagens para melhoria de processo de software e os modelos de capacidade de processo foram desenvolvidos para grandes e médias organizações, e (ii) as micro e pequenas empresas representam a maioria das empresas de software no Brasil, é importante identificar as características específicas das micro e pequenas empresas e desenvolver métodos para elas. A Figura 76 ilustra a relação de PRO2PI-WORK com PRO2PI-CYCLE e relaciona as quatro fases de PRO2PI-WORK.

PRO2PI-WORK é um exemplo de um método para realizar estas atividades, para um contexto determinado. Esse contexto é caracterizado por micro e pequenas organizações intensivas em software, com processos de baixa capacidade, que queiram iniciar um ciclo de melhoria o mais rápido possível, investindo poucos recursos e com resultados de curto prazo. Em geral PRO2PI-WORK pode ser realizado em 5 dias de trabalho, orienta melhorias que podem ser realizadas pelos próprios membros da organização, com resultados em cerca de 4 a 6 meses. A estratégia é identificar melhorias importantes para a organização, que possam ser realizadas com seus próprios recursos humanos e tenham resultados a curto prazo.

Uma variação do método PRO2PI-WORK pode ser utilizado também para cursos de engenharia de processo, com uma visão construtivista do aprendizado. Esta visão construtivista é baseada informalmente na linha para educação defendida, entre outros, por Freire [1996, p.25].

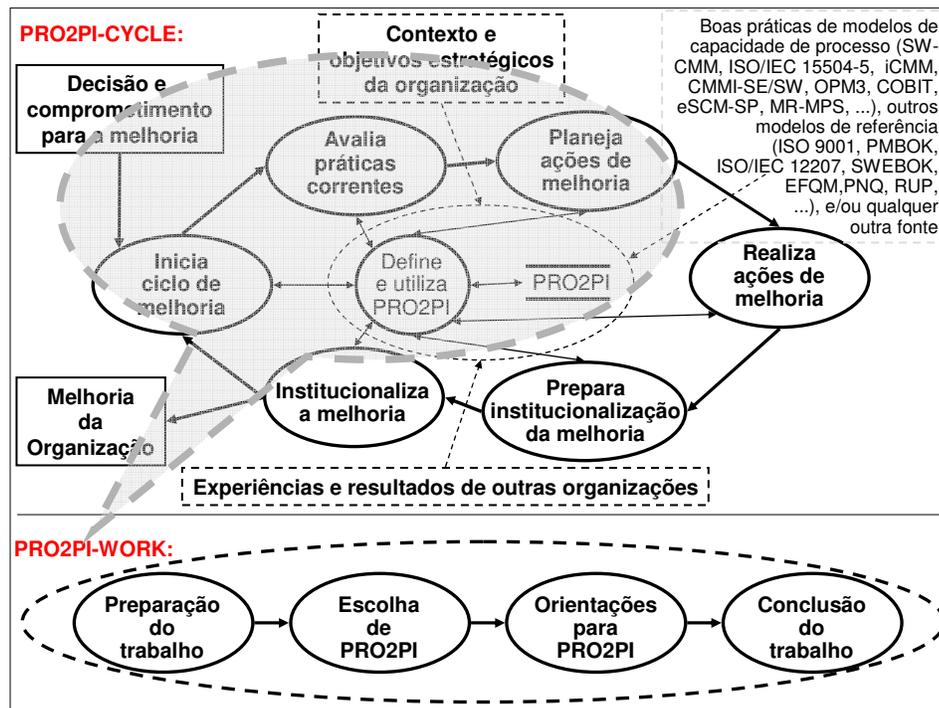


Figura 76 - Fases do método PRO2PI-WORK e relação com PRO2PI-CYCLE

O método PRO2PI-WORK é uma evolução do método MEP1<sup>12</sup> (Método para Escolha dos Processos para uma melhoria alinhada aos objetivos estratégicos) [Salviano 2001]. O método MEP1 foi desenvolvido em 1999 e foi utilizado inicialmente em três trabalhos de melhoria de processo, em três diferentes organizações de software brasileiras entre 1999 e 2001 [Salviano et al. 1999, Salviano e Souza 2000]. O método MEP1 foi também utilizado como base para o desenvolvimento do método MARES [Anacleto 2004, Anacleto et al. 2004a], que também experimentou parte das evoluções consolidadas em PRO2PI-WORK. Desta forma PRO2PI-WORK ao mesmo tempo influenciou e também utilizou MARES. PRO2PI-WORK tem características similares a outros métodos, como, por exemplo, a Abordagem Baseada em Problemas e Metas [Potter e Sakry 2002], método *CMMI in Small Settings Toolkit* [Garcia et al. 2005] e o método RAPID [Rout et al. 2000c]. Versões do que viria a ser o método PRO2PI-WORK foram também utilizadas em programas de melhoria de processo [Silva et

<sup>12</sup> Esta sigla (MEP1) foi definida neste trabalho para facilitar a referência a este método.

al. 2003, Anacleto et al. 2004b]. Como referência para cursos em melhoria de processo, PRO2PI foi utilizado em quatro cursos de pós-graduação em 2005.

A estratégia para definição do perfil de capacidade de processo é baseada na consolidação de informações relevantes sobre a organização, análise de um conjunto de áreas de processo em relação à organização e escolha das áreas de processo. As informações sobre a organização incluem pontos fortes, fraquezas, oportunidades, ameaças e objetivos estratégicos. São levantadas informações objetivas que servem de subsídios para uma decisão subjetiva e coletiva.

Esse método foi desenvolvido para, e está sendo aplicado em, organizações de software com processos com baixa capacidade, ou seja, buscando até o nível 3 de capacidade. Organizações com processos com alta capacidade, principalmente com equivalente aos níveis 4 e 5 de maturidade do SW-CMM ou CMMI-SE/SW, já devem ter mecanismos mais sistemáticos já estabelecidos de estratégia e gestão que podem ser utilizados para a escolha dos processos. O método desenvolvido é simples o suficiente para ser viável e útil para organizações com baixa capacidade e por isto tem limitações que podem torna-lo inadequado para organizações com alta capacidade. Sheard e Roedler [1999], por exemplo, apresentam considerações sobre interpretação da visão contínua de modelos de capacidade para os níveis alto de capacidade.

## **6.2 PRO2PI-WORK descrito como um processo**

A seguir o método PRO2PI-WORK é ilustrado [Figura 77] e descrito como um processo, na notação ETVX [Radice e Phillips 1988] em termos de propósito, critério para início, entradas, tarefas ou fases, critério para término, saídas e atores. Esta descrição é complementada com considerações sobre medições e relação dos artefatos produzidos ou utilizados no processo.

O propósito do processo PRO2PI-WORK é estabelecer um perfil de capacidade de processo como indutor de um ciclo de melhoria de processo (PRO2PI). Esse propósito é desdobrado nos seguintes objetivos específicos:

- Capacitar membros da organização em fundamentos da engenharia de processo e de modelos relevantes de capacidade de processo,
- Consolidar informações relevantes sobre a organização,
- Consolidar objetivos estratégicos para a melhoria,
- Definir um PRO2PI,
- Entender os processos atuais em relação ao PRO2PI definido,

- Consolidar uma representação em alto nível dos processos atuais,
- Definir orientações para atingir PRO2PI, e
- Reforçar motivação para melhoria de processo.

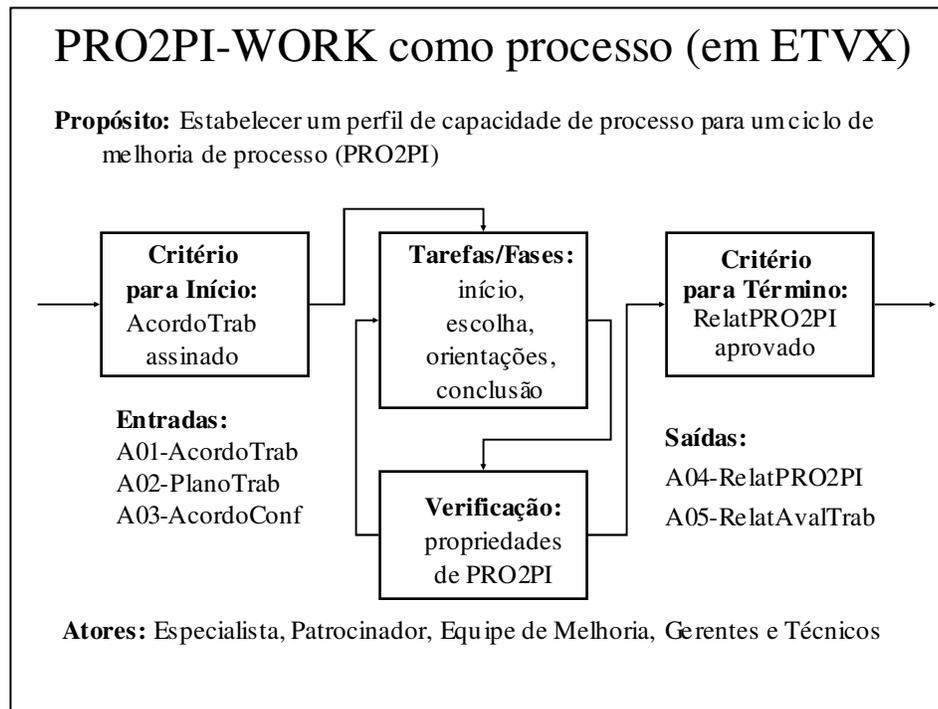


Figura 77 – PRO2PI-WORK como um processo em ETVX

Os atores (ou papéis) para esse processo são os seguintes:

- Especialistas: Equipe de consultores, internos e/ou externos à organização, especialistas em melhoria de processo, no processo PRO2PI-WORK e nos modelos relevantes de capacidade processo.
- Patrocinador: Representante da organização, com cargo de direção apropriado para garantir os recursos para a melhoria e demandar o atendimento dos objetivos estratégicos.
- Contato: Pessoa da empresa, indicada pelo patrocinador, para contato com os especialistas.
- Multiplicadores: Equipe de melhoria de processo da organização, participantes e treinandos no trabalho para serem multiplicadores, coordenadas por um líder.
- Dirigentes: Representantes dos dirigentes da organização.
- Gerentes: Representantes dos gerentes da organização.
- Técnicos: Representantes dos técnicos da organização.
- Outros: Outras pessoas.

Todas as equipes têm um coordenador designado. Todas as referências para os atores serão feitas no singular, indicando, no caso de equipe, tanto a equipe toda, ou apenas o coordenador, ou mesmo um subconjunto dos integrantes.

As entradas para esse processo são os seguintes:

- A01-AcordoTrab,
- A02-PlanoTrab, e
- A03-AcordoConf.

Os resultados desse processo são os seguintes:

- A04-RelatPRO2PI, e
- A05-RelatAvalTrab.

O critério de início para esse processo é o artefato A01-AcordoTrab assinado pelas partes, e o critério de conclusão é o artefato A04-RelatPRO2PI aceito pelo cliente.

As fases são início, definição, orientações e conclusão. A Figura 78 relaciona o nome das fases e atividades do método.

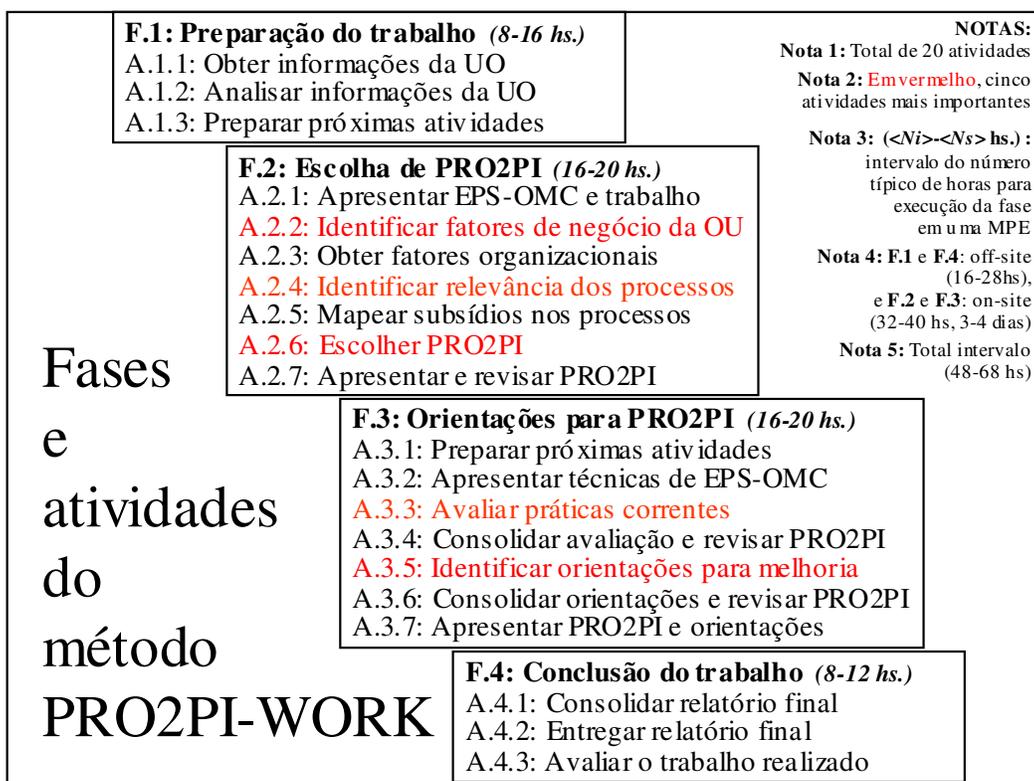


Figura 78 – Fases e atividades do método PRO2PI-WORK

A Figura 79 relaciona o tempo típico para realização de cada atividade e a soma das horas das atividades de cada fase, em uma micro empresa.

A medição principal utilizada é em relação às propriedades de um PRO2PI descrita como PRO2PI-PROP. A viabilidade de PRO2PI pode ser medida com o produto de informação Viabilidade de PRO2PI, descrita como PRO2PI-MEAS.

Vinte e cinco artefatos são produzidos e utilizados em PRO2PI-WORK. A Tabela 19 identifica, relaciona e descreve esses artefatos.

<b>Fases e Atividades</b>	<b>local e tipo</b>	<b>horas</b>
<b>F.1: Preparação do trabalho</b> .....	<b>fora da empresa</b>	<b>08</b>
A.1.1: Obter informações da Unidade Organizacional .....	pela empresa	02
A.1.2: Analisar informações da Unidade Organizacional ...	especialista	03
A.1.3: Preparar próximas atividades .....	especialista	03
<b>F.2: Escolha de PRO2PI</b> .....	<b>na empresa</b>	<b>12</b>
A.2.1: Apresentar abordagem e trabalho .....	apresentação	01
A.2.2: Identificar fatores de negócio .....	reunião	01
A.2.3: Obter fatores organizacionais .....	oficina	02
A.2.4: Identificar relevância dos processos .....	oficina	04
A.2.5: Mapear subsídios nos processos .....	interno	02
A.2.6: Escolher perfil de processo .....	interno	01
A.2.7: Apresentar e revisar perfil .....	oficina	01
<b>F.3: Orientações para PRO2PI</b> .....	<b>na empresa</b>	<b>12</b>
A.3.1: Preparar próximas atividades .....	interno	01
A.3.2: Apresentar técnicas da abordagem .....	apresentação	01
A.3.3: Identificar práticas correntes .....	oficina	02
A.3.4: Consolidar identificação e revisar perfil .....	interno	03
A.3.5: Definir orientações para melhoria .....	interno	01
A.3.6: Consolidar orientações e revisar perfil .....	oficina	03
A.3.7: Apresentar perfil e orientações .....	apresentação	01
<b>F.4: Conclusão do trabalho</b> .....	<b>fora da empresa</b>	<b>08</b>
A.4.1: Consolidar relatório final .....	especialista	06
A.4.2: Entregar relatório final .....	especialista	01
A.4.3: Avaliar o trabalho realizado .....	especialista	01

Figura 79 – Locais, tipos e tempos típicos de uma realização do método

Tabela 19 - Artefatos de PRO2PI-WORK

<b>Id</b>	<b>Nome</b>	<b>Descrição resumida</b>
A01	AcordoTrab	Acordo para realização do trabalho
A02	PlanoTrab	Plano do trabalho a ser realizado
A03	AcordoConf	Acordo de confidencialidade
A04	RelatPRO2PI	Relatório com contexto, PRO2PI, orientações para melhoria,...
A05	RelatAvalTrab	Relatório com avaliação do trabalho

O artefato A04-RelatPRO2PI é o principal resultado do processo e representa uma composição e consolidação de vários resultados intermediários. Esse artefato é iniciado com capa, controle, título, apresentação e resumo executivo; continuado com consolidações de R1DescUO, R2IntroEPS, R3ResTrab, R4DescSubsid, R5DescObjEstr, R6PCPAtual, R7DescPRO2PI, R8OrientMelh e R9GerMelh; e finalizado com anexos, que podem incluir os subsídios levantados e outros.

A descrição das fases e atividades do método PRO2PI-WORK é apresentada no Apêndice A.3.

### **6.3 Gerência de PRO2PI-WORK**

O método PRO2PI-WORK deve ser realizado como um processo de no mínimo nível 3 de capacidade. Ou seja, como um processo executado, gerenciado e estabelecido. Os parâmetros para adaptação do método PRO2PI incluem características do tipo de trabalho, da distribuição dos participantes, perfil dos participantes, distribuição do curso e a abrangência. Quanto ao tipo de trabalho, pode ser como uma consultoria, um curso ou um tutorial. Quanto à distribuição dos participantes, pode ser todos de uma mesma empresa, todos de um grupo de empresas ou diversos. Quanto ao perfil dos participantes, pode ser com pouco conhecimento dos modelos e de MPS, ou com um conhecimento razoável dos modelos e de MPS. Quanto à distribuição do curso, pode ser concentrado em um período contíguo ou distribuído em períodos de dias regulares. Quanto à abrangência, pode ser para uma determinada área de processo, algumas áreas de processo de um ou mais modelos, áreas de processo de um único modelo ou totalmente aberto.

### **6.4 PRO2PI-WORK para ensino de engenharia de processo**

O método PRO2PI-WORK também tem sido utilizado para ensino da engenharia de processo, com por exemplo em disciplinas relacionadas à engenharia de processo em cursos de pós-graduação (especialização) relacionados a engenharia e qualidade de software, e em treinamentos de projetos de engenharia de processo.

No caso das disciplinas de pós-graduação, os alunos são geralmente profissionais em software, com necessidades de conhecimento úteis ao seu dia-a-dia e com pouco tempo para aprendizado. Estas características apontaram para a utilização de PRO2PI-WORK, o que tem permitido a atividade de ensino como "ensinar não é transferir conhecimento, mas criar as possibilidades para a sua produção ou a sua construção" [Freire 1996, p. 25]. Na verdade PRO2PI-WORK foi desenvolvido ao tempo como método para consultoria em empresas e método para cursos de melhoria de processo. Esse

desenvolvimento também seguiu a orientação dos níveis de capacidade de processo, nesse caso para o processo de ensino de engenharia de processo. Foram desenvolvidos três versões do método: a versão 0.1, orientada pelo nível 1 de capacidade; a versão 0.2, orientada pelo nível 2 de capacidade, e a versão 0.3, orientada pelo nível 3 de capacidade. As fases e atividades da versão 0.3 estão descritas na Figura 80.

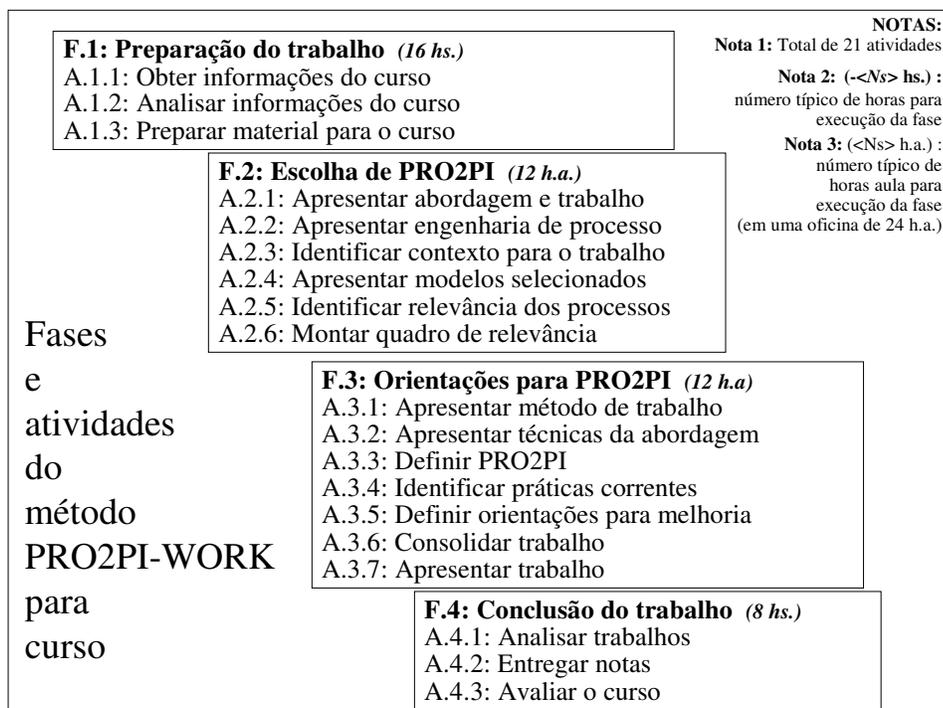


Figura 80 – PRO2PI-WORK para curso

## 6.5 Síntese do capítulo

Esse capítulo descreveu o método PRO2PI-WORK (*PRO2PI Establishment Workshop Method*, Método para Oficina de Estabelecimento de PRO2PI) para uma oficina de estabelecimento de PRO2PI com o objetivo principal de induzir o início de um ciclo de melhoria de processo baseado em um perfil de capacidade de processo. Esse método realiza de forma compactada as atividades da três primeiras etapas do ciclo de melhoria PRO2PI-CYCLE, mais as atividades de definição e utilização de PRO2PI, e é mais apropriado para micro e pequenas empresas. PRO2PI-WORK é composto por quatro fases sequenciais: preparação do trabalho, escolha de PRO2PI, orientações para PRO2PI e conclusão do trabalho. PRO2PI-WORK também pode ser utilizado para cursos de melhoria de processo de software.

# Capítulo 7

## 7 Análises e aplicações

O objetivo deste capítulo é validar as propostas desta tese em relação aos seus objetivos. Essa validação é buscada com análises e reflexões sobre aplicações. As análises realizadas estão relacionadas ao relacionamento dos elementos dos modelos de capacidade de processo mais relevantes em relação a propostas de gerações de arquiteturas, os três conceitos da terceira geração e os elementos do modelo unificado. As aplicações estão relacionadas com utilizações das várias versões de PRO2PI em projetos de melhoria de processo de software, estabelecimento de PRO2PI e capacitação em melhoria de processo de software.

Para definir os objetivos das análises e aplicações é utilizada a estrutura descrita por Travassos et al. [2002] que é baseada na engenharia de software experimental proposta por Basili et al. [1999]. Essa estrutura organiza a definição dos objetivos de uma análise em cinco itens, nos quais são identificados o que é analisado, com que propósito, com respeito a qual aspecto, sob qual ponto de vista e em que contexto.

A Tabela 20 descreve o objetivo da análise das gerações de arquitetura de modelos de capacidade de processo, dos conceitos de área de processo, nível de capacidade de processo e perfil de capacidade de processo, e dos elementos estruturais do modelo PRO2PI-MODEL, em relação aos modelos de capacidade de processo e modelos de outros tipos mais relevantes para a melhoria de processo.

Tabela 20 - Objetivo da análise das gerações de arquitetura

<b>Analisar</b>	as gerações de arquitetura de modelos de capacidade de processo, os conceitos de área de processo, nível de capacidade de processo e perfil de capacidade de processo e os elementos estruturais do modelo PRO2PI-MODEL,
<b>com o propósito</b>	de avaliar,
<b>com respeito</b>	à sua adequação para representar a estabilidade e variabilidade,
<b>do ponto de vista</b>	da unificação da representação das práticas representadas
<b>no contexto</b>	dos modelos de capacidade de processo e modelos de outros tipos mais relevantes para a melhoria de processo.

A Tabela 21 descreve o objetivo da análise das utilizações das várias versões de PRO2PI.

Tabela 21 – Objetivo da análise das utilizações de PRO2PI

<b>Analisar</b>	utilizações das várias versões de PRO2PI
<b>com o propósito</b>	de avaliar
<b>com respeito</b>	à viabilidade
<b>do ponto de vista</b>	a utilização prática para orientar a melhoria de processo de software
<b>no contexto</b>	de uma pesquisa científica orientada pela abordagem indústria-como-laboratório

A Tabela 22 descreve um objetivo particular, como desdobramento do objetivo de utilizações de PRO2PI. Nesse caso são analisadas as utilizações de PRO2PI-WORK em cursos de pós-graduação.

Tabela 22 – Objetivo da análise das utilizações de PRO2PI-WORK

<b>Analisar</b>	utilizações das várias versões PRO2PI-WORK
<b>com o propósito</b>	de avaliar
<b>com respeito</b>	à viabilidade
<b>do ponto de vista</b>	relacionamento de contexto e fatores de negócio com áreas de processos de modelos de capacidade de processo
<b>no contexto</b>	de disciplinas curtas de pós-graduação em melhoria de processo

## 7.1 Análise de modelos relevantes, gerações de arquitetura e PRO2PI

Esta seção descreve considerações sobre as gerações de arquiteturas e elementos de PCDE e PRO2PI e análises de dezesseis dos modelos mais relevantes segundo estas gerações e seus três conceitos básicos (área de processo, nível de capacidade de processo e perfil de capacidade de processo), incluindo considerações sobre os elementos estruturais do modelo PRO2PI-MODEL e quatro modelos de capacidade de processo.

Cada geração de arquitetura de modelos de capacidade de processo tem aumentado a flexibilidade para definição e utilização dos modelos de capacidade como referência para avaliação e melhoria de processo. É um fenômeno semelhante ao que ocorre com as linguagens de programação, onde a evolução é em direção ao que Lieberherr et al. [1994] denominou de evolução para “*ever late binding*” (acoplamento cada vez mais tardio). Com a flexibilidade das linguagens, a determinação do endereço é cada vez mais transferida para o tempo de compilação e execução. No caso dos modelos de capacidade, o perfil a ser utilizado como referência pode ser definido pela organização e não é apenas fixado pelo modelo.

Esta evolução também pode ser caracterizada pelo compromisso entre estabilidade e variabilidade identificado por Coplien e Schmidt [1995] para frameworks orientados a objetos:

“Desenvolvedores de frameworks são confrontados com várias soluções de compromisso desafiadores de projeto. Um dos mais cruciais é determinar quais componentes do framework deverão ser variáveis e quais deverão ser estáveis. Variações insuficientes torna muito difícil para os usuários a customização dos componentes. Isto resulta em frameworks que não conseguem acomodar os requisitos funcionais de várias aplicações. Por outro lado, uma estabilidade insuficiente torna difícil a compreensão” [Coplien e Schmidt 1995]. Com isto, ainda segundo Coplien e Schmidt [1995] o reuso efetivo de um modelo depende do entendimento, identificação e representação dos pontos de pressão de estabilidade e variabilidade em um domínio da aplicação. Isto determina o que deve ser fixo e o que deve ser variável em um modelo.

Variações insuficientes, como a predefinição das áreas de processo em cada nível do SW-CMM, dificultam a adequação do modelo às características particulares de cada organização. Por outro lado, estabilidade insuficiente dificulta o entendimento, a confiança na qualidade da escolha e a troca de experiências de utilização. Os modelos contínuos precisam ser complementados com métodos para orientar a escolha dos processos, como a abordagem PRO2PI apresentada, para atingir o balanço ideal entre estabilidade e flexibilidade.

Uma regra que suporta o argumento de que realmente existe uma evolução, é que: um modelo da geração G-1 sempre pode ser expresso como um exemplo de um modelo da geração G. Ou seja, qualquer modelo da primeira geração pode ser expresso como um exemplo de um modelo da segunda geração, e qualquer modelo da segunda geração pode ser expresso como um exemplo de um modelo da terceira geração. Consequentemente, também qualquer modelo da primeira geração pode ser expresso como um exemplo de um modelo da terceira geração. Um modelo da segunda geração pode ser definido como um exemplo de utilização de um modelo desta terceira geração. O conjunto de processos é uma escolha entre os vários processos definidos e pode ter processos específicos.

As diferenças e similaridades entre os modelos mais relevantes podem ser descritas em função da variação de estabilidade e flexibilidade de três conceitos básicos: área de processo, nível de capacidade de processo e perfil de capacidade de processo. A Figura 81 relaciona três frameworks, dezesseis modelos e um posicionamento deles nas três gerações propostas. Esses frameworks e modelos foram selecionados por serem considerados mais representativos.

Na Figura 81, o termo ‘e(<nome-de-modelo>)’ significa que esse modelo é um exemplo representativo dos modelos desse framework. O termo ‘m\*(<nome-de-modelo>)’ também significa que esse modelo é um exemplo representativo dos modelos desse framework e cada modelo é disponibilizado com uma representação estagiada (arquitetura estagiada fixa) e com outra

representação contínua (arquitetura contínua fechada). O termo ‘v\*(<nome-de-modelo>)’ significa que esse modelo não é um modelo de capacidade de processo, mas pode ser visto com um, usando as características da terceira geração (arquitetura contínua aberta). Finalmente o termo ‘u(PRO2PI)’ significa as utilizações de PRO2PI.

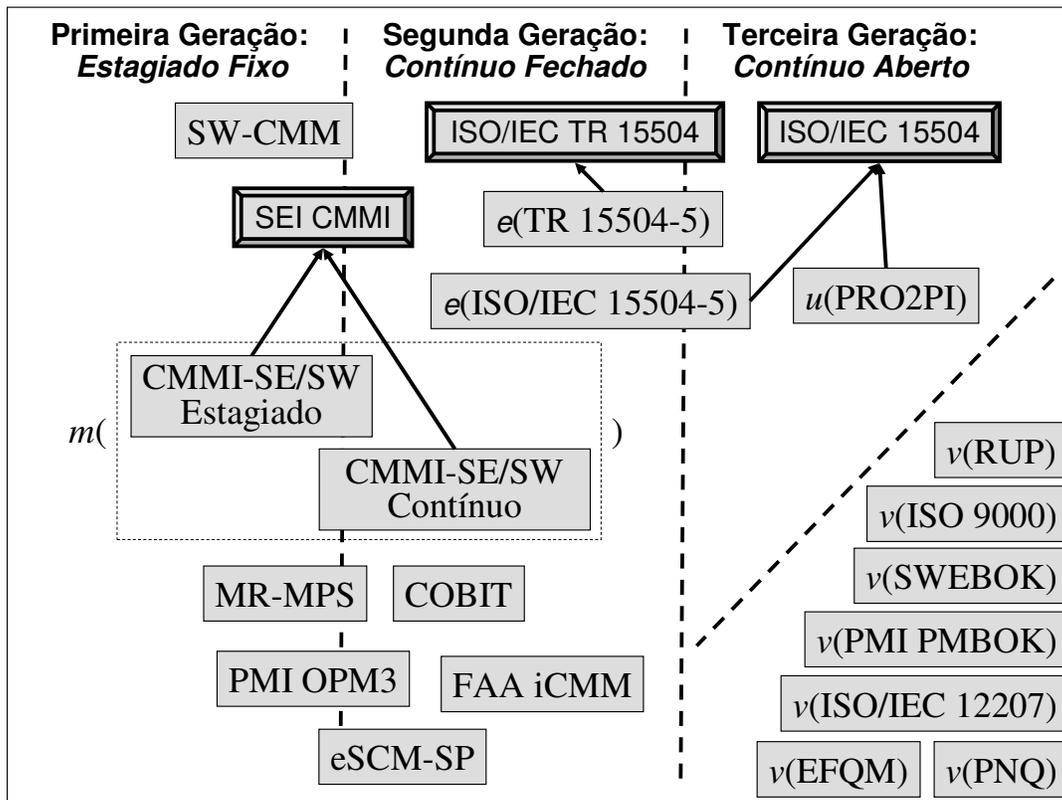


Figura 81 – Posicionamento dos modelos relevantes nas gerações de modelos

Modelos e frameworks com partes em diferentes gerações indica que esse modelo ou framework pode ser considerado com tendo aspectos das duas gerações. O quanto ele é colocado em cada geração, indica aproximadamente o quanto ele pode ser considerado daquela geração.

A Tabela 23 descreve uma caracterização dos dezesseis principais modelos descritos no capítulo 2 em termos de como os três principais conceitos da engenharia de processo (área de processo na coluna AP, níveis de capacidade de processo na coluna NCP e perfil de capacidade de processo na coluna PCP) e o conceito de processo (na coluna Descrição de processo) são considerados em cada modelo, e também o número de entidades que podem ser consideradas como áreas de processo (na coluna #PA).

Tabela 23 - Modelos e conceitos da engenharia de processo

Entidade, modelo e versão	Tipo	AP	#PA	NCP	PCP	Descrição de processo
01: SEI SW-CMM v1.1	EF	P #	22	P #	L #	N 0
02: SEI CMMI-SE/SW E* v1.1	EF	L #	18	F #	F #	N 0
03: SEI CMMI-SE/SW C* v1.1	CF	L #	18	F #	F &	N 0
04: ISO/IEC 15504-5	CA	F @	48	F #	F &	N 0
05: FAA iCMM v2.0	CF	F #	23	F #	F @	N 0
06: ISO/IEC 12207 Amd2	OR	F #	49	N 0	N 0	P 0
07: ISO 9001:2000	OR	P #	23	P #	P #	N 0
08: PMI PMBOK Third Edition	OR	P #	9	N 0	N 0	P #
09: PMI OPM3 v1.0	CF	F #	3	F #	F @	P #
10: ITsqc eSCM-SP v2	CF	F #	10	F #	F &	P #
11: MPS-BR MR-MPS v1.0	EF	L #	19	L #	L #	N 0
12: IEEE SWEBOOK 2004	OR	P #	10	N 0	N 0	N 0
13: ITGI COBIT v4.0	CF	L #	34	L #	F &	N 0
14: EFQM	OR	P #	6	P #	P #	N 0
15: FPNQ PNQ	OR	P #	8	N 0	P #	N 0
16: RUP	OR	P #	9	N 0	N 0	F @

Na Tabela 23 os modelos são classificados na coluna ‘Tipo’ pela geração de arquitetura predominante do modelo: (EF) para estagiada fechada, (CF) para contínua fechada, (CA) para contínua aberta e (OR) para outros tipos de modelo de referência. Na Tabela 23 o termo ‘E\*’ indica a representação estagiada e o termo ‘C\*’ indica a representação contínua do modelo CMMI-SE/SW.

Para a caracterização dos quatro conceitos nos modelos é utilizada na Tabela 23 a escala de pontuação de atributos de processo com uma adaptação do significado. Desta forma a escala é composta por: (F) quando o conceito é totalmente (“*fully*”) considerado, (L) quando o conceito é largamente considerado, (P) quando o conceito é parcialmente considerado e (N) quando o conceito não é considerado. A esta escala é acrescentado os símbolos para designar se os elementos do conceito são (#) fixos, (&) variáveis, (@) variáveis com um conjunto proposto ou (0) quando o conceito não é considerado. A seguir esta caracterização é comentada para cada doze dos dezesseis modelos da Tabela 23.

**O modelo SEI SW-CMM v1.1:**

- pode ser considerado um modelo de capacidade de processo da primeira geração (estagiado fixo);
- contém o conceito de área chave de processo que corresponde parcialmente ao conceito de área de processo, pois falta a independência ao nível de capacidade de processo, pois cada uma das 18 áreas chave de processo é representada para um determinado nível de capacidade;
- contém o conceito de nível de maturidade que corresponde parcialmente ao conceito de nível de capacidade de processo, pois apesar de ser semelhante, é associado ao processo de desenvolvimento e manutenção de software e não a cada área de processo;
- define quatro níveis de maturidade fixos (níveis 2, 3, 4 e 5), que correspondem largamente a quatro perfis de capacidade de processo fixos, pois falta a caracterização como perfil, e os níveis 4 e 5 podem ser considerados como simulações da arquitetura contínua fechada nos níveis 4 e 5 de capacidade de processo; e
- não define características de descrição de processo pois não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

**O modelo CMMI-SE/SW v.1.1, representação estagiada:**

- pode ser considerado um modelo de capacidade de processo da primeira geração (estagiado fixo);
- define o conceito de área de processo que corresponde largamente ao conceito de área de processo, faltando apenas uma maior independência em relação aos níveis de capacidade de processo pois cada uma das 22 áreas de processo é representada com uma tendência para um determinado nível de capacidade;
- define o conceito de nível de capacidade de processo que corresponde completamente ao conceito de nível de capacidade de processo;
- define quatro níveis de maturidade fixos (níveis 2, 3, 4 e 5), que correspondem completamente a quatro perfis de capacidade de processo fixos embora os níveis 4 e 5 de maturidade possam ser considerados como simulações da arquitetura contínua fechada em relação aos níveis 4 e 5 de capacidade de processo; e
- não define características de descrição de processo pois não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

**O modelo CMMI-SE/SW v.1.1, representação contínua:**

- pode ser considerado um modelo de capacidade de processo da segunda geração (contínua fixo):

- define o conceito de área de processo, com mesma denominação, com 22 áreas de processos, mas existe uma inclusão de características de níveis de capacidade avançados em algumas delas;
- define o conceito de nível de capacidade, com mesma denominação, com seis níveis de capacidade;
- define o conceito de perfil de capacidade de processo, com possibilidade de definição como combinação de qualquer processo com qualquer nível de capacidade, mas certas áreas de processo já induzem a certos níveis de capacidade, e
- não define características de descrição de processo pois é não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

**O modelo da Norma ISO/IEC 15504-5:**

- define o conceito de área de processo, denominado de processo, com 48 processos e possibilidade de definição de novos processos, podendo ser então considerado como um modelo de capacidade de processo da terceira geração com um exemplo na segunda geração;
- define o conceito de nível de capacidade, com mesma denominação, com seis níveis de capacidade;
- define o conceito de perfil de capacidade de processo, com possibilidade de definição como combinação de qualquer processo com qualquer nível de capacidade,
- não define características de descrição de processo pois é não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

**O modelo FAA iCMM v2.0:**

- pode ser considerado um modelo de capacidade de processo da segunda geração (continua fixo);
- define o conceito de área de processo, com mesma denominação, com 23 áreas de processos, mas existe uma inclusão de características de níveis de capacidade avançados em algumas delas;
- define o conceito de nível de capacidade, com mesma denominação, com seis níveis de capacidade;
- define o conceito de perfil de capacidade de processo, com possibilidade de definição como combinação de qualquer processo com qualquer nível de capacidade; e
- não define características de descrição de processo pois é não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

**O modelo da Norma ISO/IEC 12207 Amd2:**

- define o conceito de área de processo, denominado de processo ou componente de processo, com um total de 49,
- não define o conceito de nível de capacidade e em consequência também não existe o conceito de perfis de capacidade de processo,
- não define características de descrição de processo pois é não é um modelo de processo e sim um modelo de referência para processos e como tal orienta o estabelecimento dos processos de uma organização, embora exista uma tendência de relacionar cada processo da 12207 a um processo na organização.

**O modelo da Norma ISO 9001:2000:**

- não define o conceito de perfil de capacidade de processo mas os requisitos definidos representam um exemplo de um perfil de capacidade de processo fixo,
- não define o conceito de nível de capacidade, nem de áreas de processo, mas os requisitos definidos pode ser representados como uma combinação de áreas de processo com níveis de capacidade, conforme realizado no projeto SPICE 9000 (S9K) [Dorling 2002], e
- não define características de descrição de processo pois é não é um modelo de processo e sim um modelo de referência para processos e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

**O modelo do PMI PMBOK Third edition:**

- define o conceito de área de processo, denominado processos , com 39 processos, organizados em 9 áreas de conhecimento,
- não define o conceito de nível de capacidade e em consequência também não existe o conceito de perfil de capacidade de processo,
- não define características de descrição de processo pois é não é um modelo de processo e sim um modelo de referência de processo e como tal orienta o estabelecimento dos processos de uma organização, embora defina cinco grupos de processos e o relacionamento desses grupos como um processo.

**O modelo do PMI OPM3 v.1.0:**

- pode ser considerado um modelo de capacidade de processo com aspectos da primeira geração (estagiada fixa) e da segunda geração (continua fixo),
- os três domínios representam as áreas de processo,

- não existe o conceito de nível de capacidade, mas as práticas de cada domínio em cada nível de maturidade representam a aplicação de um nível de capacidade para esse domínio,
- não existe o conceito de perfil de capacidade de processo, mas os quatro níveis de maturidade, com as práticas de cada domínio naquele nível, representam exemplos de perfil de capacidade de processo, e
- não define características de descrição de processo pois não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização, embora defina cinco grupos de processos e o relacionamento desses grupos como um processo para cada combinação de nível de maturidade e domínio.

#### **O modelo MPS-BR MR-MPS v1.0:**

- pode ser considerado um modelo de capacidade de processo da primeira geração (estagiado fixo);
- define o conceito de processo que corresponde largamente ao conceito de área de processo, faltando apenas uma maior independência em relação aos níveis de capacidade de processo pois cada um dos 22 processos é representado com uma tendência para um determinado nível de capacidade;
- define o conceito de nível de capacidade de processo que corresponde completamente ao conceito de nível de capacidade de processo;
- define sete níveis de maturidade fixos (níveis G a A), que correspondem completamente a sete perfis de capacidade de processo fixos embora os níveis B e A de maturidade possam ser considerados como simulações da arquitetura contínua fechada em relação aos níveis 4 e 5 de capacidade de processo; e
- não define características de descrição de processo pois não é um modelo de processo e sim um modelo de capacidade de processo e como tal orienta o estabelecimento, avaliação e melhoria dos processos de uma organização.

#### **O modelo do IEEE SWEBOOK:**

- define o conceito de área de processo, denominado área de conhecimento, com 9 áreas de conhecimento, que são compostas por processos que representam exemplos de sub-áreas de processo,
- não existe o conceito de nível de capacidade e em consequência também não existe o conceito de perfil de capacidade de processo,
- não define o conceito de processo, e não é voltado para definição de processos em uma organização.



Cl_PracticesSystem	F	N	F	N	F	N	F	N
Cl_ProcessAreaCapabilityProfile	F	F	F	F	F	F	F	F
Cl_ProcessCapabilityProfile	P	F	F	F	F	F	F	F
Cl_ProcessCapabilityProfilesHierarchy	P	N	P	L	P	L	P	L

## 7.2 Aplicações de PRO2PI

Esta seção descreve uma serie de aplicações com as várias versões de PRO2PI.

### 7.2.1 Melhoria de processo de software na Empresa E1 [1999-2002]

O programa de melhoria na empresa Senior Sistemas, denominada nessa tese de Empresa E1, foi iniciado em 1999 e acompanhado até 2002 [Salviano et al. 1999a, Nicoletti e Salviano 2003, Salviano 2003]. O programa foi orientado pelos objetivos, contexto e estratégia de negócio da empresa e teve como referência inicial ISO/IEC TR 15504-5 [ISO/IEC TR 15504 1998]. O programa teve também como referência adicional a Norma ISO 9001:2000, que especifica requisitos para um sistema de gestão da qualidade. Os resultados dos primeiros três anos desse programa são descritos principalmente pela comparação entre duas avaliações de processo: uma realizada em 1999 e outra em 2002.

A Empresa E1, fundada em 1988 com sede na cidade de Blumenau-SC, é uma empresa nacional que, no início do programa de melhoria, desenvolvia, evoluía e comercializava quatro produtos: Produto P1, para administração de recursos humanos; Produto P2, para gestão empresarial; Produto P3, para administração de agências de viagens; e Produto P4, para controle de acesso. A Empresa E1 contava com cerca de 200 funcionários em Blumenau, sendo que aproximadamente 50% deles estão na área de desenvolvimento de software.

A Empresa E1 pode ser considerada, para avaliação de processos, como composta por cinco unidades organizacionais: uma para os Produtos P1 e P4, outra para o Produto P2, outra para o Produto P3, outra para o desenvolvimento comum aos produtos (chamada de Tecnologia) e outra para a empresa como um todo.

Como forma de estruturar seus esforços na melhoria de processos de software, em 1999 a Empresa E1 realizou uma avaliação de processos baseada no ISO/IEC TR 15504. Foram selecionados cinco processos para serem avaliados: CUS.4.2 suporte ao cliente, SUP.3 garantia da qualidade, MAN.2 gerenciamento de projeto, ORG.1 alinhamento organizacional e ORG.2.1 estabelecimento de

---

processo. Esta avaliação teve como objetivo principal prover um dos subsídios para a construção de um plano de melhoria da empresa, baseado nesses cinco processos. Todo o trabalho de escolha dos processos, avaliação e planejamento da melhoria teve outro objetivo importante de capacitar pessoas da empresa em melhoria de processo de software de tal maneira que elas pudessem conduzir a execução do plano de melhoria.

O plano do projeto de melhoria incluía várias atividades relacionadas a esses cinco processos. Estas atividades foram planejadas para um período de 18 meses. A execução do projeto de melhoria foi iniciada em setembro de 1999. Em janeiro de 2000 foi realizado um balanço dos primeiros cinco meses desta execução e na análise dos motivos do não atendimento de alguns objetivos, percebeu-se a dificuldade da empresa em estabelecer processos e garantir a qualidade de produtos. Foi evidenciada então a necessidade de enfatizar a criação de uma estrutura permanente para suportar a definição, estabelecimento, controle e melhoria contínua dos processos, além de estruturar as atividades para garantia da qualidade. A solução definida foi a implantação de um Sistema de Gestão da Qualidade, baseado na norma ISO 9001:2000. Com isto o plano do projeto de melhoria foi revisto e orientado para a implantação do Sistema de Gestão da Qualidade, com contratação de uma consultoria externa para apoiar esta implantação.

Esta implantação ocorreu no período de fevereiro de 2000 à março de 2001. Todos os objetivos foram completamente atendidos ao final da implantação do SGQ, que foi certificado em abril de 2001. Após esta certificação, o foco passou a ser o estabelecimento definitivo de uma área e funções específicas para a qualidade na empresa, responsável pelo SGQ.

Em maio de 2002, pouco mais de um ano após a certificação ISO 9001:2000, foi realizada uma nova avaliação de processos com a 15504. A avaliação dos processos selecionados foi realizada em 2002 por dois avaliadores, um externo e outro interno à empresa, em três fases sequenciais: (1) preparação da avaliação, (2) obtenção e análise de dados e determinação dos resultados, e (3) produção do relatório de avaliação. Na fase de preparação da avaliação foram selecionados os processos e respectivos níveis de capacidade a serem avaliados, planejada a avaliação e aplicados questionários sobre os processos a pessoas selecionadas da empresa.

Foram selecionados sete processos da 15504 para serem avaliados até o nível 3 de capacidade. Como o objetivo principal da avaliação foi comparar o estado atual com o estado avaliado em 1999, foram selecionados os mesmos cinco processos avaliados em 1999. Esses cinco processos foram escolhidos em 1999 porque eram aqueles cuja melhoria causaria um maior impacto na empresa. Os dois processos adicionais foram selecionados para a avaliação de 2002 porque havia a percepção que

houve uma grande melhora no processo de medição e que seria bom incluir um processo da categoria engenharia. Foram então selecionados os processos ENG.1.2 Análise de Requisitos e ORG.5 Medição.

Os três processos organizacionais foram avaliados para a empresa como um todo. Os quatro processos restantes foram avaliados para apenas duas das outras cinco unidades organizacionais da empresa. A decisão de avaliar apenas duas unidades foi devido a restrições do esforço de avaliação (apenas três dias). A escolha de quais duas unidades seriam avaliadas foi devido à representatividade destas unidades na empresa.

O resultado das avaliações realizadas em 1999 e em 2002 estão descritos na Figura 82.

Resultado das avaliações, com pontuações dos atributos de processo e níveis de capacidade dos processos em 1999 e 2002				
	1.1	2.1	2.2	NC
<b>Alin.Org.1999</b>	P	P	N	0
<b>2002</b>	F	L	L	2
<b>Est.Proc.1999</b>	P	N	N	0
<b>2002</b>	F	L	F	2
<b>Medição1999</b>	X	X	X	X
<b>2002</b>	F	L	F	2
<b>Ger.Proj.1999</b>	P	N	N	0
<b>2002</b>	F	P	L	1
<b>Gar.Qual.1999</b>	P	N	N	0
<b>2002</b>	F	P	L	1
<b>Sup.Clie.1999</b>	P	N	N	0
<b>2002</b>	F	L	F	2
<b>Ana.Req.1999</b>	X	X	X	X
<b>2002</b>	F	L	L	2

Balanco da Melhoria (média do resultado)				
	1.1	2.1	2.2	NC
<b>1999:</b>	P	N	N	0
<b>2002:</b>	F	L	L	2

**legenda:**  
 N: *Not achieved*  
 P: *Partially achieved*  
 L: *Largely achieved*  
 F: *Fully achieved*  
 X: *Not rated*

Figura 82 – Resultados da avaliação na Empresa E1 em 1999 e 2002

A Figura 82 mostra que os três processos organizacionais avaliados estão no nível 2 de capacidade. Em 1999, os dois processos avaliados estavam no nível 0 e o processo de medição certamente também estava no nível 0, pois não havia a prática de medições na empresa. Houve uma evolução de processos improvisados para processos gerenciados.

A Figura 82 mostra que três dos processos avaliados para a unidade organizacional responsável por um dos quatro produtos da empresa estão no nível 2 de capacidade, enquanto que o quarto processo está no nível 1. Nesse caso também houve melhoria em relação a 1999. A Figura 82 também mostra que dois dos processos avaliados para a unidade organizacional responsável por um dos quatro

produtos da empresa estão no nível 2 de capacidade, enquanto que os outros dois estão no nível 1. Nesse caso também houve melhoria em relação a 1999.

O resultado indica uma melhoria em termos de capacidade de processo. Em geral houve uma melhora do nível 0 para o nível 2, com alguns casos de melhoria para o nível 1. Existem evidências que houve uma melhoria compatível no desempenho da empresa. A existência de uma cultura de gerenciamento é evidente na empresa e foi relatada por várias pessoas. A prática de utilizar indicadores para acompanhar as atividades e subsidiar a tomada de decisões também existe na empresa. Existem atualmente doze indicadores relacionados ao desenvolvimento e quatro relacionados ao suporte ao cliente. A prática de pesquisas de satisfação dos clientes, realizadas nos dois últimos anos e previstas para serem realizadas anualmente, permite um melhor entendimento da empresa. Praticamente tudo é registrado e analisado, incluindo reclamações dos clientes e requisitos para evolução dos produtos. O sentimento generalizado na empresa é que houve uma melhoria nos últimos três anos devido à melhoria de processo e que esta melhoria é apenas o começo de novas melhorias.

Os resultados dos três primeiros anos do programa de melhoria, identificados pela evolução do nível de capacidade de processos importantes de 0 (ad-hoc, improvisado, inconstante) para os níveis 1 (o processo geralmente atinge seus objetivos) e 2 (o processo atinge seus objetivos de forma gerenciada e com melhor qualidade), indicam uma melhoria da empresa. Esses resultados também mostraram que é possível combinar os dois modelos utilizados de forma harmônica: ISO/IEC TR 15504-5 e ISO 9000:2000. Pela abrangência e resultados dos processos avaliados, estima-se que a empresa esteja pelo menos próxima do nível 2 de maturidade do SW-CMM.

No trabalho de planejamento da melhoria realizado em 1999, foram identificadas evidências do sucesso do início do trabalho, mas já existia a consciência de que o sucesso seria atingido apenas com a melhoria da empresa. Três anos depois, os resultados da nova avaliação são evidências que estas melhorias foram atingidas.

A experiência na Empresa E1 mostrou na prática uma utilização do que veio a ser a metodologia PRO2PI. O programa foi iniciado baseado em uma hierarquia com dois PCPs: Um baseado na situação atual da época (1999) e outro na meta de melhoria (para 2001). Durante a execução do programa, em função do aprendizado e principalmente de um par necessidade/oportunidade, a hierarquia de PCPs foi alterada. A necessidade foi de conhecimento e experiência para o estabelecimento e avaliação de processos. A oportunidade foi utilizar um conhecimento e experiência existente na localidade e com custo acessível de processo, mas baseado na ISO 90001. Então, como o PCP identificado tinha relacionamento com a ISO 9001, foi decidido ampliar o PCP para incorporar a ISO 9001. Desta forma

foi ajustado o objetivo do programa. Em 2002 o processo foi reavaliado, com ampliação do PCP. Segundo a abordagem PRO2PI, esta experiência pode ser relacionada a quatro perfis de capacidade de processo [Tabela 25].

Tabela 25 – PCPs da melhoria de processo

Ano		
1999	PCP Alvo:	[ {AlinOrg, EstProc, GerProj, GarQual, SupClie} : CL2 ]
1999	PCP Atual:	[ {AlinOrg, EstProc, GerProj, GarQual, SupClie} : CL0 ]
2000	PCP Alvo:	[{AlinOrg, EstProc, GerProj, GarQual, SupClie, ISO9001}: CL2]
2002	PCP Atual:	[ {GerProj, GarQual} : CL1, {AlinOrg, EstProc, Medic, SupClie, AnaReq} : CL2 ]

Na Tabela 25, os termos GerProj, GarQual, AlinOrg, EstProc, Medição, SupClie, e AnaReq identificam sete processos da ISO/IEC TR 15504-5 e o termo ISO9001 indica a ISO 9001 vista como uma área de processo.

### 7.2.2 Melhoria de processo de software na Empresa E2 [2002-2003]

O programa de melhoria na empresa Ampla Consultoria em Informação ([www.amplaconsultoria.com.br](http://www.amplaconsultoria.com.br)), denominada de Empresa E2 nessa tese, foi iniciado em 2002 e acompanhada até 2004 [Silva et al. 2003a, Silva et al. 2003b, Silva et al. 2004].

A Empresa E2 é uma pequena empresa de desenvolvimento de projetos de software. Fundada em 1995 e instalada em Campinas, SP, nas proximidades da Universidade Estadual de Campinas. A Empresa E2 tem, desde o início de suas atividades, como foco principal o desenvolvimento de projetos de software. Esta área representa cerca de 90% de seu faturamento mensal. Os clientes da empresa são, em sua maioria, grandes empresas, multinacionais em muitos casos. Os projetos têm entre 2 e 6 meses de duração e envolvem entre 2 e 4 profissionais.

A empresa resolveu realizar um projeto de melhoria de seu processo de desenvolvimento porque seu processo possuía pouca transparência. Não obstante, grande parte de seus projetos era finalizada com sucesso e possuía qualidade. A maioria entrava em produção com algum atraso no cronograma e alguma variação nos custos, esse último nem sempre era negociável e os prejuízos eram absorvidos pela empresa.

Um dos principais desafios desse projeto está relacionado com a adaptação da abordagem de melhoria de processo para o contexto das pequenas empresas de software. Nesse caso foi necessário identificar os elementos essenciais da abordagem e utilizá-los de forma eficiente e eficaz, compatíveis com as características de agilidade das pequenas empresas de software.

O projeto foi iniciado em janeiro de 2002. O projeto de melhoria de processo que está sendo realizado na Empresa E2, tem quatro referências conceituais básicas: a ISO/IEC TR 15504-5, a abordagem para melhoria de processo AMP1, o ProGer [Rouiller 2001] e uma metodologia para testes definida pelo CenPRA. O ProGer é um modelo de processo para gerenciamento de projetos de software, baseado no PMBOK:2000 e na ISO/IEC TR 15504-5, para organizações de desenvolvimento de software de pequeno porte. A metodologia de testes é uma metodologia para a introdução ou melhoria do processo de teste de software em empresas, capacitando-as a desenvolver produtos de melhor qualidade.

As três primeiras atividades e o início da quarta atividade da abordagem AMP1 foram realizadas entre junho e dezembro de 2002. De janeiro a setembro de 2003 foi continuada a implantação da melhoria, que inclui revisões e utilizações parciais nos projetos da Empresa E2.

A primeira fase foi composta pela elicitação do negócio da empresa, estratégias e objetivos. A escolha da ISO/IEC TR 15504-5 se deu devido à flexibilidade para adaptações às necessidades da Empresa E2. Decidiu-se utilizar cinco processos da ISO/IEC TR 15504-5 para o nível 2 de capacidade. Os processos escolhidos foram: CUS.2 *fornecimento*, CUS.3 *elicitação de requisitos*, MAN.2 *gerência de projeto*, ENG.1.6 *teste de software* e ORG.5 *medição*. Esses processos foram selecionados de forma a cobrir as necessidades de melhoria de processo da Empresa E2 e para orientar a melhoria do processo da fábrica de software [Figura 83].

Foi realizada uma avaliação das práticas correntes, seguindo os requisitos da ISO/IEC TR 15504 para avaliação de processo, em relação a esses cinco processos até o nível 3 de capacidade. Os resultados indicaram nível 2 de capacidade para o processo ORG.5 *Medição* e nível 1 para os outros. Após a avaliação um plano simples de melhoria foi estabelecido visando duas frentes de trabalho: o estabelecimento do processo da fábrica de software da empresa e o estabelecimento do processo de teste de software.

A definição do processo da fábrica de software da empresa iniciou-se com um mapeamento das práticas utilizadas, algumas documentadas, mas nem sempre repetidas e outras não documentadas. A empresa possuía poucos artefatos mas nenhum processo para seguir, o que dificultava a evolução dos artefatos utilizados e a criação (ou eliminação) de outros. O processo foi definido e criado como um fluxo com seis fases, cada uma com atividades, regras do processo, papéis e artefatos que deveriam ser produzidos. As fases são: *Prospecção*, *Proposta*, *Execução*, *Garantia* e *Encerramento*. Os principais artefatos gerados são: atas de reunião, proposta de projeto de software, documento de especificação,

plano de projeto, plano de testes e relatório de aceitação. Para cada um foi definido um modelo que deve ser utilizado pelos líderes de projeto e analistas.

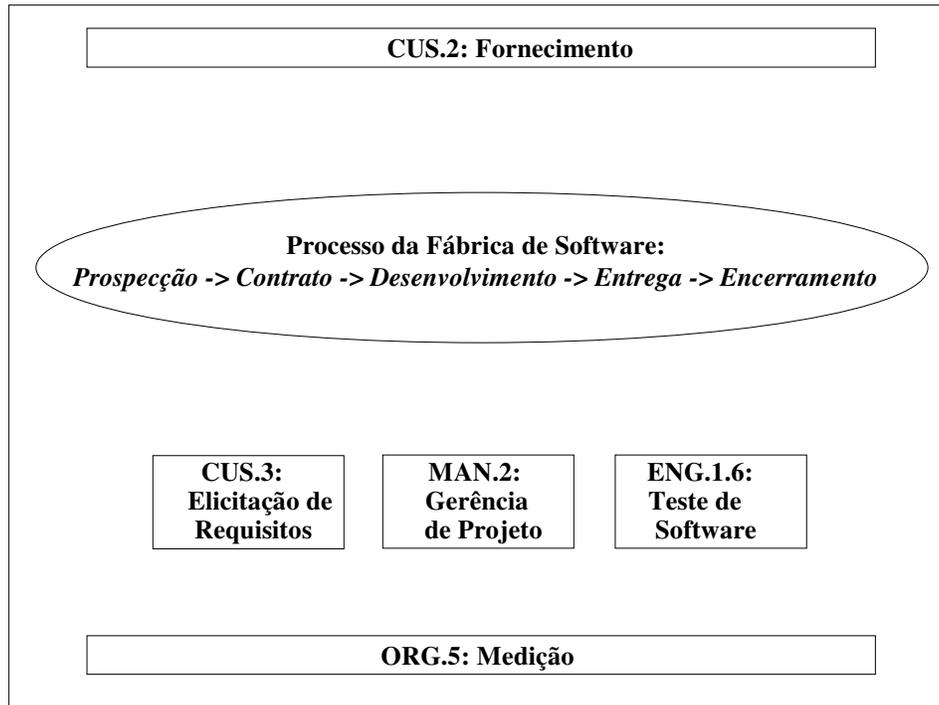


Figura 83 – Processo da fábrica de software e processos escolhidos

O programa foi iniciado baseado no perfil de capacidade de processo

[ { FornSw, ElicReq, GerProj, TestSw } : CL2, Medic : CL3 ].

Ao contrário da maioria das micro e pequenas empresas, a Empresa E2 já tinha uma cultura de utilizações de medições de forma sistemática para apoio à gestão. Por isto o processo de medição foi utilizado.

### 7.2.3 Estabelecimento de perfil de capacidade de processo [2000-2005]

As três primeiras fases da abordagem AMP1 (início e definição de metas, avaliação e planejamento da melhoria.), e em alguns casos, o método MEP1, e as várias versões de PRO2PI-WORK, foram utilizados em nove projetos. Em cada um deles foi definido um perfil de capacidade de processo diferente orientado pelas características da organização.

O primeiro projeto foi realizado em 2000 em uma organização orientada pelo desenvolvimento de projetos em um domínio específico, no caso sistemas para apoio administrativo de prefeituras e secretarias de governos estaduais. Um dos objetivos da organização era desenvolver um sistema nesta

área como um produto a ser comercializado em várias prefeituras. Nesse projeto foi utilizado o modelo do *Balanced Scorecard* para representar os objetivos estratégicos e estratégia da organização, conforme descrito na Figura 84.

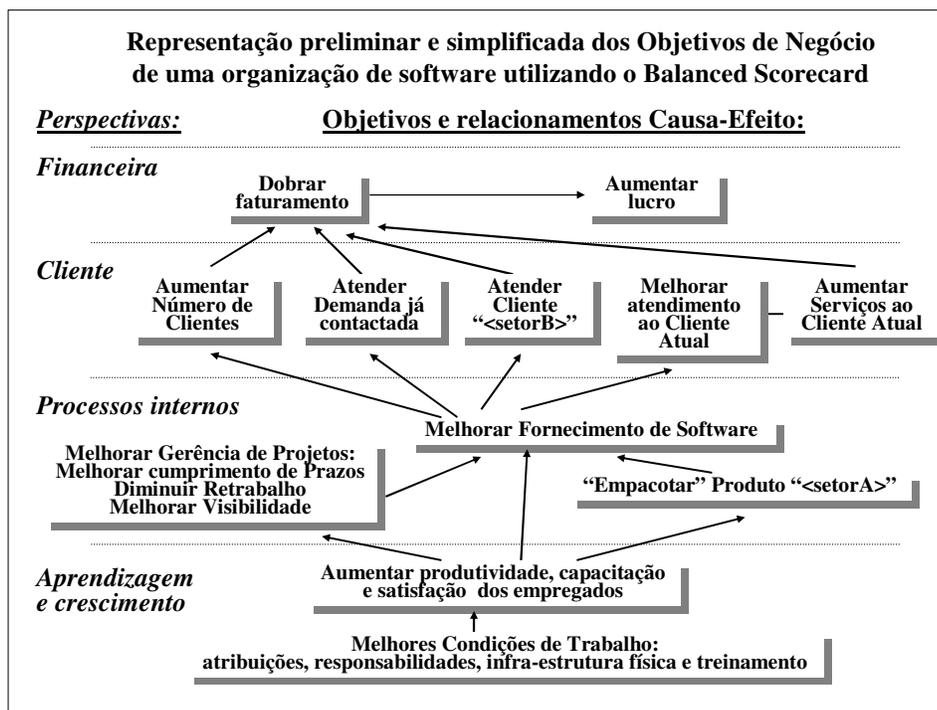


Figura 84 – Objetivos e estratégia de negócios em Balanced Scorecard

Com base na representação com o Balanced Scorecard e outros subsídios, o perfil de capacidade de processo estabelecido foi o seguinte:

[ FornSw:CL2, GerProj:CL2, { GarQual, ElicReq, TestSw, Docum, GerConf } : CL1 ]

onde os termos FornSw, GerProj, GarQual, ElicReq, TestSw, Docum e GerConf representam respectivamente os processos de fornecimento, gerência de projeto, garantia da qualidade, alinhamento organizacional, elicitação de requisitos, teste de software, documentação e gerência de configuração, do modelo ISO/IEC TR 15504-5.

O segundo projeto, realizado nos anos de 2000 e 2001, foi iniciado com o objetivo de atingir o nível 2 de maturidade do modelo SW-CMM. A unidade organizacional era um departamento de uma grande entidade estatal relacionada com a fiscalização de receitas.

Esse departamento era basicamente responsável pelo entendimento das necessidades de sistemas de informação para uso na atividade fim e para a população em geral, pela especificação dos sistemas de informação e pela aquisição do desenvolvimento desses sistemas.

Desta forma, foi percebido no início do projeto que o nível do SW-CMM não era o perfil mais adequado. Foi definido como perfil de capacidade de processo o seguinte:

```
[ { RM, SSM : CL2,
  { FornSw, PrepAqui, AceitCl, GerProj,
    Docum, GerConf, GarQual, EstProc, Medic } : CL1 } ]
```

onde RM e SSM representam as áreas chave de processo de gerência de requisitos e gerência de subcontratação de software do SW-CMM, e FornSw, PrepAqui, AceitCl, GerProj, Docum, GerConf, GarQual, EstProc e Medic representam os processos de fornecimento (*supply*), preparação para aquisição (*acquisition preparation*), aceitação do cliente (*customer acceptance*), gerência de projetos (*project management*), documentação (*documentation*), gerência de configuração (*configuration management*), garantia da qualidade (*quality assurance*), estabelecimento de processo (*process establishment*), medições (*measurement*), do modelo ISO/IEC TR 15504-5 [1998].

A Figura 85 ilustra os processos de gestão de contrato e acompanhamento de contrato definidos para a organização como resultado do projeto de melhoria [Salviano et al. 2001].

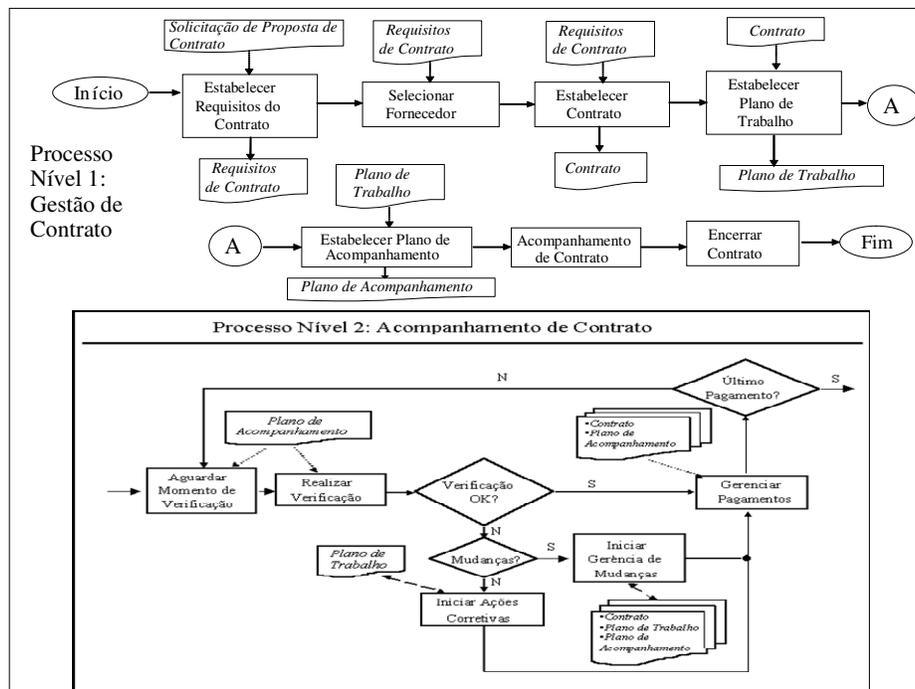


Figura 85 – Processo para Gestão e acompanhamento de contrato

---

O terceiro projeto foi realizado em 2001 em uma unidade organizacional responsável pelo desenvolvimento e operação de sistemas de software para um ministério do governo federal. O perfil de capacidade de processo definido foi o seguinte:

[ { PrepAqui, SelFor, MonFor, AceiCl, DesSw, Docum, GerRH } : CL2 ]

onde os termos PrepAqui, SelFor, MonFor, AceiCl, DesSw, Docum e GerRH, representam respectivamente os processos de preparação para aquisição, seleção de fornecedor, monitoramento de fornecedor, aceite do cliente, desenvolvimento de software, documentação e gerência de recursos humanos, todos da ISO/IEC TR 15504-5, e no nível 2 de capacidade de processo.

O terceiro projeto foi realizado em 2001 em uma unidade organizacional responsável pelo desenvolvimento e operação de sistemas de software para um ministério do governo federal. O perfil de capacidade de processo definido foi o seguinte:

[ { PrepAqui, SelFor, MonFor, AceiCl, DesSw, Docum, GerRH } : CL2 ]

onde os termos PrepAqui, SelFor, MonFor, AceiCl, DesSw, Docum e GerRH, representam respectivamente os processos de preparação para aquisição, seleção de fornecedor, monitoramento de fornecedor, aceite do cliente, desenvolvimento de software, documentação e gerência de recursos humanos, todos da ISO/IEC TR 15504-5, e no nível 2 de capacidade de processo.

O quarto projeto, realizado no final de 2002, foi em uma empresa privada, já certificada na ISO 9001:2000, que desenvolve projetos de software e desenvolve e comercializa um produto para gerenciamento de documentos. O objetivo foi avaliar o processo em relação a áreas de processo do modelo CMMI-SE/SW com o objetivo de identificar ações de melhoria. Como o processo corrente já estava bem estruturado, foi decidido focar nas áreas de processo da categoria de engenharia no nível 3 de capacidade. O perfil de capacidade de processo definido foi o seguinte:

[ { ISO9001, REQM, RD, TS, PI, VER, VAL } : CL3 ]

onde o termo ISO9001 representa os requisitos da ISO 9001 como se fosse uma área de processo, e os termos REQM, RD, TS, PI, VER, VAL representam respectivamente as seis áreas de processo da categoria de engenharia do modelo CMMI-SE/SW.

O quinto projeto, realizado entre 2002 e 2003, também foi iniciado com o objetivo de atingir o nível 2 de maturidade do modelo SW-CMM. Durante o início da execução, porem foi percebido que os objetivos da organização e sua natureza exigiam ajustes nesta referência para o projeto. A unidade organizacional era um centro de computação responsável pelo desenvolvimento e evolução de sistemas para a área fim de uma entidade militar. Após uma análise e discussão com a organização foi então definido um perfil de processo com alguns elementos do nível 2 do SW-CMM e inclusão de processos

da ISO/IEC TR 15504-5 para adequar aos objetivos. Esta experiência pode ser descrita segundo a perspectiva da abordagem PRO2PI, conforme Tabela 26.

Tabela 26 – PCPs do quinto projeto

PCP1:	[ {RM, SPP, SPTO, SSM, SQA, SCM}: CL2 ]
PCP2:	[ {RM, SPP, SPTO, SSM, SQA, SCM, SwTest}: CL2 ]
PCP3.1:	[ {RM, SCM, SwTest, InfraStr}: CL2 ]
PCP3.2:	[ {RM, SCM, SwTest, InfraStr, SPP, SPTO, SSM, SQA} : CL2 ]

Na Tabela 26 os termos RM, SCM, SPP, SPTO, SSM e SQA identificam as seis áreas chave de processo do SW-CMM, vistas como áreas de processo, e os termos SwTest e InfraStr indicam processos da ISO/IEC TR 15504-5.

O sexto projeto, realizado em 2004, foi o estabelecimento de um perfil de capacidade de processo para a contratação e gerência de serviços externos de uma grande organização privada responsável por vários jornais [Lobo 2004, Lobo e Salviano 2005]. A estratégia para seleção dos processos foi a representação em alto nível do processo atual e a atribuição de processos da ISO/IEC TR 15504-5 a cada atividade. A Figura 86 ilustra esse processo e os processos atribuídos.

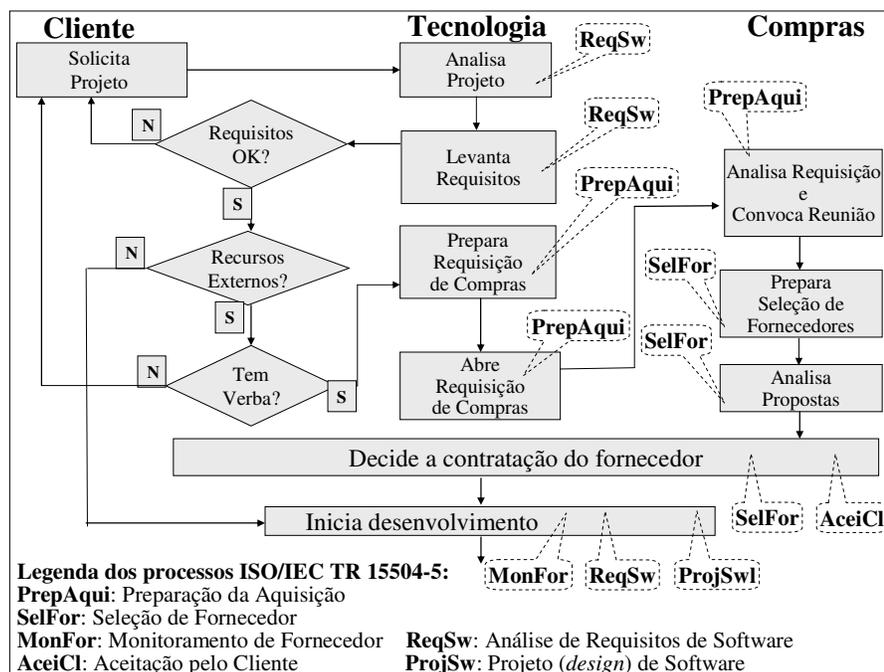


Figura 86 – Processo de contratação da organização

Os processos selecionados foram identificados por PrepAqui, SelFor, MonFor, AceiCl, ReqSw e ProjSw, que são respectivamente os processos de preparação para aquisição, seleção de fornecedor, monitoramento de fornecedor, aceite do cliente, análise de requisitos de software e projeto (design) de

---

software, todos da ISO/IEC TR 15504-5, e no nível 2 de capacidade de processo, conforme descrito a seguir:

[ { PrepAqui, SelFor, MonFor, AceiCl, ReqSw, ProjSw } : CL2 ].

O sétimo projeto, realizado em 2005, foi o estabelecimento de um perfil de capacidade de processo para a melhoria de uma empresa de pequeno porte especializada em fornecimento de soluções de software para gestão pública, mais especificamente atendendo prefeituras e câmaras municipais, localizada em Campo Grande-MS, com cerca de vinte colaboradores dos quais 60% se encontram na área técnica [Tomi 2005]. A modalidade de prestação de serviços é de locação de sistemas aplicativos específicos para utilização em entidades públicas, com implantação, treinamento e manutenção de ordem legal e tecnológica dos produtos. Para orientar a melhoria, foi definido o seguinte perfil de capacidade de processo:

[ { EstProc, SupClie, ManSw, Valid, GerProj } : CL2 ]

onde EstProc é o processo de estabelecimento de processo, selecionado para uma melhor organização dos processos e busca de melhoria contínua, SupClie é o processo de suporte ao cliente, selecionado para manter um nível aceitável de serviços ao cliente, ManSw é o processo de manutenção de sistemas e *software*, selecionado para garantir o atendimento dos requisitos dos produtos oferecidos, Valid é o processo de validação, selecionado para certificar que novas rotinas sejam disponibilizadas com garantia de atendimento dos requisitos, GerProj é o processo de gerência de projetos, selecionado para manter a integração e coerência das ações da organização.

O oitavo projeto, também realizado em 2005, foi o estabelecimento de um perfil de capacidade de processo para a melhoria de uma empresa de desenvolvimento WEB com atuação direta no extremo sul da Bahia, leste mineiro e norte do Espírito Santo [Almeida 2005]. A empresa atua no mercado há aproximadamente 3 anos e é composta por 8 funcionários, sendo 2 analistas, 1 consultor, 3 programadores e 2 técnicos de rede. A atividade principal é o fornecimento de soluções voltadas para o ambiente corporativo no desenvolvimento de sistemas integrados a Internet, rede de computadores, serviços de banco de dados e a segurança dos dados com aplicação de planos de contingência e continuidade da informação. A empresa tem como foco disponibilizar os serviços e produtos de software desenvolvidos para pequenas e médias empresas, privadas ou públicas, com respectiva instalação, implantação, customização, treinamento e suporte. Para orientar a melhoria, foi definido o seguinte perfil de capacidade de processo:

[ { PropFor, ElicReq, AlinhOrg, GerProj, GerConf, GerSMud } : CL2 ]

onde PropFor é o processo de proposta de fornecimento, ElicReq é o processo de elicitação de requisitos, AlinhOrg é o processo de alinhamento organizacional, GerProj é o processo de gerência de projetos, GerConf é o processo de gerência de configuração, e GerSMud é o processo de gerência de solicitação de mudanças. Os processos PropFor e ElicReq foram selecionados para obterem melhores resultados e propiciar um maior controle dos processos de fornecimento e engenharia objetivando a melhor interação entre o cliente e a empresa, evitando desvios não calculados do projeto original e o cumprimento de prazos e custos apresentados. Os processos AlinhOrg e GerProj foram selecionados para trabalhar as informações da empresa (metas e regras de negócio) e o controle dos projetos desenvolvidos com a finalidade de controlar adequadamente e atingir excelência na execução de suas atividades dentro de um mercado altamente competitivo e com fortes influências de globalização. Os processos GerConf e GerSMud foram selecionados porque se relacionam com o gerenciamento dos artefatos, documentos gerados pelos processos e com a necessidade do cliente em requisitar serviços e melhorias para um determinado produto adquirido, existe a necessidade de avaliação e melhoria desses processos vinculados a um maior controle de solicitações; análise crítica de impacto no projeto inicial; a devida especificação de recursos e definição de prazos e custos para a implementação do mesmo.

O nono projeto, também realizado em 2005, foi o estabelecimento de um perfil de capacidade de processo para a melhoria de uma empresa, com a utilização da versão corrente de PRO2PI-WORK. Esse projeto é mencionado em von Wangenheim et al. [2005b]. A empresa é especializada no planejamento e desenvolvimento de sistemas informatizados para mercados específicos. Esta empresa atualmente oferece soluções baseadas em quatro sistemas principais, para as áreas de transportes, judiciário, gestão de projetos e construção civil. Cada um desses sistemas é evoluído constantemente, orientado por mudanças de tecnologia, visões da empresa e solicitações dos clientes. Estas evoluções são realizadas por meio do processo de manutenção de curto prazo e processo de desenvolvimento de novos projetos. Cada sistema ou conjunto de subsistemas é parametrizado e adaptado para cada cliente.

Como o objetivo da empresa incluía a busca do nível 2 de maturidade do CMMI-SE/SW, foram apresentados para pessoas chave da empresa 12 áreas da processo. Durante a apresentação e discussões foi percebido que as áreas de processo do CMMI atenderiam apenas a área de projetos da empresa, que corresponde a cerca de 15 % das atividades. Os 85 % restantes, que eram relacionados a manutenções rápidas, não seriam tratados de forma satisfatória. Com isto foi apresentado o processo de gerência de solicitação de mudanças da ISO/IEC 15504-5 como orientação para estas manutenções rápidas.

O escopo proposto inicialmente foi aplicar a melhoria no processo de desenvolvimento de novos projetos em algumas áreas da empresa tendo como referência o nível 2 de maturidade do CMMI-SE/SW. Isto foi motivado pela proximidade do foco para o qual o CMMI foi desenvolvido (para projetos de software desenvolvidos por encomenda). Entretanto, a maior parte das atividades da empresa são relacionadas a manutenções de curto prazo. Os novos projetos são eventuais e são pouco representativos no negócio da empresa, além de poderem ser caracterizados como manutenções com prazos maiores. Por isto, foi decidido direcionar o projeto para a:

- melhoria gradual e contínua do processo de manutenção de curto prazo, utilizando práticas selecionadas do CMMI que sejam relevantes para esse processo no contexto da empresa, e
- definição de um processo de desenvolvimento de novos projetos baseado no nível 2 de maturidade do CMMI, a partir dos procedimentos já existentes e outros relacionados .

Esse foco da melhoria representa a primeira parte da implantação de uma estratégia composta por duas fases, conforme descrito na Figura 87.

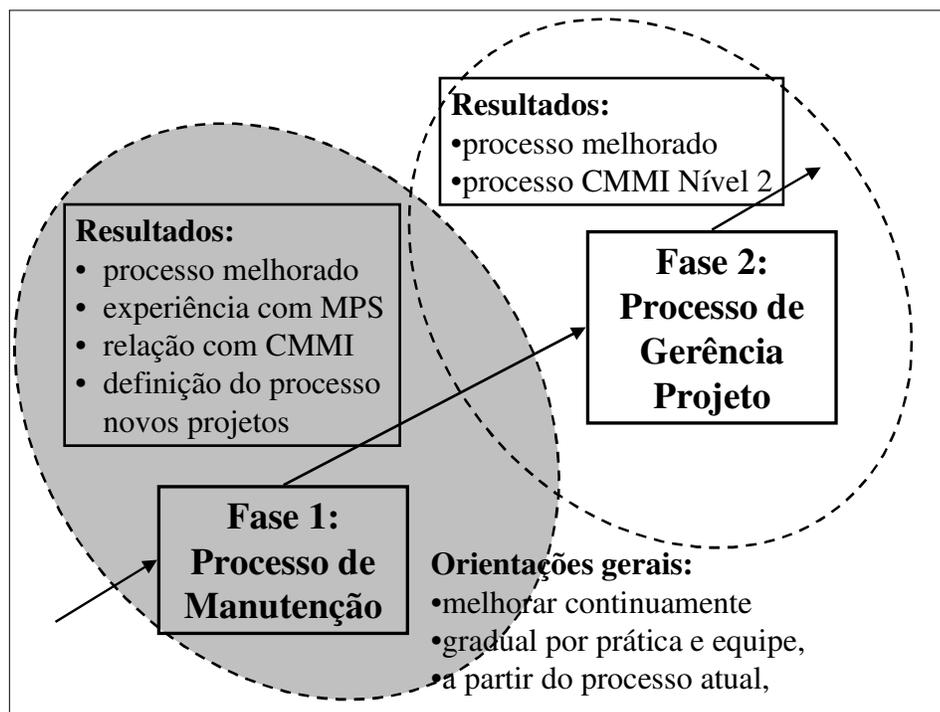


Figura 87 – Duas fases do projeto de melhoria

A fase 1 é orientada por um perfil de capacidade de processo baseado no processo de Gerência de solicitação de mudanças da ISO/IEC 15504-5 e elementos do CMMI. A fase 2 é orientada pelo nível 2 de maturidade do CMMI. Estas fases 1 e 2 podem ser descritas por dois perfis de capacidade de

processo, acrescido de mais um perfil para cobrir o correspondente ao nível 3 de maturidade, conforme Tabela 26.

Tabela 27 – PCPs do nono projeto

<b>Fase</b>	<b>Perfil de Capacidade de Processo para a fase</b>
1	[ {GerOrg, SupClie, EngDom} : CL1, GerSMud : CL2, { RD, REQM, PP, PMC, SAM, CM, PPQA, MA } : CL1, { TS, PI, VAL, VER, OPD, OPF, DAR, RSKM, OT } : CL1 ]
2:	[ {GerOrg, SupClie, GerSMud, EngDom} : CL2, { RD, REQM, PP, PMC, SAM, CM, PPQA, MA } : CL2, { TS, PI, VAL, VER, OPD, OPF, DAR, RSKM, OT } : CL2 ]
3	[ {GerOrg, SupClie, GerSMud, EngDom} : CL3, { RD, REQM, PP, PMC, SAM, CM, PPQA, MA } : CL3, { TS, PI, VAL, VER, OPD, OPF, DAR, IPM, RSKM, OT } : CL3 ]

Na Tabela 25 o termo GerOrg representa o processo de gerência organizacional, SupClie representa o processo de suporte ao cliente, EngDom representa o processo de engenharia de domínio, e GerSMud representa o processo de gerência de solicitação de mudanças, todos definidos na ISO/IEC 15504-5. Esses quatro processos formam uma base para empresas orientadas ao desenvolvimento, comercialização e evolução contínua de sistemas. Os outros termos da tabela representam as áreas de processo do CMMI-SE/SW.

A Tabela 28 descreve uma síntese dos perfis de capacidade de processo estabelecidos nos nove projetos de melhoria.

Tabela 28 – Síntese dos PCPs estabelecidos em nove projetos de melhoria

<b>Projeto</b>	<b>Perfil de Capacidade de Processo</b>
1	[ FornSw : CL2, GerProj : CL2, { GarQual, ElicReq, TestSw, Docum, GerConf } : CL1 ]
2	[ { RM, SSM : CL2, {FornSw,PrepAqui,AceitCl,GerProj,Docum,GerConf,GarQual,EstProc,Medic}: CL1} ]
3	[ { PrepAqui, SelFor, MonFor, AceiCl, DesSw, Docum, GerRH } : CL2 ]
4	[ { ISO9001, REQM, RD, TS, PI, VER, VAL } : CL3 ]
5	[ { RM, SCM, SwTest, InfraStr } : CL2 ]
6	[ { PrepAqui, SelFor, MonFor, AceiCl, ReqSw, ProjSw } : CL2 ]
7	[ { EstProc, SupClie, ManSw, Valid, GerProj } : CL2 ]
8	[ { PropFor, ElicReq, AlinhOrg, GerProj, GerConf, GerSMud } : CL2 ]
9	[ {GerOrg, SupClie, EngDom} : CL1, GerSMud : CL2, { RD, REQM, PP, PMC, SAM, CM, PPQA, MA } : CL1, { TS, PI, VAL, VER, OPD, OPF, DAR, RSKM, OT } : CL1 ]

Nos nove perfis de capacidade de processo sintetizados na Tabela 28 foram utilizadas áreas de processo do modelo SW-CMM (projetos 2 e 5), CMMI-SE/SW (projetos 4 e 9), ISO/IEC 15504-5

(projetos 1, 2, 3, 5, 6, 7, 8 e 9) e ISO/9001 (projeto 4). Não houve coincidência do mesmo perfil em nenhum par de projetos. Isto evidencia a relevância da busca de um perfil de capacidade de processo para cada projeto. Dos nove projetos, em cinco deles houve a participação direta do autor de PRO2PI (projetos 1, 2, 3, 5, e 9). Nos outros quatro projetos (projetos 4, 6, 7 e 8), não houve participação direta do autor no estabelecimento do perfil.

#### **7.2.4 Projeto 15504MPE [2003-2004]**

O projeto de pesquisa 15504MPE (ISO/IEC 15504 para Micro e Pequenas Empresas) foi financiado pelo CNPq (CT-INFO: CNPq 11/2002 - PDPG-TI) e executado pelo LQPS/UNIVALI em parceria com o Centro de Pesquisas Renato Archer – CenPRA, a incubadora Centro GeNESS e empresas da Grande Florianópolis. O objetivo desse projeto foi consolidar um grupo em melhoria de processo de software na UNIVALI, incluindo o desenvolvimento e experimentação de uma metodologia de avaliação de processos de software baseado na Norma ISO/IEC 15504 (SPICE), para o contexto de micro e pequenas empresas (MPEs) de software, como um instrumento para o início de um programa de melhoria de processo alinhado aos objetivos, contexto e estratégia de negócios da organização. Esse grupo é o Laboratório de Qualidade e Produtividade de Software (LQPS) da UNIVALI.

Parte desse projeto, no caso a consolidação da metodologia realizada em 2004, foi submetido ao Programa Brasileiro de Qualidade e Produtividade em Software (PBQP-Software) no ciclo 2004, na categoria 2 (Método de Gestão) com o identificador 2.32, e premiado em segundo lugar [von Wangenheim e Salviano 2005].

A metodologia desenvolvida foi baseada inicialmente na metodologia do CenPRA para avaliação e melhoria de processo de software.

A principal contribuição científica desse projeto foi o desenvolvimento de um método inovador para avaliação dos processos de software em MPEs seguindo a Norma ISO/IEC 15504, com objetivo de melhorar os processos. Como fruto do projeto foram geradas experiências inéditas sobre a aplicação da Norma em MPEs Brasileiras. Isto iniciou a consolidação de uma base para melhoria da qualidade e produtividade dos processos de MPEs brasileiras auxiliando no aumento da sua competitividade.

Foi desenvolvido MARES/15504, um método de avaliação de processos de software conforme a 15504 customizado para MPEs enfocando na melhoria de processo. O método descreve o modelo de avaliação de processo e o processo de avaliação.

O modelo de avaliação de processo é baseado no modelo ISO/IEC FCD 15504-5 considerando um subconjunto de 26 processos tipicamente relevantes em micro e pequenas empresas de software. São considerados os níveis de capacidade 0-3 com base na norma ISO/IEC 15504.

O processo de avaliação engloba gerenciamento, contextualização e execução da avaliação conforme os requisitos da ISO/IEC 15504, incluindo guias para a sua aplicação na prática e documentos padrão. Uma contribuição principal do método é a fase de contextualização provendo suporte para a definição de perfis alvo e a seleção dos processos a serem avaliados em detalhes de acordo com as características de uma empresa e as suas metas de negócio e melhoria. A Figura 88 ilustra o método MARES e um detalhamento da contextualização conforme descrito em Anacleto [2005].

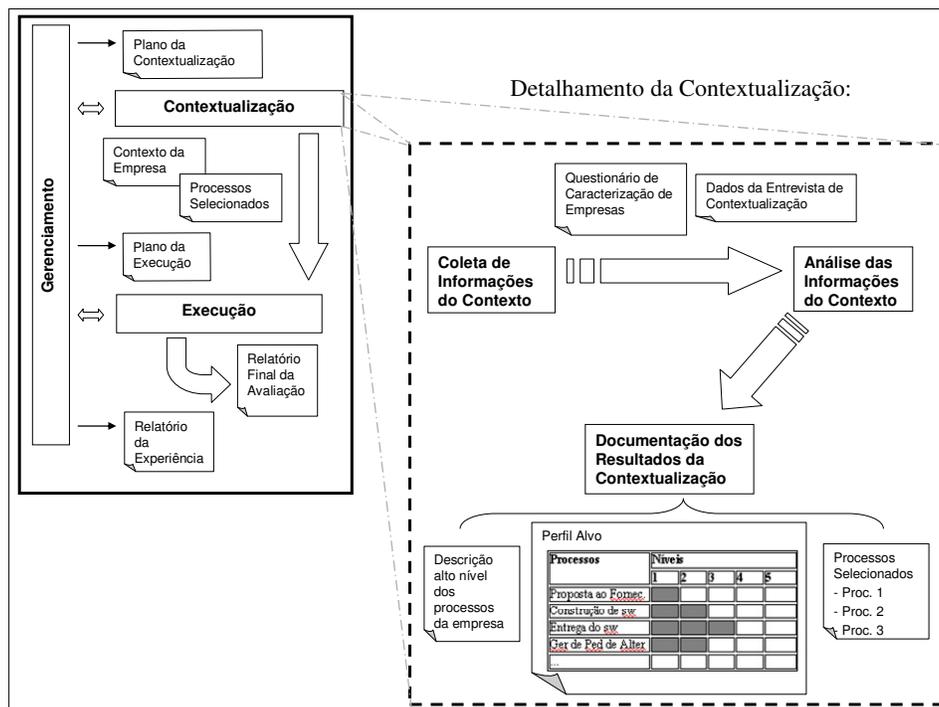


Figura 88 – Método MARES, com detalhe da Contextualização [Anacleto 2005]

Um dos principais resultados do projeto é o método de avaliação de processo de software adaptado a MPEs. O método está detalhadamente documentado e sendo disponibilizado publicamente na Internet. As principais inovações são:

- suporte à definição de perfis de capacidade de processo alvo e à escolha dos processos a serem avaliados alinhado à estratégia de negócio de uma empresa, e
- a adaptação da Norma ISO/IEC 15504 especificamente para o contexto de micro e pequenas empresas.

As experiências de aplicação do método na prática, indicam a sua adequação para esse tipo de organização de forma efetiva e eficiente. Também foi desenvolvido um protótipo de uma ferramenta de software que suporta a aplicação do método. Dentro do contexto desse projeto também foram realizados 7 avaliações de processo de software em MPEs, iniciando a validação do método provendo dados quantitativos e qualitativos relacionados ao seus benefícios e custos na prática.

Esse projeto se mostrou significativo para facilitar a melhoria e avaliação de processos de software no contexto de MPEs Brasileiras, conforme uma Norma internacionalmente reconhecida. Isto contribuirá para uma maior competitividade das MPEs no mercado global podendo também aumentar as exportações de software. Em geral, os resultados do projeto também podem ser aplicados internacionalmente ou em outros tipos de empresa de software com pequenas adaptações.

Durante a realização do projeto foram publicados vários artigos técnicos em conferências nacionais e internacionais [Anacleto et al. 2003, Anacleto et al. 2004c, Anacleto et al. 2005a, Pickler et al. 2005, Anacleto et al. 2004a, Anacleto et al. 2004b]. Além destas publicações, foram produzidos seis relatórios técnico com o resultado da avaliação de cada uma das sete empresas de software da Grande Florianópolis/SC, que participaram do projeto.

Foram realizadas avaliações de processo com as várias versões do método para início de programas de melhoria de processo em sete empresas de software da grande Florianópolis [Anacleto et al. 2005]. Durante a realização do projeto foi desenvolvida uma tese de mestrado [Anacleto 2004].

Os resultados desse projeto facilitam uma ampla aplicação da Norma internacionalmente reconhecida para tipo de empresa no Brasil, contribuindo na melhoria do processo de software e auxiliando assim no aumento da sua competitividade no mercado nacional e internacional.

Esse projeto, junto com outros projetos do CenPRA, LQPS/UNIVALI e outras instituições, representam também um avanço na utilização dos modelos contínuos de capacidade de processo (neste caso o modelo ISO/IEC 15504-5, mas também poderia ser a representação contínua do modelo CMMI-SE/SW) como uma alternativa viável e eficaz para a melhoria das organizações intensivas em software, com foco nos objetivos, contexto e estratégia de negócios específicos de cada organização. Existe, na nossa visão, uma tendência do aumento da utilização dos modelos contínuos em relação aos modelos por estágio, com o desenvolvimento e disseminação de metodologias para definição de perfis de capacidade de processo, como a fase de contextualização da metodologia MARES e a abordagem PRO2PI. Esta experiência foi generalizada [von Wangenheim et al. 2006b] e comparada com experiências semelhantes na Finlândia [von Wangenheim et al. 2005a, von Wangenheim et al. 2006a].

### **7.2.5 Melhoria de processo em grupos de empresas [2004-2005]**

O método PRO2PI-WORK foi utilizado em dois programas cooperativos com grupos de empresas para melhoria de seus processos, baseados nos modelos CMMI-SE/SW e MR-MPS. O termo cooperativo é utilizado para indicar a realização conjunta de projetos. Nesses programas, juntam-se os esforços de várias empresas, compartilhando custos de treinamentos e permitindo a troca de experiências. São fornecidos treinamentos e cada empresa, apoiada por consultores qualificados, desenvolve o seu projeto de melhoria de processo de software consoante com a sua realidade e cultura empresarial e inserida no seu planejamento estratégico, com resultados muito positivos.

O primeiro grupo desta experiência foi organizado como uma cooperação entre o CenPRA e o ITS – São Paulo e foi composto por cinco empresas. O segundo grupo desta experiência foi organizado como uma cooperação entre o CenPRA, a ASR consultoria e o Núcleo Softex Campinas e foi composto por três empresas.

Em ambos, foi apresentado no início do programa, uma oficina baseada no método PRO2PI-WORK para apresentação de áreas de processo do modelo CMMI-SE/SW, identificação da correspondência, importância e risco de cada área para cada empresa, e produção do quadro de importância e risco. No primeiro grupo desta experiência, foram apresentadas 12 áreas de processo, enquanto que no segundo foram apresentadas 18 áreas de processo. No segundo grupo foram utilizados os formulários do PRO2PI-WORK. A Figura 89 apresenta o quadro das três empresas.

A partir desta análise da relevância de cada área de processo para cada empresa, foi definido um plano de melhoria único para as três empresas, conciliando as prioridades de cada uma.

### **7.2.6 PRO2PI para domínios específicos [2004-2005]**

As idéias e propostas de PRO2PI influenciaram, direta ou indiretamente, alguns trabalhos de desenvolvimento de abordagens e modelos mais específicos para determinados contextos ou segmentos.

No primeiro trabalho Oliveira [2004] adaptou a abordagem AMP1 para aplicação em uma organização estatal, com a seguinte estratégia: inclusão de uma nova fase como a fase inicial, para tratar da busca do comprometimento, ao invés de ter isto como um pré-requisito e customização das outras fases para o cenário específico.

No segundo trabalho o MPS-BR identificou a oportunidade de construção de um modelo de capacidade de processo que refletisse melhor características particulares do cenário nacional,

principalmente das micro e pequenas empresas. Foi então definido um modelo estagiado, baseado nos níveis de maturidade do CMMI, mas com a inclusão de níveis intermediários para reconhecer a melhoria mais gradual das organizações [Weber et al. 2005a, Weber et al. 2005b].

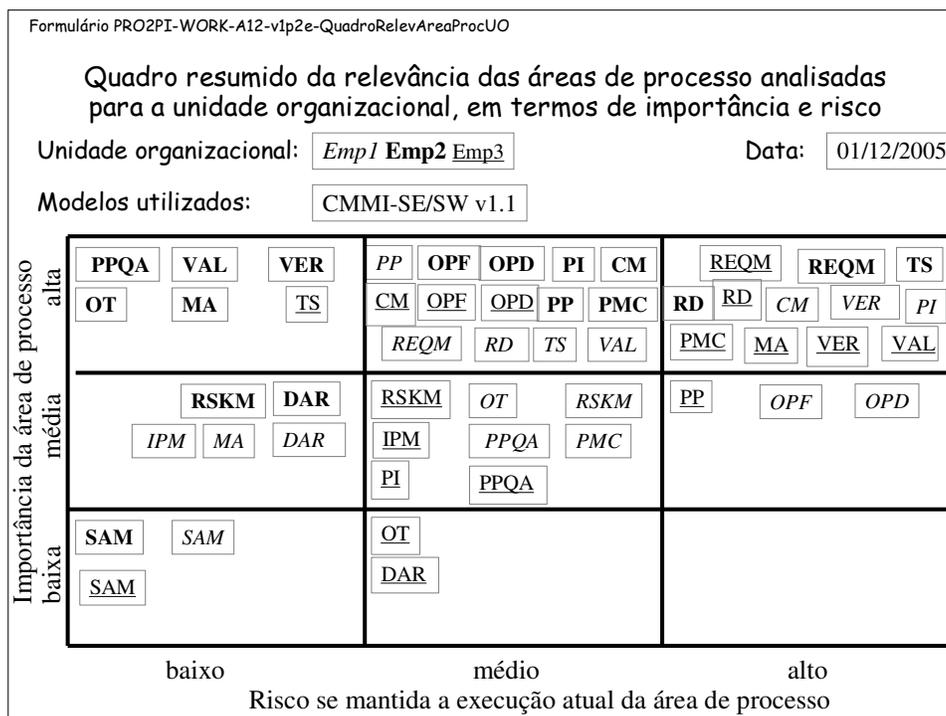


Figura 89 – Quadro de relevância dos processos em três empresas

Como terceiro e quarto trabalhos, estão sendo desenvolvidos dois modelos de capacidade de processo para o desenvolvimento de software baseado em componentes. Para tanto, em um deles foi selecionado as três áreas de processo do grupo de reuso da ISO/IEC 15504-5, convertido para a notação do CMMI e produzido orientações para as áreas de processo da categoria engenharia dos modelos do CMMI. Uma proposta de descrição dos três processos do grupo de reuso da ISO/IEC 15504-5 no padrão do CMMI foi realizado por Ferreira [2005]. Para o outro modelo, estão sendo utilizadas os processos do grupo de processos de engenharia, com orientações adicionais para o desenvolvimento de software baseado em componentes, e os três processos do grupo de reuso da ISO/IEC 15504-5.

No quinto trabalho Cavalcante e Costa [2005] desenvolveram um método para especialização de modelos de capacidade de processo para domínios específicos. Esse método é baseado na linha de pesquisa de PRO2PI que defende o desenvolvimento e utilização de modelos de capacidade de processo para domínios e segmentos de negócio e propõe elementos para apoiar esse desenvolvimento. O método desenvolvido é composto por oito fases:

- (seleção e) caracterização do domínio,
- (escolha de um ou mais modelos de capacidade de processo) e seleção das áreas de processo para o domínio,
- descrição inicial do domínio,
- exploração da descrição do domínio e especialização das áreas de processo,
- consolidação da descrição do domínio e especialização das áreas de processo,
- validação, e
- revisão e conclusão.

Para especialização das áreas de processo selecionadas é utilizado um formulário baseado no formulário PRO2PI-WORK-A20. Nesse caso, para cada prática da área de processo é identificado o grau de importância desta área para o domínio em relação ao domínio mais genérico original do modelo da área de processo. Esse método foi aplicado para o domínio bancário com a seleção de quatro áreas de processo do modelo CMMI-SE/SW: desenvolvimento de requisitos, gerência de requisitos, planejamento de projeto, e controle e monitoramento de projeto.

No sexto trabalho foi desenvolvida uma área de processo para ensino técnico como uma nova área de processo no formato da ISO/IEC 15504-5 [Miranda 2005].

No sétimo trabalho, a partir de experiências com programas cooperativos de melhoria de processo de software, o CenPRA consolidou um Método para Projetos Cooperativos de Melhoria de Processo de Software (Coop-MPS) alinhado aos objetivos, contexto e estratégia de negócio das organizações e orientado por modelos de capacidade de processo, como os modelos MPS.BR, CMMI, ISO/IEC 15504 e iCMM. Esse utiliza elementos de PRO2PI-WORK para selecionar e identificar a relevância de cada área de processo a ser utilizada como referência para a melhoria [Salviano e Tsukumo 2005]. A grande distinção do Coop-MPS em relação a outros métodos é que na fase inicial do projeto em que as metas do melhoria são definidas, a referência não se limita aos níveis de maturidade, mas a outras áreas de processo, que, de forma coerente com a arquitetura contínua de modelos, e alinhadas aos objetivos estratégicos e aos problemas prementes, tratam melhor os objetivos das organizações. PRO2PI-WORK é utilizado em Coop-MPS exatamente para esse início.

### **7.2.7 Cursos com PRO2PI-WORK [2004-2005]**

O método PRO2PI-WORK também tem sido utilizado em cursos de pós-graduação (especialização) relacionados a engenharia e qualidade de software, em disciplinas de melhoria de processo de software. Para ajudar o entendimento da melhoria de processo com 15504 ou CMMI, em um curso de

pós-graduação é solicitado o desenvolvimento de um trabalho abordando o início de um programa de melhoria. Em duas turmas a distância e uma presencial, 135 (cento e trinta e cinco) trabalhos foram desenvolvidos seguindo três versões de PRO2PI-WORK. A Tabela 29 relaciona 10 experiências.

Tabela 29 – Experiência com PRO2PI-WORK e trabalhos gerados

<b>Id</b>	<b>Curso</b>	<b>Versão</b>	<b>Horas Aula.</b>	<b>Meses/Ano</b>	<b>Alunos</b>	<b>Trabalhos</b>
E01	MPS 15504CMMI	V0.1	36 (d) e 4 (p)	04-05/2004	18	18
E02	QDS NPS 15504	V0.1	40 (p)	08-09/2004	22	10
E03	MPS 15504CMMI	V0.1	36 (d) e 4 (p)	10-11/2004	37	31
E04	QDS NPS 15504	V0.2	40 h.a. (p)	02-05/2005	11	4
E05	MPS 15504CMMI	V0.2	36 (d) e 4 (p)	04-05/2005	27	20
E06	Modelos SJT T1	V0.2	12 (p)	05-06/2005	31	13
E07	Modelos SJT T2	V0.2	12 (p)	02-06/2005	24	9
E08	MPS 15504CMMI	V0.3	36 (d) e 4 (p)	10-11/2005	42	32
E09	CMMI-MPSBR	V0.3	36 (d) e 4 (p)	10-11/2005	22	17
E10	ES CMMI15504	V0.3	24 (p)	10-11/2005	27	10

A versão 0.1 foi utilizada nas experiências E01, E02 e E03. O processo de PRO2PI-WORK, ou seja, a realização de cada curso e o desenvolvimento do trabalho por cada aluno ou grupos de alunos, foi realizada como um processo no nível 1 de capacidade. Foi solicitado aos alunos o desenvolvimento do trabalho, sem um estrutura definida para o trabalho, nem templates para o desenvolvimento.

A experiência E01 (MPS 15504CMMI) na disciplina Melhoria e Avaliação de Processo com ISO/IEC 15504 (SPICE) e CMMI, com 36 horas aula à distância e 4 horas aula presenciais, do curso de pós-graduação (especialização) de Melhoria de Processo de Software da Universidade Federal de Lavras (UFLA), realizada durante os meses de abril e maio de 2004. Nesta disciplina foram apresentados os modelos ISO/IEC 15504-5 e CMMI-SE/SW para dezoito alunos. Cada aluno desenvolveu um trabalho.

A experiência E02 (QDS NPS 15504) foi a disciplina Normas de Processo de Software, com 40 horas aula, do curso de pós-graduação em Qualidade no Desenvolvimento de Software da Faculdade SENAC de Ciências Exatas e Tecnologia, realizada durante os meses de agosto a setembro de 2004 para 22 alunos. Foram desenvolvidos 10 trabalhos, individuais ou em grupos de dois a três alunos.

A experiência E03 (MPS 15504CMMI) na disciplina Melhoria e Avaliação de Processo com ISO/IEC 15504 (SPICE) e CMMI, com 36 horas aula à distância e 4 horas aula presenciais, do curso

de pós-graduação (especialização) de Melhoria de Processo de Software da Universidade Federal de Lavras (UFLA), realizada durante os meses de outubro e novembro de 2004, para 37 alunos. Foram desenvolvidos 31 trabalhos, individuais ou em grupos de dois a três alunos.

A versão 0.2 foi utilizada nas experiências E04, E05, E06 e E07. O processo de PRO2PI-WORK, ou seja, a realização de cada curso e o desenvolvimento do trabalho por cada aluno ou grupos de alunos, foi realizada como um processo no nível 2 de capacidade. Foi solicitado aos alunos o desenvolvimento do trabalho, a partir de uma estrutura definida para o trabalho, mas sem templates para o desenvolvimento.

A experiência E04 (QDS NPS 15504) foi a disciplina Normas de Processo de Software, com 40 horas aula, do curso de pós-graduação em Qualidade no Desenvolvimento de Software da Faculdade SENAC de Ciências Exatas e Tecnologia, realizada durante os meses de fevereiro a maio de 2004 para 11 alunos. Foram desenvolvidos 4 trabalhos, em grupos de dois a três alunos.

A experiência E05 (QDS NPS 15504) foi a disciplina Normas de Processo de Software, com 40 horas aula, do curso de pós-graduação em Qualidade no Desenvolvimento de Software da Faculdade SENAC de Ciências Exatas e Tecnologia, realizada durante os meses de fevereiro a abril de 2004, para 27 alunos. Foram desenvolvidos 20 trabalhos, individuais ou em grupos de dois a três alunos

A experiência E06 (Modelos SJT T1) foi composta por 12 horas aula da disciplina Modelos de Qualidade de Software, com um total de 40 horas aula, do curso de pós-graduação (especialização) em Engenharia de Software da Universidade São Judas Tadeu, realizada durante os meses de maio e junho de 2005, para 31 alunos. Foram desenvolvidos 13 trabalhos em grupos de dois, três ou quatro alunos.

A experiência E07 (Modelos SJT T2) foi composta por 12 horas aula da disciplina Modelos de Qualidade de Software, com um total de 40 horas aula, do curso de pós-graduação (especialização) em Engenharia de Software da Universidade São Judas Tadeu, realizada durante os meses de maio e junho de 2005, para 24 alunos. Foram desenvolvidos 9 trabalhos em grupos de dois, três ou quatro alunos.

A versão 0.3 foi utilizada nas experiências E08, E09 e E10. O processo de PRO2PI-WORK, ou seja, a realização de cada curso e o desenvolvimento do trabalho por cada aluno ou grupos de alunos, foi realizada como um processo no nível 3 de capacidade. Foi solicitado aos alunos o desenvolvimento do trabalho, a partir de uma estrutura definida para o trabalho e templates para o seu desenvolvimento.

A experiência E08 foi na disciplina Melhoria e Avaliação de Processo com ISO/IEC 15504 (SPICE) e CMMI, com 36 horas aula à distância e 4 horas aula presenciais, do curso de pós-graduação (especialização) de Melhoria de Processo de Software da Universidade Federal de Lavras (UFLA),

realizada durante os meses de outubro e novembro de 2005, para 42 alunos. Foram desenvolvidos 32 trabalhos, sendo 22 individuais e 10 em grupos de dois alunos cada.

A experiência E09 foi na disciplina Introdução à Melhoria de Processo de Software com Modelos do CMMI e MPS.BR, com 36 horas aula à distância e 4 horas aula presenciais, do curso de pós-graduação (especialização) de Modelo de Capacidade da Maturidade Integrado CMMI e MPS-BR, da Universidade Federal de Lavras (UFLA), realizada durante os meses de julho, agosto e novembro de 2005, para 22 alunos. Foram desenvolvidos 17 trabalhos, sendo 13 individuais, 2 em grupos de dois alunos cada e 1 em grupo de quatro alunos.

A experiência E10 (ES CMMI15504MPSBR) foi a disciplina Qualidade de Processo, com 24 horas aula do curso de pós-graduação (especialização) em engenharia de software de Universidade Metodista de Piracicaba, realizada durante os meses de outubro e novembro de 2005, para 27 alunos. Foram desenvolvidos 10 trabalhos em grupos de dois, três ou quatro alunos.

A Tabela 30 resume a quantidade de alunos que realizaram o trabalho e a quantidade de trabalhos realizados com cada uma das três versões de PRO2PI-WORK.

Tabela 30 – Resumo dos trabalhos de cursos PRO2PI-WORK

<b>Versão</b>	<b>Alunos</b>	<b>Trabalhos</b>	<b>Comentários</b>
V0.1	77	59	Método no nível 1 de capacidade
V0.2	93	46	Método no nível 2 de capacidade
V0.3	91	59	Método no nível 3 de capacidade
Total	261	164	

### **7.3 Síntese das análises e aplicações**

Esse capítulo descreveu análises e aplicações dos fundamentos e elementos de PCDE e PRO2PI com o objetivo de validar as propostas.

O posicionamento dos dezesseis modelos mais relevantes nas três gerações de arquiteturas proposta, a análise desses mesmos dezesseis modelos segundo os três elementos conceituais mais importantes para PRO2PI (área de processo, nível de capacidade de processo e perfil de capacidade de processos) e o posicionamento dos quatro modelos de capacidade de processo em relação aos elementos estruturais do modelo PRO2PI-MODEL mostram que as propostas dessa tese podem representar as boas práticas que estão representadas nos modelos de capacidade de processo mais relevantes. Isso viabiliza o estabelecimento de um PRO2PI a partir de referências desses modelos. Os

modelos analisados foram SEI SW-CMM v1.1, SEI CMMI-SE/SW estagiado v1.1, SEI CMMI-SE/SW contínuo v1.1, ISO/IEC 15504-5, FAA iCMM v2.0, ISO/IEC 12207 Amd2, ISO 9001:2000, PMI PMBOK Third Edition, PMI OPM3 v1.0, ITsqc eSCM-SP v2, MPS-BR MR-MPS v1.0, IEEE SWEBOK 2004, ITGI COBIT v4.0, EFQM, FPNQ PNQ e RUP.

A Tabela 31 sintetiza os tipos de utilizações, a quantidade de projetos, e a quantidade de utilizações nesses projetos, dos fundamentos e elementos de PRO2PI, descritas nas seções anteriores.

Tabela 31 – Síntese das utilizações de PRO2PI

<b>Característica e período da utilização</b>	<b>Projetos</b>	<b>Usos</b>
Melhoria de processo de software na Empresa E1 [1999-2002]	1	3
Melhoria de processo de software na Empresa E2 [2002-2003]	1	1
Estabelecimento de perfil de capacidade de processo [2000-2005]	9	12
Projeto 15504MPE [2003-2004]	1	5
Melhoria de processo em grupos de empresas [2004-2005]	2	8
PRO2PI para domínios específicos [2004-2005]	7	7
Cursos com PRO2PI-WORK [2004-2005]	10	164
<b>TOTAL</b>	<b>32</b>	<b>200</b>

Ao todo, os fundamentos e elementos de PRO2PI foram utilizados em 32 (trinta e dois projetos) e nesses projetos houve um total de 200 (duzentas) utilizações de PRO2PI, principalmente relacionados ao estabelecimento de um perfil de capacidade de processo para orientar um ciclo de melhoria em organizações intensivas em software. Com isso existe uma confiança adequada que PRO2PI e PRO2PI-WORK são viáveis para a melhoria de processo e para o ensino de melhoria de processo de software baseado no estabelecimento de um PRO2PI, com três das oito propriedades: relevante, viável e rastreável.

As análises e aplicações dos fundamentos e elementos de PCDE e PRO2PI em conjunto com o desenvolvimento com a abordagem *indústria-como-laboratório*, provêm uma confiança adequada da qualidade de PCDE e PRO2PI como propostas para a evolução da melhoria de processo de software.

# Capítulo 8

## 8 Conclusão

Os objetivos deste capítulo incluem (i) a apresentação de uma síntese do trabalho desenvolvido, (ii) a descrição das contribuições principais desta tese em termos dos objetivos, propostas, validações, e os relacionamentos entre eles, (iii) a descrição do relacionamento desta tese com outros projetos e outras propostas, (iv) limitações das contribuições, (v) uma agenda de pesquisa para trabalhos futuros e (vi) considerações finais.

### 8.1 Síntese do trabalho

A área de melhoria de processo de software, baseada em níveis de maturidade da capacidade de processo, se estabeleceu a partir anos 1980 como uma abordagem eficiente para a construção do processo de projetos de desenvolvimento de software por encomenda. Com isto a melhoria de processo acrescentou uma nova dimensão na engenharia de software, a dimensão de estabelecimento de processos, e conseguiu influenciar mais o estado da prática de produção de software.

Esta pesquisa começou com o desenvolvimento de um método para escolha de processos de um modelo contínuo, no caso ISO/IEC TR 15504-5, como um impulsionador para uma melhoria de processo de software mais alinhada com o contexto, objetivos e estratégia de negócio de uma organização intensiva em software, do que a melhoria com os níveis de maturidade fixos dos modelos estagiados, principalmente o modelo SW-CMM e os modelos do CMMI. Com o detalhamento e utilização desse método, os objetivos foram ampliados para uma extensão da melhoria de processo, utilizando a abordagem de pesquisa indústria-como-laboratório.

São identificadas três gerações de arquitetura de modelos de capacidade de processo, organizadas pelos conceitos de níveis de capacidade de processo, áreas de processo e perfis de capacidade de processo. Estas três gerações, denominadas de arquitetura estagiada fixa, arquitetura contínua fechada e arquitetura contínua aberta, substituem a caracterização atual de arquitetura estagiada e contínua.

É proposta uma {(Engenharia de processo) (de Software e de qualquer outro Trabalho Humano Intensivo em Conhecimento) Dirigida por (Perfis de Capacidade de Processo)} como evolução da atual

melhoria de processo de software baseada em modelos de capacidade de processo. É proposta então uma abordagem exemplo para esta engenharia. Esta abordagem é descrita por um conjunto de oito propriedades que caracterizam um perfil para a melhoria, um metamodelo que unifica os elementos dos modelos de capacidade de processo mais relevantes, um ciclo de melhoria que inclui a definição e utilização de perfil e um conjunto de medições que inclui uma medição para a viabilidade de um perfil. É também proposto um método para o estabelecimento de perfis de capacidade de processo como uma oficina que é utilizado para ensino, consultoria, ou combinações de ensino e consultoria.

As propostas são validadas pelas análises dos modelos mais relevantes e pelas análises da aplicação em trinta e dois projetos, com um total de duzentas utilizações principalmente relacionadas ao estabelecimento de um perfil de capacidade de processo para orientar um ciclo de melhoria em organizações intensivas em software. A tese é concluída com uma proposta de agenda para continuação das pesquisas.

A engenharia de processo proposta entende a natureza das atividades da prática de software como algo a ser considerado e utilizado para a construção gradativa dos processos da engenharia de software. Esta construção é orientada por perfis de capacidade de processo para a melhoria dos fatores de negócio de uma organização.

## **8.2 Principais contribuições do trabalho**

As contribuições principais desta tese estão sintetizadas na Figura 90 em termos dos objetivos, propostas e validações, e os relacionamentos entre eles.

A primeira contribuição (Proposta 1) é a identificação de três gerações de arquitetura de modelos de capacidade de processo, organizadas pela estabilidade e variabilidade dos conceitos de níveis de capacidade de processo, áreas de processo e perfis de capacidade de processo. Estas três gerações, denominadas de arquitetura estagiada fixa, arquitetura contínua fechada e arquitetura contínua aberta, substituem a caracterização atual de arquitetura estagiada e contínua. Esta contribuição atende ao Objetivo 1 e é validada pela Validação 1, Validação 2 e pelo desenvolvimento da Proposta 2.

A segunda contribuição (Proposta 2) é a proposta da {(Engenharia de processo) (de Software e de qualquer outro Trabalho Humano Intensivo em Conhecimento) Dirigida por (Perfis de Capacidade de Processo)} (engenharia de processo PCDE), como evolução da atual melhoria de processo de software baseada em modelos de capacidade de processo. Esta proposta atende ao Objetivo 2 e o seu

desenvolvimento é uma validação da Proposta 1. Esta proposta é validada pela Validação 2 e pelo desenvolvimento da Proposta 3.

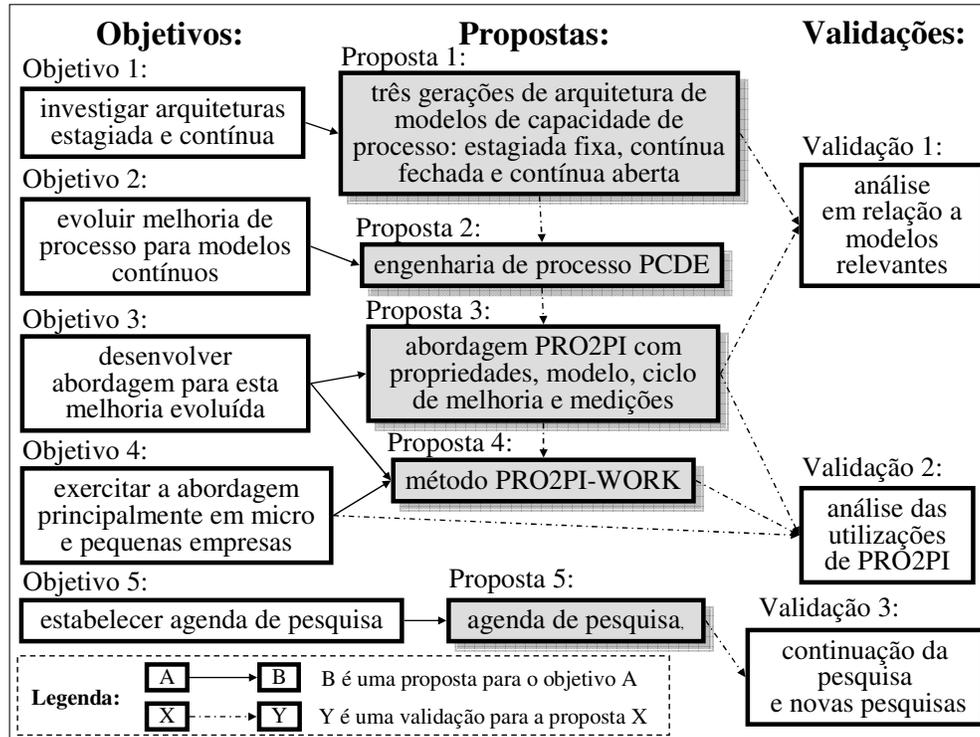


Figura 90 – Síntese dos objetivos, propostas e validações da pesquisa

A terceira contribuição (Proposta 3) é o desenvolvimento e utilização da abordagem exemplo desta engenharia de processo (abordagem PRO2PI) para melhoria de processo dirigida por perfil de capacidade de processo relevante, viável, oportunista, sistêmico, representativo, rastreável, específico e dinâmico. Esta abordagem é definida por um conjunto de propriedades, um modelo, um ciclo de melhoria e um conjunto de medições. Esta proposta atende ao Objetivo 3 e o seu desenvolvimento é uma validação da Proposta 2. Esta proposta é validada pela Validação 2.

A quarta contribuição (Proposta 4) é o desenvolvimento e utilização do método PRO2PI-WORK para estabelecimento de um perfil de capacidade de processo para a melhoria para capacitação em engenharia de processo e início de um ciclo de melhoria principalmente em micro e pequenas empresas. Esta proposta atende ao Objetivo 3 e o seu desenvolvimento é uma validação da Proposta 3. Esta proposta é validada pela Validação 3.

A quinta contribuição (Proposta 5) é o estabelecimento de uma agenda de pesquisa e desenvolvimento em engenharia de processo dirigida por perfis de capacidade de processo. Esta

proposta atende ao Objetivo 5. Esta quinta contribuição orienta a continuação dos trabalhos é validada pela Validação 4.

### 8.3 Relacionamentos com outras propostas

A Figura 91 relaciona no tempo as três fases do desenvolvimento de PRO2PI.

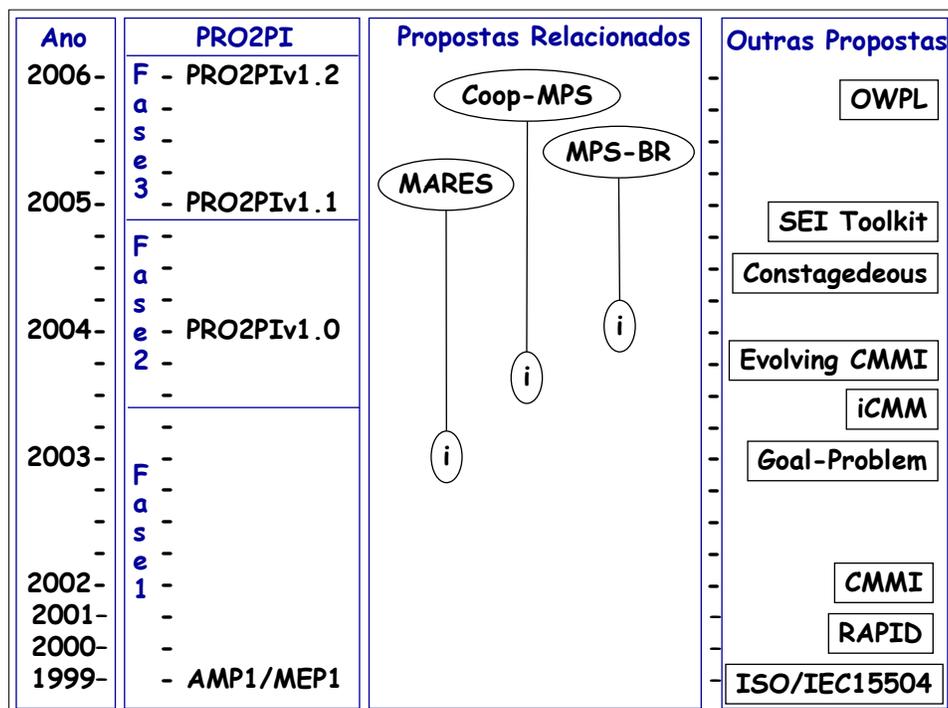


Figura 91 – Versões de PRO2PI e relacionamentos com outras propostas

A Figura 91 contém a identificação da versão inicial, chamada de AMP1 e MEP1, e as três versões com o nome PRO2PI. São também identificadas no tempo três outras propostas relacionadas a PRO2PI e nove outras propostas com os quais PRO2PI pode ser comparado e tem algum tipo de relacionamento.

A seguir estão descritos os relacionamentos dos resultados desta pesquisa, principalmente PRO2PI, com três outras propostas. Estas propostas estão relacionadas na Figura 91, com a identificação da época de início do projeto que produziu a proposta e a época de publicação do resultado principal. Todos esses projetos tiveram a participação do autor de PRO2PI, principalmente no início, e as idéias e elementos de PRO2PI influenciaram o desenvolvimento do projeto.

A metodologia MARES [von Wangenheim e Salviano 2005] foi desenvolvida no projeto 15504MPE e teve como referência inicial a abordagem AMP1, o método MEP1 e as idéias que viriam

a ser PRO2PI. Durante o projeto houve um desenvolvimento paralelo de MARES e PRO2PI. MARES pode ser considerado como um exemplo de como iniciar uma melhoria segundo a engenharia PCDE e a abordagem PRO2PI.

O projeto MPS-BR [Weber et al. 2005a] começou em 2003 com uma proposta técnica do CenPRA para a Softex. Esta proposta inicial era totalmente baseada na proposta de PRO2PI, com base na ISO/IEC 15504. No início do projeto MPS.BR foi decidido focar mais na melhoria das empresas e com isto não ser um projeto de pesquisa. Com a participação do autor de PRO2PI no projeto, foi sugerido a criação de níveis de maturidade intermediários em relação aos níveis de maturidade do CMMI e a utilização da ISO/IEC 15504-5 em conjunto com o CMMI. O projeto MPS-BR é independente de PRO2PI, mas o modelo MR-MPS pode ser visto como um exemplo da utilização de PRO2PI.

O projeto Coop-MPS [Salviano e Tsukumo 2005] teve como objetivo adaptar PRO2PI para uso em melhoria cooperativa de grupos de empresas. Coop-MPS é um método independente de PRO2PI, que atualmente utiliza características de PRO2PI-WORK no início do projeto, particularmente a matriz de identificação da importância e risco de cada área de processo.

A seguir estão descritos os relacionamentos dos resultados desta pesquisa, principalmente PRO2PI, com nove outras propostas. Estas propostas estão relacionadas na Figura 91, com a identificação da época que a proposta foi conhecida e então considerada na pesquisa. Esses relacionamentos são descritos em termos das concordâncias de PRO2PI com a proposta e as extensões de PRO2PI em relação às propostas.

A base de PRO2PI é o framework para modelos de capacidade de processo definido na ISO/IEC 15504 [2003]. PRO2PI estende a 15504 ao definir uma abordagem para estabelecimento de perfis de capacidade de processo para orientar a melhoria de processo. Tal abordagem está fora do escopo da 15504 e esta pesquisa considera fundamental o desenvolvimento de abordagens como PRO2PI para a evolução da melhoria de processo.

O método RAPID [Rout et al. 2000] definiu um perfil fixo baseado na ISO/IEC 15504-5 e influenciou PRO2PI ao definir uma avaliação rápida em micro e pequenas empresas como um meio para a melhoria. PRO2PI acrescenta a orientação para estabelecimento do perfil a ser avaliado, enquanto que no RAPID o perfil é fixo.

O CMMI [Chrissis et al. 2003] permite a utilização das duas arquiteturas de modelos de processo: estagiada e contínua. PRO2PI compartilha com o CMMI a consideração dos níveis de maturidade do estagiado como um determinado perfil de capacidade de processo do contínuo. Mas o

CMMI não permite a utilização da terceira geração: arquitetura contínua aberta. Em relação ao CMMI, PRO2PI permite a utilização de boas práticas de qualquer fonte, suportando portanto a terceira geração, além de prover a orientação para o estabelecimento de perfis.

A Abordagem Baseada em Problemas e Metas [Potter e Sakry 2002] aponta para as vantagens de utilização das metas e problemas de uma organização como orientação para a melhoria de processo, em complementação às orientações dos modelos do SW-CMM e CMMI. PRO2PI acrescenta a visão de perfis de capacidade de processo, baseado nas metas e problemas da organização, além de em outros fatores, como orientação para a melhoria. Com isto PRO2PI unifica as duas visões: a baseada em metas e problemas e a baseada em modelos.

O modelo iCMM utiliza a arquitetura contínua da ISO/IEC 15504, integra vários modelos e sugere níveis de maturidade fixos. Em relação ao iCMM, PRO2PI permite a utilização de boas práticas de qualquer fonte, além daqueles modelos integrados pelo iCMM, suportando portanto a terceira geração, além de prover a orientação para o estabelecimento de perfis.

*Evolving CMMI* [Ahern et al. 2001] representa as três alternativas de arquitetura para o CMMI consideradas pela equipe de desenvolvimento do CMMI. Todas elas são na verdade variações de ter apenas a arquitetura contínua. PRO2PI suporta estas três alternativas de arquitetura.

A abordagem *Constagedeous* [Kasse 2004] apresenta orientações para utilização da representação contínua do CMMI. Kasse considera os níveis de maturidade da representação estagiada do CMMI como as únicas possibilidades de perfis de capacidade, considerando outros perfis como intermediários para os níveis de capacidade.

A abordagem *CMMI in Small Settings Toolkit* [Garcia et al. 2005] apresenta técnicas para a seleção, entre as áreas de processo do CMMI, do subconjunto viável e mais apropriado para um micro ou pequeno empreendimento. Algumas destas técnicas são coincidentes com as técnicas já propostas em MEPI, como, por exemplo, analisar a importância e risco de cada área de processo para o empreendimento. PRO2PI utilizou duas técnicas da abordagem *CMMI in Small Settings Toolkit*: os sintomas e benefícios de cada área de processo, e as sugestões coletivas de práticas atuais e sugestões para a melhoria. *CMMI in Small Settings Toolkit*, como todas as propostas consultadas relacionadas ao CMMI, consideram apenas as duas primeiras gerações de arquitetura: estagiada fixa e contínua fechada. PRO2PI acrescenta o apoio para uso da terceira geração.

A metodologia OWPL [Habra et al. 1999, Habra et al. 2002, Laporte et al. 2005] definiu um perfil fixo baseado na ISO/IEC 15504-5 e SW-CMM, e divide o trabalho de avaliação em três

avaliações. PRO2PI acrescenta a orientação para estabelecimento do perfil a ser avaliado, enquanto que no OWPL, com também no RAPID, o perfil é fixo.

## 8.4 Limitações

Quatro limitações principais foram identificadas neste trabalho, relacionadas respectivamente ao uso dos níveis de capacidade de processo, validação qualitativa, propriedades e contexto de aplicação de PRO2PI-WORK.

Todo o trabalho utiliza como referência conceitual básica os seis níveis de capacidade de processo, conforme definidos na ISO/IEC 15504 e utilizado, por exemplo, nos modelos do CMMI, iCMM e MPS-BR. O nível 0 é apenas para referência inicial da escala, os níveis 1, 2 e 3 estão bem validados na prática, mas os níveis 4 e 5 ainda não podem ser considerados como validados. Existem até visões preliminares e informais de que esses níveis deveriam ser diferentes daqueles descritos atualmente.

Pela natureza do trabalho não é possível nem apropriado desenvolver uma abordagem quantitativa para validação do trabalho. É necessário estabelecer melhor a atual melhoria de processo de software, e também a engenharia de processo PCDE, para permitir uma análise quantitativa.

Foram exercitadas e analisadas de forma satisfatória cinco das oito propriedades de PRO2PI, que são as propriedades de relevante, viável, oportuno, rastreável e dinâmico. Faltam portanto experimentos para analisar e validar melhor as outras três propriedades, que são as propriedades de sistêmico, representativo e específico.

O método PRO2PI-WORK foi desenvolvido para micro e pequenas organizações com processos de baixa capacidade e não é adequado para médias e grandes organizações e nem para micro e pequenas organizações com processos de alta capacidade.

## 8.5 Trabalhos futuros

Um dos objetivos desta tese e ao mesmo tempo uma das contribuições é o estabelecimento de uma agenda de pesquisa e desenvolvimento em engenharia de processo dirigida por perfis de capacidade de processo. Esta agenda é composta pela seguinte relação de dezesseis propostas de trabalhos futuros:

TF1: continuação do desenvolvimento e utilização das oitos propriedades definidas em PRO2PI-PROP incluindo outras estruturas de medições para as propriedades como elementos de PRO2PI-MEAS;

- TF2: continuação do desenvolvimento e utilização do modelo PRO2PI-MODEL incluindo sua especificação como um metamodelo conforme o metamodelo MOF;
- TF3: continuação do desenvolvimento e utilização do ciclo de melhoria PRO2PI-CYCLE incluindo o desenvolvimento de ferramentas e artefatos de apoio;
- TF4: continuação do desenvolvimento e utilização do conjunto de medições PRO2PI-MEAS com outros tipos de medições, além daquelas relacionadas diretamente com as propriedades;
- TF5: continuação do desenvolvimento do método PRO2PI-WORK, incluindo o desenvolvimento, já iniciado, de ferramentas e outros artefatos de apoio;
- TF6: utilização de PCDE e PRO2PI em outras áreas intensivas em atividades humanas, como, por exemplo, a exploração inicial já realizada na área de educação em tecnologia onde foi definido uma área de processo para cursos de tecnologia e iniciado um método para definição de novas áreas de processo [Miranda 2005];
- TF7: realização de experiências de identificação do perfil de capacidade de processo que represente o processo de uma organização em um determinado momento e representação desse perfil como um PCP;
- TF8: exploração de uma análise semiótica dos fundamentos e elementos dos níveis de capacidade de processo, da engenharia de processo PCDE e da abordagem PRO2PI, como continuação da análise já iniciada em Salviano [2005], em direção a uma teoria semiótica desses fundamentos e elementos, seguindo a semiótica triádica peirciana (signo-objeto-interpretante, primeiridade-secundidade-terceiridade) [Peirce 1868, Peirce 1983, Santaella 2000] com o roteiro para “percurso para aplicação semiótica” [Santaella 2002] e a ferramenta  $\text{L}^*k/\text{GRASS}$  [Moor et al. 2002, apud Richmond e Udell 2004]
- TF9: Exploração do relacionamento do modelo PRO2PI-MODEL com as categorias universais de Peirce [Peirce 1868] e aos três tipos de conhecimento (remático, dicente e argumentativo) [Peirce 1883, Gudwin 1999] como uma base para um sistema inteligente com base nas três classes elementos de prática, grupo de práticas e sistema de práticas;
- TF10: exploração de outros métodos para determinação de perfil de capacidade de processo ou mesmo apenas para seleção de processo, a partir de PRO2PI, como, por exemplo, a abordagem MARES [Anacleto 2005], com participação do autor, e uma abordagem para seleção de processos [Faria 2005], sem a participação do autor;
- TF11: exploração da teoria de requisitos para revisão e consolidação de PRO2PI-PROP, como, por exemplo, as propriedades de qualidade de requisitos definidas como correto, viável, necessário,

---

priorizado, não ambíguo e verificável, para cada requisito individualmente completo, consistente, modificável, e rastreável para o conjunto de requisitos [Wiegiers 1999];

- TF12: . utilização de PRO2PI-MODEL para o desenvolvimento de um “super modelo, com super método de avaliação”, com a modelagem unificada dos elementos dos modelos mais relevantes e dos requisitos e elementos dos métodos de avaliação mais relevantes na linha dos trabalhos já iniciados de cooperação DMPS-CenPRA e LQPS-Univali [von Wangenheim e Salviano 2005] e descrição de processos da ISO/IEC 15504-5 em CMMI [Ferreira 2005];
- TF13: estabelecimento de um ambiente livre para evolução e exemplos de utilização de PCDE e PRO2PI, que seria estruturado com PCDE e PRO2PI e evoluído e utilizado pela comunidade, segundo uma abordagem semelhante à do software livre, sob uma coordenação, semelhante à coordenação do Linux;
- TF14: desenvolvimento de uma Linguagem de Padrões (*pattern language*) seguindo as orientações da comunidade de padrões [Alexander 1979, Alexander et al. 1977, Gamma et al. 1994, Coplien 1995, Coplien e Schmidt 1995, Gabriel 1998, Salviano 1998, Manns e Rising 2004] para comunicação dos fundamentos e elementos dos níveis de capacidade de processo, da engenharia de processo PCDE e da abordagem PRO2PI;
- TF15: pesquisa sobre as abordagens de pesquisa para esta área de engenharia de processo e outras áreas similares, incluindo a continuação, utilização e consolidação da abordagem utilizada nesta tese que incluiu (i) a abordagem “indústria-como-laboratório” (que pode ser renomeada para “pesquisa-pela-prática” para indicar melhor os papéis da pesquisa com a indústria), (ii) os próprios níveis de capacidade de processo, (iii) a abordagem de software patterns (que privilegia o registro e comunicação por meio de literatura de boas experiências em vez de formalizações prematuras) [Coplien 1995], (iv) uso de slides para registro e comunicação de conhecimento (como vem sendo feito por conferências na área de melhoria de processo, como, por exemplo, as conferências SEPG [SEI 2001c], SIMPROS [Salviano e Santana 2000] e CMMI [NDIA 2004]), sem a necessidade de elaboração de texto ou da própria apresentação, como uma alternativa, (v) a descrição do processo da pesquisa para compartilhar como a pesquisa foi realmente realizada, mais próxima do “pensamento lateral” [de Bono 1992], e ajudando a desfazer o “mito do projetista racional” já identificado por Parnas e Clements [1986], e (vi) sempre considerando o necessário rigor acadêmico na linha por exemplo das estratégias para pesquisa em engenharia de software compiladas e organizadas por Mary Shaw [2001, 2002]; e

TF16: desenvolvimento de hierarquia de perfis de capacidade de processo para segmentos ou domínios específicos. Um exemplo é o domínio de micro e pequenas empresas orientadas ao desenvolvimento, comercialização e evolução contínua de uma sistema.

## 8.6 Considerações finais

O poema *Lost* foi escrito por David Wagoner como uma orientação que um velho e sábio índio daria a um jovem índio para um conhecimento fundamental para eles, habitantes da floresta: como não se perder na floresta [Whyte 1996]. Podemos fazer uma analogia com a engenharia de software. A prática da concepção, construção e utilização de software criou a engenharia de software. O distanciamento entre teoria e prática, quando não tratamos a prática como “um estranho poderoso”, que “fez este lugar em volta de você”, talvez seja uma das principais causas de ainda não termos uma engenharia de software. Tratar a pesquisa em engenharia de software como “pesquisa-depois-transfere” é não “pedir permissão para conhece-lo e ser conhecido”. A abordagem de pesquisa “indústria-como-laboratório”, que talvez devesse ser chamada de “pesquisa-pela-prática”, reconhece que a prática da concepção, construção e utilização de software “sabe onde você está e que você tem que deixar que ela encontre você”.

A melhoria de processo de software entendeu melhor a prática de software e abstraiu níveis de maturidade que orientam o estabelecimento dos processos da engenharia de software em cada organização. Com isto acrescentou a dimensão de construção da engenharia de software pela comunidade da prática, seguindo a visão que software está diretamente relacionado a conhecimento e “ensinar não é transferir conhecimento, mas criar as possibilidades para a sua produção ou a sua construção”.

A arquitetura contínua oferece uma oportunidade de evolução da melhoria de processo, com a utilização da flexibilidade de estabelecimento do perfil de capacidade de processo mais adequado à realidade de uma organização, em vez de sempre utilizar os níveis de maturidade fixos. É importante para as organizações intensivas em software a capacidade de “ter de a todo momento escolher, com acerto e precisão, a melhor direção” sem ficar “esperando que façam por nós o que é nosso dever, recusar o poder”. A engenharia de processo PCDE e a abordagem PRO2PI apoiam esta escolha e expandem o uso de modelos, no caso como um perfil de capacidade de processo, que seja relevante, oportuno, viável, sistêmico, representativo, rastreável, específico e dinâmico. Todos os modelos são errados, no sentido que retratam uma visão parcial da realidade. Mas alguns são úteis, quando

orientam de forma satisfatória. Apesar do fato de que nenhum modelo possa fazer uma melhoria, nem impedir que uma melhoria seja feita, um modelo adequado pode ajudar bastante. Também apesar do fato de que nenhum modelo possa fazer ações equivocadas, e nenhum modelo possa impedir que você faça ações equivocadas, novamente um modelo adequado pode ajudar bastante.

Com o sucesso da melhoria de processo de software, exatamente onde a engenharia de software tem tido mais dificuldades, que é na redução da distância entre teoria e prática, podemos entender que a forma como estávamos vendo o problema (como transferir a teoria para a prática) é que era o problema.

## Referências bibliográficas

- [Abrahamsson 2002] Pekka Abrahamsson, The Role of Commitment in Software Process Improvement, Ph.D. Thesis, Department of Information Processing Science, University of Oulu, 116 pages, ISBN 951-42-6730-3, 2002. (disponível em <http://herkules.oulu.fi/isbn9514267303/>, acessado mais recentemente em 30/01/2003)
- [Abran et al. 2004] Alain Abran and James W. Moore (executive editors) Pierre Bourque, Robert Dupuis and Leonard L. Tripp (editors), Guide to the Software Engineering Body of Knowledge - SWEBOK, 2004 version, The Institute of Electrical and Electronics Engineers, Inc. - IEEE, 210 pages, 2004. (disponível em <http://www.swebok.org/>, acessado mais recentemente em 18/06/2004)
- [Adamonis et al. 2005] Andrius Adamonis, Antanas Mitašiūnas, Irmantas Naujikas, Saulius Ragaišis and Martynas Reingardtas, Dependencies of Process Capability Levels, ISSN 1392 – 124X Information Technology and Control, Vol. 34, No. 2A, 2005. (disponível em [itc.ktu.lt/itc342/Adamon342.pdf](http://itc.ktu.lt/itc342/Adamon342.pdf) acessado mais recentemente em 30/11/2005)
- [Aguiar 2002] Maurício Aguiar, PSM: Medição Prática de Software e Sistemas, 120 slides, em Anais do SIMPROS 2002 [Salviano et al. 2002].
- [Ahern et al. 2001] Dennis M. Ahern, Aaron Clouse, e Richard Turner, CMMI Distilled: A Practical Introduction to Integrated Process Improvement, SEI Series in Software Engineering, Addison-Wesley, 306 pages, 2001.
- [Alexander 1979] Christopher Alexander, The Timeless Way of Building, Oxford University, New York, 552 pages, 1979.
- [Alexander et al. 1977] Christopher Alexander, Sara Ishikawa, Murray Silverstein, Max Jacobson, Ingrid Fiksdahl-King and Shlomo Angel, A Pattern Language: Towns, Buildings and Construction, Oxford University, New York, 1171 pages, 1977.
- [Almeida 2005] Roner Cláudio de Oliveira Almeida, Avaliação e Melhoria de Processos Essenciais de uma micro-empresa baseada na ISO/IEC 15504, Monografia de conclusão do curso de Pós-Graduação Latu Senso em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2005.
- [Alves 1991] Rubem Alves, Conversas com quem gosta de ensinar, 26ª edição, Coleção polêmicas do nosso tempo, Cortez Editora e Editora Autores Associados, 1991.
- [Alves 1999] Rubem Alves, Entre a ciência e a sapiência – o dilema da educação, Edições Loyola, 1999.
- [Anacleto 2004] Alessandra Anacleto, “Método e modelo de avaliação para melhoria de processos de software em micro e pequenas empresas”, Dissertação de mestrado, Programa de pós-graduação em Ciência da Computação da Universidade Federal de Santa Catarina, 173 páginas, 2004.

- [Anacleto et al. 2003] Alessandra Anacleto, Christiane G. von Wangenheim, Clênio F. Salviano e Rafael Savi, “15504MPE- Desenvolvendo um Método para Avaliação de Processos de Software em MPEs Utilizando a ISO/IEC 15504”, em Anais do SIMPROS 2003 [Salviano e Santana 2003, p. 61-70].
- [Anacleto et al. 2004a] Alessandra Anacleto, Christiane G. von Wangenheim, Clênio F. Salviano e Rafael Savi, A Method for Process Assessment in Small Software Companies, in Proceedings of SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, p. 69-76, April 28-29, 2004.
- [Anacleto et al. 2004b] Alessandra Anacleto, Christiane G. von Wangenheim, Clênio F. Salviano e Rafael Savi, Experiences from Applying 15504 to Small Software Companies in Brazil, in Proceedings of SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, p. 33-37, April 28-29, 2004.
- [Appleton 2000] Brad Appleton, Patterns and Software: Essential Concepts and Terminology, disponível em [www.cmcrossroads.com/bradapp/docs/patterns-intro.html](http://www.cmcrossroads.com/bradapp/docs/patterns-intro.html), 2000.
- [April et al. 2004a] Alan April, Alain Abran and Reiner R. Dumke, SM<sup>CMM</sup> Model to Evaluate and Improve the Quality of Software Maintenance Process: Overview of the model, SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, p. 19-32, April 28-29, 2004.
- [April et al. 2004b] Alan April, J. Huffman Hayes, Alain Abran and Reiner R. Dumke, Software Maintenance Maturity Model (SM<sup>MM</sup>): The software maintenance process model, in Journal of Software Maintenance and Evolution: Research and Practice, vol. 17(3), 2005, pp. 197-223.
- [Armour 2003] Phillip Armour, Of Zeppelins and Jet Planes: Agile Development and the New Paradigm for Software, slides from presentation, 2003.
- [Automotive SIG 2005] Automotive SIG, Automotive SPICE Process Assessment Model, Final Release, v2.2, 144 pages, © The SPICE User Group, 21 August 2005 (disponível a partir de <http://www.automotivespice.com>, last accessed 09/11/2005).
- [Basili et al. 1999] Victor R Basili, F. Shull and F. Lanubile, Using Experiments to Build a Body of Knowledge, Proceedings of the Third International PSI Conference, Novosibirsk, Russia, pp. 265-282, July 1999.
- [Beitz et al. 1999] Andrew Beitz, Khaled El-Eman and Janne Jarvinen, A Business-Focus Assessment, National Research Council Canada NRC 43615 and Fraunhofer Institute Experimentelles Software Engineering, in Proceedings of The European Conference on Software Process Improvement (SPI), Barcelona, Spain, December 1999 ([iit-iti.nrc-cnrc.gc.ca/iit-publications-iti/docs/NRC-43615.pdf](http://iit-iti.nrc-cnrc.gc.ca/iit-publications-iti/docs/NRC-43615.pdf) May 5th, 2004).
- [Bernardo 2005] Claudio Bernardo, APS-FINAN – Um método baseado no SPICE para avaliação de processo de software de instituições financeiras, em Anais do SIMPROS 2005 [Salviano e Santana 2005].
- [Bézivin 1998] Jean Bézivin, Who's Afraid of Ontologies?, article presented at Model Engineering, Methods and Tools Integration with CDIF Workshop of OOPSLA 98, Vancouver, Canada, 18-22 October 1998. (disponível em [www.metamodel.com/oopsla98-cdif-workshop/bezivin1/](http://www.metamodel.com/oopsla98-cdif-workshop/bezivin1/) acessado mais recentemente em 27/07/2004)
- [Bézivin 2003a] Jean Bézivin, MDA<sup>TM</sup> : From Hype to Hope and Reality, Slides from a guess talk presentation at UML'2003, San Francisco, 96 slides, 2003 (disponível em: [www.sciences](http://www.sciences).

- univ-nantes.fr/info/perso/permanents/bezivin/UML.2003/UML.SF.JB.GT.ppt, acessado mais recentemente em 27/07/2004)
- [Bézivin 2003b] Jean Bézivin, Ingénierie des modèles logiciels, Slides from a course at Ecole d'Eté d'Informatique CEA EDF INRIA, 808 slides, 2003 (disponível em <http://aristotel.aristote.asso.fr/Presentations/CEA-EDF2003/Cours/JeanBezivin/IndexJeanBezivin.html>, acessado mais recentemente em 27/07/2004).
- [Bézivin 2003c] Jean Bézivin, On the Unification Power of Models, 31 pages, 2003 (disponível a partir de: [www.sciences.univ-nantes.fr/info/perso/permanents/bezivin](http://www.sciences.univ-nantes.fr/info/perso/permanents/bezivin), acessado mais recentemente em 27/07/2004).
- [Booch et al. 1999] Grady Booch, James Rumbaugh and Ivar Jacobson, The Unified Modeling Language User Guide, Addison Wesley Longman, 1999.
- [Bourque et al. 1999] Pierre Bourque, Robert Dupuis, Alain Abran, James W. Moore and Leonard Tripp, The Guide to the Software Engineering Body of Knowledge, IEEE Software, Vol. 16, no. 6, November/December 1999.
- [Box 1979] George Box, Robustness in the Strategy of Scientific Model Building, Robustness in Statistics, ed. By Launer, R. L. and Wilkinson, G. N., Academic Press, New York, 201-236, 1979 (apud Easterling 2001).
- [Brooks 1975] Frederick P. Brooks, Jr., The Mythical Man-Month: Essays on Software Engineering, Addison-Wesley, 1975 (reprinted with comments in [Brooks 1995]).
- [Brooks 1986] Frederick P. Brooks, Jr., No Silver Bullet, in Proceedings of the IFIP Tenth World Computing Conference, pp. 1069-1076, 1986 (reprinted with comments in Brooks 1995)).
- [Brooks 1995] Frederick P. Brooks, Jr., The Mythical Man-Month Essays on Software Engineering – Twenty anniversary edition with four new chapters, Addison-Wesley, 1995.
- [Buglione e Abran 2000] L. Buglione and A. Abran, Balanced Scorecards and GQM: what are the differences?, FESMA/AEMES 2000 Conference, October, 18-20, 2000, Madrid, Spain.
- [Campos 2005] Christina Aparecida R. Campos, Pojeto 'Software+ Rumo ao CMMI® Nível 2' – Um Relato de Experiência, Monografia de conclusão do curso de Pós-Graduação Latu Senso em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2005.
- [Capps 2003] Brent Capps, CSE 503 Software Engineering Processes, slides of course presentation, Department of Computer Science and Engineering of Oregon Health & Science University, Summer 2003. (disponível em <http://www.cse.ogi.edu/class/cse503/>, acessado mais recentemente em January 28, 2004)
- [Caputo 1998] Kim Caputo, CMM Implementation Guide: Choreographing Software Process Improvement, Addison Wesley, ISBN 0-201-37938-4, 319 pages, 1998.
- [Card 2002] David N. Card, Published Sources of Benchmarking Data, memorandum, 5 pages, Software Productivity Consortium, March 2002
- [Card 2004] David N. Card, Research Directions in Software Process Improvement, Proceedings of 28th International Computer Software and Applications Conference (COMPSAC 2004), Design and Assessment of Trustworthy Software-Based Systems, Hong Kong, China, IEEE Computer Society, p. 238, 27-30 September 2004.

- 
- [Card e Jones 2004] David N. Card and Cheryl L. Jones, Status Report: Practical Software Measurement, 2004.
- [Castro et al. 2005] Antônio Maria G. de Castro, Suzana Maria V. Lima e Jairo Eduardo Borges-Andrade, Metodologia de Planejamento Estratégico para as Unidades do Ministério da Ciência e Tecnologia, publicação do Ministério da Ciência e Tecnologia, 141 p., 2005.
- [Cater-Steel et al. 2004] Aileen Cater-Steel, Mark Toleman and Terry Rout, After the Assessment: Actions and Reactions of 22 Small Australian Firms, in Proceedings of SPICE 2004 The Fourth International SPICE Conference, Lisbon, Portugal, p. 54-63, April 28-29, 2004.
- [Cavalcante e Costa 2005] Kelly Vieira Cavalcante e Roberta Passeti da Costa, Um Método para Especialização de Modelos de Capacidade de Processo para Domínios e sua aplicação na Especialização de Quatro Áreas de Processo do CMMI para o Domínio Bancário, Monografia de conclusão do curso de especialização de Qualidade no Desenvolvimento de Software da Faculdade Senac de Ciências Exatas e Tecnologia, 2005.
- [CCSDS 2001] Consultative Committee for Space Data Systems, Reference Model for an Open Archival Information System (OAIS), CCSDS 650.0-R-1.2, Red Book, June 2001 (disponível em <http://ssdoo.gsfc.nasa.gov/nost/wwwclassic/documents/pdf/CCSDS-650.0-R-2.pdf>, acessado mais recentemente em 16/07/2004)
- [Chrissis et al. 2003] Mary Beth Chrissis, Mike Konrad and Sandy Shrum, CMMI: Guidelines for Process Integration and Product Improvement, Addison-Wesley Pub Co, 2003.
- [Cockburn 2004] Alistair Cockburn, The end of software engineering and the start of Economic-Cooperative Gaming, Humans and Technology, HaT TR 2004.02, 21 pages, January 14<sup>th</sup>, 2004. (disponível a partir de <http://alistair.cockburn.us>, acessado mais recentemente em 26/07/2005)
- [Conradi e Fuggetta 2002] Reidar Conradi and Alfonso Fuggetta, Improving Software Process Improvement, IEEE Software, 19(4), p. 92-99, July/August 2002.
- [Coplien 1995] James O. Coplien, Software Patterns, SIGS Books & Multimedia, Ney York, ISBN 1-884842-50-X, 69 pages, 1995.
- [Coplien e Schmidt 1995] James O. Coplien and Douglas Schmidt (eds.), Pattern Languages of Program Design, Addison-Wesley, 562 pages, 1995.
- [Covey 1989] Stephen R. Covey, The 7 Habits of Highly Effective People, Fireside, 1989.
- [Crawford 2002] J. Kent Crawford, Project Management Maturity Model: Provinding a Proven Path to Project Management Excellence, Center for Business Practice, 2002.
- [Crespo et al. 2004] Adalberto Nobiato Crespo, Odair Jacinto da Silva, Carlos Alberto Borges, Clênio F. Salviano, Miguel de Teive e Argollo Junior e Mario Jino, Uma Metodologia para Teste de Software no Contexto da Melhoria de Processo, em *Anais do SBQS 2004: Terceiro Simpósio Brasileiro de Qualidade de Software*, pp. 271-285, 2004.
- [Croll 2003] Paul R. Croll, Eight Steps to Success in CMMI - Compliant Process Engineering: Strategies and Supporting Technology, slides from presentation at Third Annual CMMI Technology Conference and Users Group, USA, November 2003.
- [Curtis 1998] Bill Curtis, "Which Comes First, the Organization or its Processes", IEEE Software, pages 10-13, November/December 1998 (disponível em <http://www.teraquest.com/articles>).

- [DACS 1999] Data & Analysis Center for Software, A Business Case for Software Process Improvement Revised - Measuring Return on Investment from Software Engineering and Management, A DACS State-of-the-Art Report, Contract Number SP0700-98-4000, Prepared By Thomas McGibbon, 30 September 1999. (disponível em <http://www.dacs.dtic.mil/techs/roispi2/>, acessado mais recentemente em 07/06/2004)
- [de Bono 1995] Edward de Bono, Serious Creativity – Using the Power of Lateral Thinking to Create New Ideas, HarperBusiness, 1995.
- [de Petri et al. 2005a] Flávia de Petri, Juliana Rodrigues, Luiz Mapelli e Vera Lúcia Oliveira, Proposta de Melhoria de Processos de Software na Empresa Fibam Companhia Industrial aplicando MR-MPS, Trabalho de conclusão da disciplina Modelos de Qualidade de Software do curso de Pós-Graduação em Engenharia de Software da Universidade São Judas Tadeu, São Paulo, SP, 10 páginas, maio de 2005.
- [de Petri et al. 2005b] Flávia de Petri, Juliana Rodrigues, Luiz Mapelli, Vera Lúcia Oliveira and Clênio F. Salviano, Uma experiência de capacitação e início de melhoria de processo de software com o método PRO2PI-WORK, em Anais do SIMPROS 2005 [Salviano e Santana 2005], 12 páginas.
- [Deming 1986] Edward W. Deming, W. Edward, Out of the Crisis, Cambridge. MIT Center for Advanced Engineering Study, 1986.
- [Dorling 1995] Alec Dorling (SPICE Project Manager), SPICE – Software Process Assessment, version 1.0, document set with nine parts: SPICE Part 1 to SPICE Part 9, 1995 (disponível em [www.sqi.gu.edu.au/spice](http://www.sqi.gu.edu.au/spice) in Document Suite, acessado mais recentemente em 07/11/2005).
- [Dorling 2004] Alec Dorling, Automotive SPICE, keynote presentation at SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, April 28-29, 2004.
- [Dorling et al. 2003] Alec Dorling, Patrizia Secchi and Christian Völcker, The application of SPICE to ISO 9000:2000 and ECSS-Q-20, in Proceedings of SPICE 2003 The Third International SPICE Conference, Noodwijk, The Netherlands, Italy, pp. 31-39, 2003.
- [Drucker 1992] Peter Drucker, Managing for the future: The 1990s and beyond, 1992.
- [Earthy 2000] Jonatham Earthy, Usability Maturity Model: Processes, Technical Report, 2000. (disponível em <http://www.idemployee.id.tue.nl/g.w.m.rauterberg/lecturenotes/Usability-Maturity-Model%5B2%5D.PDF>)
- [Easterling 2001] Robert G. Easterling, Quantifying the Uncertainty of Computational Predictions, Proceedings of the 2001 SEM Annual Conference, June 4-6, 2001
- [EFQM 2003] European Foundation for Quality Management, Os Conceitos Fundamentais da Excelência, 2003. (disponível em [www.efqm.org](http://www.efqm.org))
- [Ehms e Langen 2002] Karsten Ehms and Manfred Langen, Holistic Development of Knowledge Management with KMMM, Siemens AG, 2002.
- [El Emam 2001] Khaled El Emam, Software Engineering Process, Chapter 9 of Guide to the Software Engineering Body of Knowledge (SWEBOK), Trial (Version 0.95), pages 9-1 to 9-13, 2001. (disponível em <http://www.swebok.org/>, acessado mais recentemente em 08/05/2004)

- 
- [El Emam 2003] Khaled El Emam, What Have we Learned: Return-on-Investment from the SW-CMM, slides from presentation at CMMI Conference 2003, [www.dtic.mil/ndia/2003CMMI/Bene.ppt](http://www.dtic.mil/ndia/2003CMMI/Bene.ppt).
- [El Emam 2004] Khaled El Emam, Software Engineering Process, Chapter 9 of Guide to the Software Engineering Body of Knowledge (SWEBOK), 2004 version, [Abran et al. 2004], pages 9-1 to 9-14, 2004. (disponível em <http://www.swebok.org/>, acessado mais recentemente em 18/06/2004)
- [El Emam e Garro 1999] Khaled El Emam and Iñigo Garro, Estimating the Extent of Standards Use: The Case of ISO/IEC 15504 Technical Report, 1999.
- [El Emam et al. 1998] Khaled El Emam, Jean-Normand Drouin and Walcélcio Melo, SPICE – The Theory and Practice of Software Process Improvement and Capability Determination. IEEE Computer Society, 1998.
- [ESI 1996] European Software Institute, SPICE Assessor Training Material, Part I and II, SPICE-TI-95-44, 1996
- [ESI 1999] European Software Engineering, Software Process Improvement, slides from presentation by Giuseppe Magnani at SIMPROS 1999 [Salviano e Santana 1999].
- [ESI 2005] Embedded Systems Institute, Boderc project web site, from [www.esi.nl](http://www.esi.nl), acessado mais recentemente em 30/11/2005.
- [Estier 1998] Thibault Estier, What is BNF notation?, Web-page, University of Geneva, in <http://cui.unige.ch/db-research/Enseignement/analyseinfo/AboutBNF.html>, 1998 (acessado mais recentemente em 17/07/2004)
- [Evers e Miller 2003] John Evers and Chris Miller, Systems Engineering Process Improvement in a Multi-Model Environment, in Thirteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE), Crystal City, Virginia, July 1-3 2003 (disponível em <http://www.software.org/pub/externalpapers/>, accessed in 29/12/2003).
- [Faria 2005] Frederico Faria, Um método para seleção de processos de software em programas de melhoria: caso de uma pequena empresa do setor de telecomunicações, Monografia de conclusão do curso de Pós-Graduação Latu Senso em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2005
- [Favre 2005a] Jean-Marie Favre, Foundations of Model (Driven) (Reverse) Engineering: Models, Episode I: Stories of The Fidus Papyrus and of The Solarus, 31 pages, Pilot of the series “From Ancient Egypt to Model Driven Engineering”, disponível em <http://www-adele.imag.fr/mda,2005>. (acessado mais recentemente em 20/03/2006)
- [Favre 2005b] Jean-Marie Favre, Foundations of Meta-Pyramids: Languages vs. Metamodels, 28 pages, Episode II of the series “From Ancient Egypt to Model Driven Engineering”, disponível em <http://www-adele.imag.fr/mda,2005>. (acessado mais recentemente em 20/03/2006)
- [Favre 2005c] Jean-Marie Favre, Megamodelling and Etymology – A story of words: from MED to MDE via MODEL in five millenniums, 22 páginas, an episode of the series “From Ancient Egypt to Model Driven Engineering”, disponível em <http://www-adele.imag.fr/mda,2005>. (acessado mais recentemente em 20/03/2006)
- [Ferreira 2005] Jean de Souza Ferreira, Descrevendo os processos do grupo de reuso da ISO/IEC 15504-5 para o padrão do CMMI, Monografia de conclusão do curso de especialização de

- Qualidade no Desenvolvimento de Software da Faculdade Senac de Ciências Exatas e Tecnologia, 2005.
- [Florac e Carleton 1999] Florac and Carleton, Measuring the Software Process – Statistical Process Control for Software Process Improvement, 1999
- [FPNQ 2005] Fundação Prêmio Nacional da Qualidade, PNQ 2006 – Critérios de Excelência, 64 páginas, Novembro de 2005. (disponível em [www.fpnq.org.br](http://www.fpnq.org.br))
- [Freire 1996] Paulo Freire, Pedagogia da autonomia – Saberes necessários à prática educativa, Coleção Leitura, Editora Paz e Terra, 1996.
- [Fuggetta 2000] Alfonso Fuggetta, Software Process: A Roadmap, in Proceedings of Future of Software Engineering Workshop, ICSE, Limerick, Ireland, pp. 25-34, 2000.
- [Gabriel 1998] Richard P. Gabriel, Patterns of Software: Tales from the Software Community, 256 pages, Oxford University Press, April, 1998.
- [Gamma et al. 1994] Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994.
- [Garcia 2005] Suzane Garcia, Thoughts on Applying CMMI in Small Settings, slides from presentation, SEI CMU, 2005 (disponível em [www.sei.cmu.edu/cmmi](http://www.sei.cmu.edu/cmmi) acessado mais recentemente em 06/10/2005)
- [Garcia et al. 2004] Suzane Garcia, Sandra Cepeda, Gene Miluk and Mary Jo Staley, Adopting CMMI for Small Organizations, slides from presentation at Fourth Annual CMMI Technology Conference and Users Group, Denver, USA, November 2004. (disponível em <http://www.dtic.mil/ndia/2004/CMMIT2Mon/110504Cepeda.pdf>, acessado mais recentemente em 17/02/2005)
- [Garcia et al. 2005] Suzane Garcia, Sandra Cepeda, Gene Miluk and Mary Jo Staley, CMMI in Small Settings Toolkit Repository from AMRDEC SED Pilot Sites; Draft 14, Software Engineering Institute, 2005. (disponível em <http://www.sei.cmu.edu/ttp/publications/toolkit>, acessado mais recentemente em 17/02/2005)
- [Garg 1996] Pankaj K. Garg, Process-Centered Software Engineering Environments, IEEE Computer Society Press, Los Alamitos, CA, 1996
- [Glass 2003] Robert L. Glass, The State of the Practice of Software Engineering, Guest editor's introduction, p. 20-21, IEEE Software, November/December 2003.
- [Haavikko et al. 2004] Matti Haavikko, Mika Johansson, Timo Makinen, Risto Nevalainen, Ilmari Saastamoinen, Markku Tukianen and Timo Varkoi, Gnosis – An Approach for SPI Knowledge management, in Proceedings of SPICE 2004 The Fourth International SPICE Conference, Lisbon, Portugal, p. 91-97, April 28-29, 2004.
- [Habra e Alexandre 2003] Naji Habra and Simon Alexandre, UML Modeling of Five Process maturity Models, Version 1, Technical Report LQL-2003-TR-02, CETIC-FUNDP, 42 pages, 2003.
- [Habra et al. 1999] Naji Habra, Eustache Nyitugabira, Anne-Catherine Lamblin and Alain Renault, Software Process Improvement for Small Structures: First Results of a Micro-Assessment Framework, in Proceedings of the European Conference on Software Process Improvement SPI99, Barcelona, Spain, December 1999.

- 
- [Habra et al. 2002] Naji Habra, Alain Renault, Simon Alexandre and Miguel Lopez, OWL Micro Assessment, in Proceedings of the Workshop on Software Quality, International Conference on Software Engineering ICSE 2002, Orlando, Florida USA, May 2002.
- [Halvorsen e Conradi 1999] Christian Printzell Halvorsen and Reidar Conradi, A Taxonomy of SPI Frameworks, in Proceedings of SEW 99 – 24th NASA/ IEEE Software Engineering Workshop, 4 pages, 1999 (disponível em [sel.gsfc.nasa.gov/website/sew/1999/topics/halvorsen\\_SEW99paper.pdf](http://sel.gsfc.nasa.gov/website/sew/1999/topics/halvorsen_SEW99paper.pdf), acessado mais recentemente em 11/05/2004)
- [Harrison et al. 2001] Warren Harrison, John Settle and David Rafflo, Assessing the Value of Improved Predictability Due to Process Improvements, paper presented at the Third International Workshop on Economics-Driven Software Engineering Research, Toronto, Canada, 2001. (disponível em <http://www.cs.virginia.edu/~sullivan/edser3/harrison.pdf>, acessado mais recentemente em 07/06/2004).
- [Hart 1997] Jim D. Hart, Introduction to Software Process and Process Improvement, Innovation Dynamics Consulting, Inc, November 1999.( disponível em IDC Articles and Papers web-site)
- [Herbsleb et al. 1994] James Herbsleb, A. Carleton, J. Rozum, J. Siegel, David Zubrow, Benefits of CMM-Based Software Process Improvement: Initial Results CMU/SEI-94-TR-013, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA: 1994.
- [Herbsleb et al. 1997] James Herbsleb, David Zubrow, Dennis Goldenson, Will Hayes and Mark Paulk, Software Quality and the Capability Maturity Model, in Communication of the ACM, Vol. 40, No. 6, pp. 30-40, June 1997.
- [Humphrey 1988] Watts S. Humphrey, Characterizing the Software Process: A Maturity Framework, in IEEE Software, Vol. 5, No. 2, pp. 73-79, March 1988. [reprinted in Software State-of-the-art: Selected papers, edited by Tom DeMarco and Timothy Lister, Dorset House Publishing, pp. 62-75, 1990]
- [Humphrey 1989] Watts S. Humphrey, Managing the Software Process, Addison Wesley, Reading, Mass. USA, 1989.
- [Humphrey 1994] Watts S. Humphrey, Process Maturity Model, topic in Encyclopedia of Software Engineering, John J. Marciniak (editor), John Wiley & Sons, pp. 851-860, 1994.
- [Humphrey 1999] Watts S. Humphrey, Putting Engineering into Software, 41 slides, Carnegie Mellon University, 1999.
- [Humphrey 2004] Albert S. Humphrey, The origins of the SWOT analysis model, in SWOT Analysis, by Alan Chapman, [www.businessballs.com](http://www.businessballs.com), 2004.
- [Hyder et al. 2004a] Elaine B. Hyder, Keith M. Heston and Mark Paulk, The eSourcing Capability Model for Service Providers (eSCM-SP) v2 : Part 1: Model Overview, Information Technology Services Qualification Center (ITsqc), Carnegie Mellon University, Technical Report No. CMU-ISRI-04-113, 2004.
- [Hyder et al. 2004b] Elaine B. Hyder, Keith M. Heston and Mark Paulk, The eSourcing Capability Model for Service Providers (eSCM-SP) v2 : Part 2: Practice Details, Information Technology Services Qualification Center (ITsqc), Carnegie Mellon University, 2004.
- [Ibrahim 2000] Linda Ibrahim, Using an Integrated Capability Maturity Model® – The FAA Experience, in Proceedings of the Tenth Annual International Symposium of the International Council on Systems Engineering (INCOSE), Minneapolis, Minnesota, pp. 643-648, July 2000.

- [Ibrahim et al. 2001] Linda Ibrahim, Bill Bradford, David Cole, Larry LaBruyere, Heidi Leinneweber, Dave Piszczek, Natalie Reed, Mike Rymond, Dennis Smith, Michael Virga and Curt Wells, The Federal Aviation Administration Integrated Capability Maturity Model® (FAA-iCMM®), Version 2.0, An Integrated Capability Maturity Model for Enterprise-wide Improvement, Published by the Federal Aviation Administration, 480 pages, September 2001 (disponível em <http://www.faa.gov/aio/ProcessEngr/iCMM/26/08/2004>)
- [infomap 2005] Infomap, Infomap web site, [www.infomap.com](http://www.infomap.com), 2005.
- [ISO 9000 2000] The International Organization for Standardization - Associação Brasileira de Normas Técnicas, NBR ISO 9000:2000 – Sistemas de gestão da qualidade – Fundamentos e vocabulário, Rio de Janeiro, 2000.
- [ISO 9001 2000] The International Organization for Standardization - Associação Brasileira de Normas Técnicas, NBR ISO 9001:2000 – Sistemas de gestão da qualidade – Requisitos, Rio de Janeiro, 2000.
- [ISO 9004 2000] The International Organization for Standardization - Associação Brasileira de Normas Técnicas, NBR ISO 9004:2000 – Sistemas de gestão da qualidade – Diretrizes para melhorias de desempenho, Rio de Janeiro, 2000.
- [ISO/IEC 12207 1998] Associação Brasileira de Normas Técnicas, NBR ISO/IEC 12207 - Tecnologia de Informação - Processos de ciclo de vida de software, Rio de Janeiro, 1998.
- [ISO/IEC 12207 2002] The International Organization for Standardization, ISO/IEC 12207 Information Technology - Amendment to ISO/IEC 12207. Montreal: ISO/IEC JTC1 SC7, 2002
- [ISO/IEC 15504 2003] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment, International Standard (IS) with five parts [ISO/IEC 15504-1 2004, ISO/IEC 15504-2 2003, ISO/IEC 15504-3 2004, ISO/IEC 15504-4 2004, ISO/IEC 15504-5 2006], 2003.
- [ISO/IEC 15504-1 2004] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 1: Concepts and Vocabulary, 2004.
- [ISO/IEC 15504-2 2003] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 2, 2003.
- [ISO/IEC 15504-3 2004] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 3: Guidance on performing an assessment, 2004.
- [ISO/IEC 15504-4 2004] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 4: Guidance on using assessment results, 2004.
- [ISO/IEC 15504-5 2006] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 5: An exemplar Process Assessment Model, 2006.
- [ISO/IEC 15939 2002] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15939 – Software engineering – Software measurement process, 37 pages, 2002.

- 
- [ISO/IEC FDIS 15504-5 2005] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC FDIS 15504 - Information Technology - Process Assessment – Part 5: An exemplar Process Assessment Model, ISO/IEC SC7 JTC1 WG10 document WG10N533 ISO-IEC\_FDIS\_15504-5, Final Disposition International Standard FDIS, 2005.
- [ISO/IEC PDTR 15504-5 1995] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC PDTR 15504 - Information Technology – Software Process Assessment – Part 5: An exemplar Process Assessment Model, ISO/IEC SC7 JTC1 WG10 document, 1995.
- [ISO/IEC TR 15504 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504 - Information Technology - Software Process Assessment, document set with nine parts: ISO/IEC TR 15504-1 to ISO/IEC TR 15504-9, 1998.
- [ISO/IEC TR 15504-1 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-1 - Information Technology - Software Process Assessment - Part 1 : Concepts and introductory guide, Technical Report, 1998.
- [ISO/IEC TR 15504-2 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-2 - Information Technology - Software Process Assessment - Part 2 : A reference model for processes and process capability, Technical Report, 1998.
- [ISO/IEC TR 15504-3 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-3 - Information Technology - Software Process Assessment - Part 3 : Performing an assessment, Technical Report, 1998.
- [ISO/IEC TR 15504-4 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-4 - Information Technology - Software Process Assessment - Part 4 : Guide to performing assessments, Technical Report, 1998.
- [ISO/IEC TR 15504-5 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-5 - Information Technology - Software Process Assessment - Part 5 : An assessment model and indicator guidance, Technical Report, 1998.
- [ISO/IEC TR 15504-6 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-6 - Information Technology - Software Process Assessment - Part 6 : Guide to competency of assessors, Technical Report, 1998.
- [ISO/IEC TR 15504-7 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-7 - Information Technology - Software Process Assessment - Part 7 : Guide for use in process improvement, Technical Report, 1998.
- [ISO/IEC TR 15504-8 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-8 - Information Technology - Software Process Assessment - Guide for use in determining supplier process capability, Technical Report, 1998.
- [ISO/IEC TR 15504-9 1998] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-9 - Information Technology - Software Process Assessment - Part 9 : Vocabulary, Technical Report, 1998.
- [ITGI 2005] IT Governance Institute, COBIT® 4.0 - Control Objectives for Information and Related Technology, [www.itgi.org](http://www.itgi.org), 2005

- [Jackson 2004] Michael Jackson, Problem Frames and Software Engineering, 16 pages, December 2004. (disponível em Jackson web site)
- [Jones 2003] Caper Jones, Variations in Software Development Practices, p. 22-27, IEEE Software, November/December 2003.
- [Kaplan e Norton 1996] Robert S. Kaplan and David P. Norton, The Balanced Scorecard, Harvard Business School Press, 1996.
- [Kaplan e Norton 2001] Robert S. Kaplan and David P. Norton, The Strategy-Focused Organization, Harvard Business School Press, 2001.
- [Kasse 2004] Tim Kasse, Practical Insight into CMMI, Artech House Publishers, 281 pages, 2004.
- [Kasse e McQuaid 2000] Tim Kasse and Patricia A. McQuaid, Entry Strategies Into the Process Improvement Initiative, Technical Report, 20 pages, 2000. (disponível a partir de [www.kasse.com](http://www.kasse.com))
- [Kent 2002] Stuart Kent, Model Driven Engineering (MDE), presentation, 36 slides, May 2002. (disponível em <http://www.cs.kent.ac.uk/projects/kmf/Documents/ModelDrivenEngineeringTurkuMay02.pdf>, acessado mais recentemente em 27/9/2004)
- [Kinnula 1999] Atte Kinnula, Software Process Engineering in a Multisite Environment: an Architectural Design of a Software Process Engineering System, PhD thesis, Department of Information Processing Science, University of Oulu, ISBN 951-42-5303-5, 119 pages, 1999. (disponível em <http://herkules.oulu.fi/isbn9514253035/>, acessado mais recentemente em 28/06/2004)
- [Kinnula 2001] Atte Kinnula, Software Process Engineering Systems: Models and Industry Cases, manuscript of Department of Information Processing Science, University of Oulu, ISBN 951-42-6508-4, 117 pages, September 2001. (disponível em <http://herkules.oulu.fi/isbn9514265084/>, acessado mais recentemente em 28/06/2004)
- [Kohan 2003] Sarah Kohan, QuickLocus: Proposta de um método de avaliação de processo de desenvolvimento de software em pequenas organizações, Dissertação de mestrado, Instituto de Pesquisas Tecnológicas do Estado de São Paulo - IPT, 234 páginas, 2003.
- [Laporte et al. 2005] Laporte, C.Y.; Renault, A.; Desharnais, J. M.; Habra, N.; Abou El Fattah, M.; Bamba, J.C. , Initiating Software Process Improvement in SMEs: Experiment with CETIC's Micro-Evaluation , in SWEDC-REK, International Conference on Software Development, Reykjavik, Islande, University of Iceland, pp. 153-164, 2005. (<http://www.gelog.etsmtl.ca/publications.html>)
- [Lepasaar e Mäkinen 2002] Marion Lepasaar and Timo Mäkinen, Integrating Software Assessment Models using a Process Meta Model, in the Proceedings of IEMC 2002: Volume I, pp. 224-229, Cambridge, UK, 2002.
- [Lepasaar et al. 2002] Marion Lepasaar, Timo Mäkinen and Timo Varkoi, Structural comparison of SPICE and continuous CMMI, in Proceedings of SPICE 2002: The Second International Conference on Software Process Improvement and Capability Determination, Venice, Italy, March 2002.
- [Lieberherr et al. 1994] Karl J. Lieberherr, Ignacio Silva-Lepe and Cun Xiao, Adaptive Object-Oriented Programming – Using Graph-Based Customization, Communication of ACM, 37(5), pp. 94-101, May 1994.

- 
- [Lobo 2004] Jan Eduardo Lobo, Experiência de Avaliação de Processos e Desenvolvimento de uma Ferramenta para Apoio Baseado na ISO/IEC TR 15504, Monografia de conclusão do curso de Pós-Graduação Lato Sensu em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2004.
- [Lobo e Salviano 2005] Jan Eduardo Lobo e Clênio F. Salviano, Experiência de Avaliação de Processos e Desenvolvimento de uma Ferramenta para Apoio Baseada na ISO/IEC TR 15504, Revista ProQualiti – Qualidade na Produção de Software, Núcleo de Estudos Avançados em Engenharia e Qualidade de Software, UFLA, Volume 1, Número 1, Maio 2005, ISSN 1807-5061.
- [Machado 2003] Cristina Ângela F. Machado, Definindo processos do ciclo de vida de software usando a Norma ISO/IEC 12207, publicação do curso de pós-graduação “Lato Sensu” / (Especialização) a distância em melhoria de processo de software, UFLA/FAEPE, 2003.
- [Mäkinen e Varkoi 2003] Timo Mäkinen and Timo Varkoi, Exposing Reference Model Associations, in Proceedings of SPICE 2003: The Joint ESA - Third International SPICE Conference on Process Assessment and Improvement, ESTEC, Noordwijk The Netherlands, 17-21 March 2003.
- [Manns e Rising 2004] Mary Lynn Manns and Linda Rising, Fearless Change: Patterns for Introducing New Ideas, 256 pages, Addison-Wesley Professional, October, 2004.
- [Marciniak 1994] John J. Marciniak (editor), Encyclopedia of Software Engineering, Volumes 1 and 2, John Wiley & Sons, 11453 pages, 1994.
- [McFeeley 1996] Bob McFeeley, IDEAL - A User's Guide for Software process Improvement, Handbook CMU/SEI-96-HB-001, 236 pages, 1996.
- [McGarry et al. 2002] John McGarry, David Card, Cheryl Jones, Beth Layman, Elizabeth Clark, Joseph Dean, Fred Hall, Practical Software Measurement: Objective Information for Decision Makers, PSM, Addison Wesley Professional, ISBN: 0201715163, 304 p., 2002.
- [Metamodel 2005] Metamodel, Metamodel web site at [www.metamodel.com](http://www.metamodel.com), 2005.
- [Miranda 2005] Adriana Pereira da Cruz Miranda, Uma abordagem com a ISO/IEC 15504 (SPICE) para melhoria no processo de ensino de cursos da área de informática em um Centro de Educação Profissional – SENAC/GO, Monografia de conclusão do curso de Pós-Graduação Lato Sensu em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2005.
- [MIT e Softex 2002] Massachusetts Institute of Technology and Brasil Coordenação Geral Sociedade SOFTEX, A Indústria de Software no Brasil 2002: Fortalecendo a Economia do Conhecimento, disponível em [www.softex.br](http://www.softex.br) (acessado em 16/12/2003), ISBN 85-89477-01-0, 80p., 2002.
- [ModelWare 2004] MODELLing solution for softWARE systems, IST FP6 Call 2, June 2004, <http://www.artist-embedded.org/PastEvents/Rome04/Tuesday/ETP-Rome04-Potier-MODELWARE.pdf>. (acessado mais recentemente em June 2004)
- [Mutafelija e Stromberg 2003] Boris Mutafelija and Harvey Stromberg, Systematic Process Improvement Using ISO 9001:2000 and CMMI, Artech House Computing Library, 312 pages, ISBN: 1580534872, 2003
- [Myers 2003] Chuck Myers, Over, Under, Around and Through: Using the CMMI Staged and Continuous Representation Together to Optimize Process Improvement, slides from presentation at SEPG Conference 2003.

- [Naur e Randell 1969] Peter Naur and Brian Randell (Editors), Software Engineering: Report on a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 October 1968, 136 pages, January 1969. (uma versão eletrônica do relatório original, editada por Robert M. McClure em 2001, está disponível em <http://homepages.cs.ncl.ac.uk/brian.randell/NATO/>, acessado mais recentemente em 10/05/2004)
- [NDIA 2004] National Defense Industrial Association, Proceedings of The Fourth Annual CMMI Technology Conference and Users Group (published as a CD), Denver, USA, November 2004. (disponível em <http://www.dtic.mil/ndia/2004>, acessado mais recentemente em 17/02/2005)
- [Nicoletti e Salviano 2003] Adilson S. Nicoletti e Clênio F. Salviano, An Experience using ISO/IEC TR 15504 and ISO 9000:2000 for Software Process Improvement, Proceedings of SPICE 2003 The Third International SPICE Conference, Nordwick, Netherlands, pp. 141-142, March 2003.
- [O'Toole 2000] Patrick O'Toole, Do's and Don'ts of Software Process Improvements, slides from tutorial presented at SIMPROS 2000 [Salviano e Santana 2000].
- [O'Toole 2006] Patrick O'Toole, Message 5138 in Yahoo CMMI Process Improvement Group, [http://groups.yahoo.com/group/cmmi\\_process\\_improvement](http://groups.yahoo.com/group/cmmi_process_improvement), January, 5<sup>th</sup>, 2005.
- [Oliveira 2004] Francisco Márcio de Oliveira, Estudo de um Modelo de Qualidade de Processos Aplicável ao Desenvolvimento de Software em um Tribunal Eleitoral Baseado no ISO/IEC TR 15504, Monografia de conclusão do curso de Pós-Graduação Latu Senso em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2004.
- [Olson 2003] Timothy G. Olson, Staged or Continuous: Which Model Should I Choose?, slides from presentation at Third Annual CMMI Technology Conference and Users Group, USA, November 2003.
- [Olson 2004] Timothy G. Olson, How to Define Short, Usable Processes: A Review of Process Definition Best Practices, slides from presentation at Fourth Annual CMMI Technology Conference and Users Group, Denver, USA, November 2004.
- [Olson et al. 1994] Timothy G. Olson, Neal R. Reizer and James W. Over, A Software Process Framework for the SEI Capability Maturity Model, Technical Report CMU/SEI-94-HB-01 September 1994.
- [OMG 2002a] Object Management Group, Meta Object Facility (MOF) Specification, version 1.4, OMG Specification, 358 pages, April 2002.
- [OMG 2002b] Object Management Group, Software Process Engineering Metamodel Specification (SPEM), version 1.0, OMG Specification, 98 pages, November 2002.
- [OMG 2002c] Object Management Group, Metamodel and UML Profile for Java and EJB, version 1.0), OMG Specification, February 2002. (disponível em <http://www.omg.org/cgi-bin/doc?formal/2004-02-02>)
- [OMG 2003a] Object Management Group, OMG Unified Modeling Language (UML) Specification, version 1.5, OMG Specification, 736 pages, March 2003.
- [OMG 2003b] Object Management Group, "Catalog of OMG Modeling and Metadata Specifications", OMG, v. 1.4, Needham, MA, USA, October 2003. (disponível em [http://www.omg.org/technology/documents/modeling\\_spec\\_catalog.htm](http://www.omg.org/technology/documents/modeling_spec_catalog.htm), acessado mais recentemente em 20/12/2003)

- 
- [Osterweil 1987] Leon J. Osterweil, Software Process are Software too, in Proceedings of the Ninth ICSE, pp. 2-13, Monterey CA, March 1997.
- [Osterweil 1997] Leon J. Osterweil, Software Process are Software too, Revised: An Invited Talk on the Most Influential Paper of ICSE9, in Proceedings of ICSE, pp. 540-548, 1997.
- [Palmer 1995] Kent D. Palmer, Advanced Process Architectures, presented at SEPG 1995, 302 slides, 1995.
- [Parnas e Clements 1986] David Parnas and Paul Clements, “A Rational Design Process: How and Why to Fake It”, in IEEE Transactions on Software Engineering, vol. SE-12, No. 2, pp. 251-257, February 1986 [reprinted in Software State-of-the-art: Selected papers, edited by Tom DeMarco and Timothy Lister, Dorset House Publishing, pp. 346-357, 1990]
- [Paulk 1999] Mark C. Paulk, Using the Software CMM® With Good Judgment, in ASQ Software Quality Professional, Vol. 1, No. 3, pp. 19-29, June 1999.
- [Paulk et al. 1994] Mark C. Paulk, Charles V. Weber, Bill Curtis and Mary Beth Chrissis, The Capability Maturity Model - Guidelines for Improving the Software Process, CMU-SEI, Addison-Wesley, 441 pages, 1994.
- [Phillips 2005] Mike Phillips, CMMI® Version 1.2 and Beyond, slides from presentation, Software Engineering Institute, Carnegie Mellon University, December 13, 2005.
- [PMI 2000] Project Management Institute PMI, A Guide to the Project Management Body of Knowledge (PMBOK Guide) 2000 Edition, PMI, 2000.
- [PMI 2003] Project Management Institute, Organizational Project Management Maturity Model Handbook (OPM3), December 2003.
- [PMI 2004] Project Management Institute PMI, A Guide to the Project Management Body of Knowledge, PMBOK Guide, Third Edition, PMI, 2004.
- [Pollice 2004] Gary Pollice, How far have we come?, in The Rational Edge, 75 pages, December 2004.
- [Pollice 2005] Gary Pollice, Teaching software development vs. software engineering, in The Rational Edge, 5 pages, December 2005.
- [Potter e Sakry 2002] Neil S. Potter and Mary E. Sakry, Making Process Improvement Work: A Concise Action Guide for Software Managers and Practitioners, Addison-Wesley Professional, ISBN 0201775778, 2002.
- [Potts 1993] Colin Potts, "Software-Engineering Research Revised", IEEE Software, Volume 10, Number 5, pages 19-28, September 1993.
- [Radice e Phillips 1988] Radice, Ronald A. and Phillips, Richard W. Software Engineering, An Industrial Approach. Englewood Cliffs, New Jersey: Prentice Hall, 1988.
- [Reifer 2003] Donald J. Reifer, Is the Software Engineering State of the Practice Getting Closer to the State of the Art?, p. 78-83, IEEE Software, November/December 2003.
- [Reo 2000] David Reo, Linking Business with Improvement – The Balanced IT Scorecard, slides from tutorial presented at SIMPROS 2000 [Salviano e Santana 2000].
- [Richmond e Udell 2004] Gary Richmond and Benjamin A. Udell, Trikonc, slides of presentation at PORT (Peirce On-line Resouce Testbed) Workshop, 2004

- [Rifkin 2002] Stan Rifkin, Is process improvement irrelevant to produce new era software? in Software Quality - ECSQ 2002, Lecture Notes in Computer Science 2349, ed. by Jyrki Kontio and Reidar Conradi, pp. 13-16, Springer-Verlag, 2002.
- [Rifkin 2003] Stan Rifkin, Why new software processes are not adopted, submitted to Advances in Computers, vol. 59, 2003.
- [Rocha et al. 2001] Ana Regina C. da Rocha, José Carlos Maldonado e Kival C. Weber (Editores), Qualidade de Software: Teoria e Prática, Prentice Hall, 303 páginas, 2001.
- [Rouiller 2001] Ana Cristina Rouiller, Gerenciamento de Projetos de Software para Empresas de Pequeno Porte, Tese de doutorado em Ciência da Computação pela UFPE, 2001.
- [Rouiller 2005] Ana Cristina Rouiller (Editora chefe), ProQualiti – Qualidade na Produção de Software, Núcleo de Estudos Avançados em Engenharia e Qualidade de Software, Universidade federal de Lavras – UFLA, Volume 1, Número 1, Maio de 2005.
- [Rout 2000] Terry Rout, Evolving SPICE – the Future for ISO 15504, in Proceedings of SPICE 2000 First International Conference on Software Process Improvement and Capability Determination, Limerick, Ireland, pages 1-8, June 2000.
- [Rout e Daoud 2003] Terry Rout and Lucas Daoud, Defining Target Process Capability Profile for Components Providers, in Proceedings of SPICE 2003: The Joint ESA - Third International SPICE Conference on Process Assessment and Improvement, ESTEC, Noordwijk The Netherlands, 17-21 March 2003, p. 121-126.
- [Rout e Dorling 2000] Terry Rout and Alec Dorling (editors), Proceedings of SPICE 2000: The First International Conference on Software Process Improvement and Capability Determination, 259 pages, Limerick, Ireland, June 2000.
- [Rout e Dorling 2002] Terry Rout and Alec Dorling (editors), Proceedings of SPICE 2002: The Second International Conference on Software Process Improvement and Capability Determination, Venice, Italy, March 2002.
- [Rout e Dorling 2003] Terry Rout and Alec Dorling (editors), Proceedings of SPICE2003: The Joint ESA - Third International SPICE Conference on Process Assessment and Improvement, ESTEC, Noordwijk The Netherlands, 17-21 March 2003.
- [Rout e Dorling 2004] Terry Rout and Alec Dorling (editors), Proceedings of SPICE 2004 The Fourth International SPICE Conference, Lisbon, Portugal, 184 pages, April 28-29, 2004.
- [Rout et al. 2000] Terry Rout, Angela Tuffley, Brent Cahill and B. Hodgen, The Rapid Assessment of Software Process Capability, Proceedings of SPICE 2000 The First International SPICE Conference, June 10-11, 2000, Limerick, Ireland, p. 47-55.
- [Rout et al. 2001] Terry Rout, Angela Tuffley and Brent Cahill, CMMI Evaluation - Capability Maturity Model Integration mapping to ISO/IEC 15504-2:1998, Technical Report, Software Quality Institute, Griffith University, prepared for the Defence Material Organisation, Purchase Order and Contract No 3726732, version 1.0, 14 pages, November 2001 (disponível em <http://www.sqi.gu.edu.au/cmmi/indexFrameset.html>, acessado mais recentemente em 11/05/2004)
- [Salviano 1998b] Clênio F. Salviano, “Searching for Software Process Introductory Patterns”, in Proceedings of Symposium on Software Technology SoST’98, Buenos Aires, Argentina, September 1st, 1998, pp. 87-93.

- 
- [Salviano 2000] Clênio F. Salviano, “Introducing Software Patterns to SPICE community”, Proceedings of SPICE 2000 The First International SPICE Conference, Limerick, Ireland, p. 185-194, June 10-11, 2000.
- [Salviano 2001a] Clênio F. Salviano, Um Método para Escolha dos Processos para uma Melhoria Alinhada aos Objetivos de Negócio, Anais do WQS'2001 Workshop de Qualidade de Software do SBES'2001, p. 39-50, Rio de Janeiro, RJ, Outubro 2001.
- [Salviano 2001b] Clênio F. Salviano, Futura Norma ISO/IEC 15504 (SPICE), Seção 1.1.4 de Qualidade e Produtividade em Software – 4ª Edição Renovada, K. C. Weber, A R. C. da Rocha e C. J. do Nascimento (editores), Makron Books, p. 19-25, 2001.
- [Salviano 2003] Clênio F. Salviano, Melhoria e Avaliação de Processo com a ISO/IEC 15504(SPICE) e CMMI, publicação do curso de pós-graduação “Lato Sensu” / (Especialização) a distância em melhoria de processo de software, UFLA/FAEPE, 2003.
- [Salviano 2005] Clênio F. Salviano, “Uma Análise Semiótica de PRO2PI: Expondo elementos de uma abordagem para melhoria de processo de software à semiótica triádica e às três categorias universais de Peirce”, Trabalho da disciplina IA 005 Semiótica e Sistemas Inteligentes, DCA FEEC UNICAMP, ministrada pelo Prof. Dr. Ricardo R. Gudwin, 6 páginas, junho de 2005.
- [Salviano e Jino 2004] Clênio F. Salviano and Mario Jino, Using Continuous Modes as “Dynamic and Specific Staged Models”, slides from presentation at Fourth Annual CMMI Technology Conference and Users Group, Denver, USA, November 2004. (disponível em [http://www.dtic.mil/ndia/2004/CMMIT1Tue/1114ClenioSalviano\\_new.pdf](http://www.dtic.mil/ndia/2004/CMMIT1Tue/1114ClenioSalviano_new.pdf), acessado mais recentemente em 12/10/2005)
- [Salviano e Santana 1999] Clênio F. Salviano e Ozeas V. Santana Filho (Editores), Anais e Slides das Apresentações do SIMPROS 1999: Primeiro Simpósio Internacional de Melhoria de Processo de Software, São Paulo, 1999.
- [Salviano e Santana 2000] Clênio F. Salviano e Ozeas V. Santana Filho (Editores), Anais e Slides das Apresentações do SIMPROS 2000: Segundo Simpósio Internacional de Melhoria de Processo de Software, São Paulo, 442 páginas, 2000.
- [Salviano e Santana 2001] Clênio F. Salviano e Ozeas V. Santana Filho (Editores), Anais e Slides das Apresentações do SIMPROS 2001: Terceiro Simpósio Internacional de Melhoria de Processo de Software, São Paulo, 2001.
- [Salviano e Santana 2003] Clênio F. Salviano e Ozeas V. Santana Filho (Editores), Anais e Slides das Apresentações do SIMPROS 2003: Quinto Simpósio Internacional de Melhoria de Processo de Software, ISBN 85-7359-326-1, Recife, PE, 3-5/11/2003.
- [Salviano e Santana 2004] Clênio F. Salviano e Ozeas V. Santana Filho (Editores), Anais do SIMPROS 2004: Sexto Simpósio Internacional de Melhoria de Processo de Software, São Paulo, SP, 2004.
- [Salviano e Souza 2000] Clênio F. Salviano e Eduardo P. de Souza, "SPICE Trials and Dissemination in Brazil: 1996-1999”, Proceedings of SPICE 2000: The First International SPICE Conference, Limerick, Ireland, June 10-11, 2000, p. 245-254.
- [Salviano e Tsukumo 2005] Clênio F. Salviano e Alfredo N. Tsukumo, Coop-MPS: Um Método para Projetos Cooperativos de Melhoria de Processo de Software e sua aplicação com o Modelo MPS.MPS, ProQualiti – Qualidade na Produção de Software, Núcleo de Estudos Avançados em

- Engenharia e Qualidade de Software, UFLA, Volume 1, Número 2, Novembro 2005, ISSN 1807-5061.
- [Salviano et al. 1992] Clênio F. Salviano, Wagner R. De Martino, Pedro Correa, Adriano Carvalho and Fuad Gattaz Sobrinho, "Signo: The SPP Object Management System", in PD-Vol. 43, Computer Applications and Design Abstraction ASME, pages 141-149, Houston, Texas, 1992.
- [Salviano et al. 1999] Clênio F. Salviano, Eduardo P. de Souza, Ana C. B. Dominoni, Adilson S. Nicoletti, "Experiência de Avaliação de Processos e Planejamento da Melhoria Utilizando a Futura Norma ISO/IEC 15504 (SPICE)", Anais do WQS'99 Workshop de Qualidade de Software do SBES'99 Simpósio Brasileiro de Engenharia de Software, p. 1-17, Florianópolis, SC, Outubro 1999.
- [Salviano et al. 2001] Clênio F. Salviano, Maria Alexandra V. C. da Cunha, Mario Lúcio Côrtes, e William L. de Oliveira, SPICE, capítulo do livro Qualidade de Software: Teoria e Prática, Ana Regina C. da Rocha, José Carlos Maldonado e Kival C. Weber (Editores), Prentice Hall, págs. 29-34, 2001.
- [Salviano et al. 2001] Clênio F. Salviano, Mary Santana e Eduardo Souza, Orientações para o estabelecimento da Diretoria de Contratos e Documentação, Relatório de Projeto, CenPRA (na época Instituto Nacional de Tecnologia da Informação ITI), 2001.
- [Salviano et al. 2002] Clênio F. Salviano, Ozeas V. Santana Filho e Teresa M. de M. Maciel (Editores), Anais e Slides das Apresentações do SIMPROS 2002: Quarto Simpósio Internacional de Melhoria de Processo de Software, Recife, PE, 2002.
- [Salviano et al. 2004] Clênio F. Salviano, Mario Jino and Manuel de Jesus Mendes, Towards an ISO/IEC 15504-Based Process Capability Profile Methodology for Process Improvement (PRO2PI), in Proceedings of SPICE 2004 The Fourth International SPICE Conference, Lisbon, Portugal, p. 77-84, April 28-29, 2004.
- [Salviano et al. 2005] Clênio F. Salviano, Ozeas V. Santana Filho e Isabelle R. Lund (Editores), Anais do SIMPROS 2005: Sétimo Simpósio Internacional de Melhoria de Processo de Software (publicado em CD, disponível em [www.simpros.com.br](http://www.simpros.com.br)), São Paulo, SP, 21-23/11/2005, 2005.
- [Scacchi 1994] Walt Scacchi, Process Models in Software Engineering, topic in Encyclopedia of Software Engineering, John J. Marciniak (editor), John Wiley & Sons, pp. 860-869, 1994.
- [Scheible 2004] Alba Scheible, Comprometimento no trabalho: um estudo de caso de suas relações com desempenho e práticas de gestão, Dissertação de mestrado, Universidade Federal da Bahia, 2004.
- [Scheible e Bastos 2005] Alba Scheible e Antônio Virgílio Bastos, CMM e Comprometimento: Um estudo de caso na implantação do nível 2, Anais do IV Simpósio Brasileiro de Qualidade de Software, SBQS, p. 255-270, Porto Alegre, RS, 6 a 10 de junho de 2005.
- [searchCRM 2006] searchCRM, The searchCRM.com website, 2006. (disponível em <http://searchcrm.techtarget.com>, acessado mais recentemente em 30/03/2006)
- [SEI 1993] Software Engineering Institute, TR-24 e TR-25.
- [SEI 2000] Software Engineering Institute, An Overview of Capability Maturity Model Integration (CMMI) - Version 1.0, Slides from tutorial presented at SIMPROS 2000 [Salviano e Santana 2000].

- 
- [SEI 2001a] Software Engineering Institute, Standard CMMI<sup>SM</sup> Appraisal Method for Process Improvement (SCAMPI<sup>SM</sup>), Version 1.1: Method Definition Document, CMU/SEI-2001-HB-001, 2001.
- [SEI 2001b] Software Engineering Institute, Concept of Operations for the CMMI®, January 15, 2001, in <http://www.sei.cmu.edu/cmmi/background/conops.html>, acessado mais recentemente em 11/05/2004.
- [SEI 2001c] Software Engineering Institute, Proceedings of SEPG 2001: Focusing on the Delta, The 13th Software Engineering Process Group Conference, New Orleans, USA, March 12–15, 2001.
- [SEI 2002a] Software Engineering Institute, Capability Maturity Model® Integration (CMMI<sup>SM</sup>), Version 1.1, CMMI for Systems Engineering and Software Engineering, (CMMI-SE/SW, V1.1) Continuous Representation, Technical Report CMU/SEI-2002-TR-011 and ESC-TR-2002-011, 724 pages, March 2002.
- [SEI 2002b] Software Engineering Institute, Capability Maturity Model ® Integration (CMMI<sup>SM</sup>), Version 1.1, CMMI for Systems Engineering and Software Engineering (CMMI-SE/SW, V1.1) Staged Representation, Technical Report CMU/SEI-2002-TR-012 and ESC-TR-2002-012, 729 pages, March 2002.
- [SEI 2004a] Software Engineering Institute, Process Maturity Profile - Software CMM® 2003 Year End Update, slides for presentation, 31 slides, March 2004. (disponível em <http://www.sei.cmu.edu/sema/pdf/SW-CMM/2004marSwCMM.pdf>, acessado mais recentemente em 07/06/2004)
- [SEI 2004b] Software Engineering Institute, Process Maturity Profile - CMMI® v1.1 - SCAMPI<sup>SM</sup> v1.1 Appraisal Results - 2003 Year End Update, slides for presentation, 21 slides, March 2004. (disponível em <http://www.sei.cmu.edu/sema/pdf/CMMI/2004marCMMI.pdf>, acessado mais recentemente em 07/06/2004)
- [SEI 2005] Software Engineering Institute, Web Site do Software Engineering Institute – SEI, <http://www.sei.cmu.edu/>. (CMMI models disponível em [www.sei.cmu.edu/cmmi](http://www.sei.cmu.edu/cmmi))
- [Selic 2003] Bran Selic, The Pragmatics of Model-Driven Development, IEEE Software, p. 19-25, September/October 2003.
- [Shaeffer 2004] Mark Shaeffer, DoD System Engineering and CMMI, slides from keynote speaker presentation at Fourth Annual CMMI Technology Conference and Users Group, Denver, USA, November 2004. (disponível a partir de <http://www.dtic.mil/ndia/2004>, acessado mais recentemente em 12/10/2005)
- [Shaw 1990] Mary Shaw, Prospects for an Engineering Discipline of Software, IEEE Software, November 1990.
- [Shaw 2001] Mary Shaw, The Coming-of-Age of Software Engineering Research?, in Proceedings of the 23rd International Conference on Software Engineering (ICSE), Toronto, Ontario, Canada, pp. 656-664, May 12 - 19, 2001 (disponível em <http://www.cs.cmu.edu/~shaw/>, acessado mais recentemente em May 7, 2004)
- [Shaw 2002] Mary Shaw, What Makes Good Research in Software Engineering?, presented as an invited lecture at the European Joint Conference on Theory and Practice of Software, in April 2002 in Grenoble, France, in International Journal on Software Tools for Technology Transfer,

- vol. 4, no. 1, Oct. 2002, pp. 1-7. (disponível em <http://www.cs.cmu.edu/~shaw/>, acessado mais recentemente em May 7, 2004)
- [Sheakley 2002] Clark A. Sheakley, Ahoy Mate! The PMBOK® Guide Distilled for the SPI Professional, slides from presentation at DC SPIN Chapter Meeting, October 2, 2002.
- [Sheard 2001a] Sarah A. Sheard, Evolution of the Frameworks Quagmire, IEEE Computer, July 2001, pp. 96-98.
- [Sheard 2001b] Sarah A. Sheard, Help! How Do I Make My Organization Comply With Yet Another New Model?, in Proceedings of the International Council on Systems Engineering, 2001 (disponível em <http://www.software.org/pub/externalpapers/>, acessado mais recentemente em 29/12/2003)
- [Sheard e Roedler 1999] Sarah A. Sheard and Garry J. Roedler, Interpreting Continuous-View Capability Models for Higher Levels of Maturity, in The Journal of the International Council on Systems Engineering, Volume 2, No. 1, John Wiley & Sons, 1999. (preprint available in <http://www.software.org/pub/externalpapers/contarch/>, acessado mais recentemente em 29/12/2003)
- [Sheard e Schoening 2002] Sarah A. Sheard and William W. Schoening, Systems Engineering Beyond Capability Models, in Proceedings of the International Council on Systems Engineering, August 2002. (disponível em <http://www.software.org/pub/externalpapers/>, acessado mais recentemente em 29/12/2003)
- [Sherer e Thrasher 2005] S. Wayne Sherer and Sandy Thrasher, CMMI and PMBOK Mappings, 2005.
- [Silva et al. 2003] Odair J. da Silva, Carlos A. Borges, Clênio F. Salviano, Ana L. Sampaio, Adalberto N. Crespo, e Ana Cristina Rouiller, An ISO/IEC 15504-Based Software Process Improvement Project in a Small Brazilian Software Organization, in Proceedings of SPICE 2003 The Third International SPICE Conference, 2003.
- [Siqueira et al. 2003] André Henrique de Siqueira, Rodrigo E. de Castro e Zalkind L. D. Rocha, “Processo de Melhoria de Software utilizando conceitos do PMBOK, SWEBOK e trabalho em times”, 12 slides, Anais do SIMPROS 2003 [Salviano e Santana 2003, p. 287-288].
- [SPC 2003] Software Productivity Consortium, The SPC Quagmap® Tool, [www.software.org](http://www.software.org). 2003.
- [Sperle e Bauer 2002] Marion Sperle and Christoph Bauer, Aligning Business and Customer Needs for Internal Process Improvement, Siemens AG, CT SE 3, MS, CBa 11.2002, 36 slides, 2002.
- [Spinola 2003] Mauro de Mesquita Spinola, ISO 9000 para Software, publicação do curso de pós-graduação “Lato Sensu” / (Especialização) a distância em melhoria de processo de software, UFLA/FAEPE, 2003.
- [Stallinger 2000] Friedrich Stallinger, Software process Simulation to Support ISO/IEC 15504 Based Software Process Improvement, in Software Process Improvement and Practice, volume 5, No. 2-3, ISSN 1077-4866, pp. 197-209, June/September 2000.
- [Stallinger et al. 2002] F. Stallinger, A. Dorling, T. Rout, B. Henderson-Sellers, B. Lefever Software Process Improvement for Component-Based Software Engineering: An Introduction to the OOSPICE Project, Proceedings of the 28<sup>th</sup> EUROMICRO Conference, September 4-6, 2002, Dortmund, Germany, IEEE Computer Society, Los Alamos, CA, 2002

- 
- [Tomi 2005] Gerson Tomi, Utilização da ISO/IEC 15504 (SPICE) para Melhoria de Processo em uma Empresa de Software, Monografia de conclusão do curso de Pós-Graduação Latu Senso em Melhoria de Processo de Software da Universidade Federal de Lavras UFLA, 2005.
- [Travassos et al. 2002] Guilherme H. Travassos, Dmytro Gurov e Edgar A. G. do Amaral, Introdução à Engenharia de Software Experimental, Relatório Técnico RT-ES-590/02, Programa de Engenharia de Sistemas e Computação COPPE / UFRJ, Rio de Janeiro, 2002.
- [van Loon 2004a] Han van Loon, Improvement and Process Assessment, in Proceedings of SPICE 2004 The Fourth International SPICE Conference, Lisbon, Portugal, p. 45-53, April 28-29, 2004.
- [van Loon 2004b] Han van Loon, Process Assessment and ISO/IEC 15504, The Kluwer International Series in Engineering and Computer Science, Volume 775, 281 pages, Springer, 2004.
- [Völcker et al. 2002] Christian Völcker, Alec Dorling, Patrizia Secchi, Paolo Zilioli and Ann Cass, A Process-Based Assessment Model for Space Quality Management Systems, in Proceedings of SPICE 2002 The Second International SPICE Conference, Venice, Italy, 2002.
- [von Wangenheim e Salviano 2005] Christiane G. von Wangenheim and Clênio F. Salviano, Consolidação de uma Metodologia para Avaliação de Processos de Software de MPEs Baseada na Norma ISO/IEC 15504 (SPICE) (Projeto PBQP 2.32 2004), artigo sobre projeto premiado no PBQP ciclo 2004 e publicado nos Anais do IV Simpósio Brasileiro de Qualidade de Software, SBQS, p. 21-26, Porto Alegre, RS, 6 a 10 de junho de 2005.
- [von Wangenheim et al. 2005a] Christiane G. von Wangenheim, Timo Varkoi and Clênio F. Salviano, Performing ISO/IEC 15504 Conformant Software Process Assessments in Small Software Companies, experience report at EUROSPI, Budapest, Hungria, 2005.
- [von Wangenheim et al. 2005b] Christiane G. von Wangenheim, Kênia K. Pickler, Marcello Thiry, Alessandra C. Zoucas, e Clênio F. Salviano, Aplicando avaliações de contextualização em processos de software alinhados ao CMMI-SE/SW”, em Anais do SIMPROS 2005 [Salviano et al. 2005, 13 páginas].
- [von Wangenheim et al. 2006a] Christiane G. von Wangenheim, Timo Varkoi and Clênio F. Salviano, Process Assessments in Small Software Companies, to be published at SPIP Software Process Improvement and Practice, 2006.
- [von Wangenheim et al. 2006b] Christiane G. von Wangenheim, Alessandra Anacleto, Clênio F. Salviano. "Helping Small Companies Assess Software Processes," IEEE Software, vol. 23, no. 1, pp. 91-98, January/February, 2006.
- [Vu 2001] John D. Vu, The e-Commerce Capability Development Model: A Framework to Implement e-Commerce Successfully, slides from a tutorial presented at The SEPG Conference, 81 slides, March 12-15, 2001.
- [Weber et al. 2005a] Kival C. Weber, Eratóstenes Araújo, Cristina A. F. Machado, Danilo Scalet, Clênio F. Salviano, e Ana Regina C. da Rocha, Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software - versão 1.0 (MR-MPS e MA-MPS), Anais do IV Simpósio Brasileiro de Qualidade de Software - SBQS, p. 347-360, Porto Alegre, RS, 6 a 10 de junho de 2005
- [Weber et al. 2005b] Kival C. Weber, Eratóstenes E. R. Araújo, Ana Regina C. da Rocha, Cristina A. F. Machado, Danilo Scalet, Clênio F. Salviano, Brazilian Software Process Reference Model and Assessment Method, in Proceedings of The 20th International Symposium on Computer and

Information Sciences - ISCIS'05, October 26-28<sup>th</sup>, Instambul, Turkey, published as LNCS 3733, pp. 402-411, P. Youlum et al. (Eds.), Springer-Verlag Berlin Heidelberg, 2005.

[Weinberg 1993] Gerald M. Weinberg, Software com Qualidade: Pensando e Idealizando Sistema, Makron Books, 1993.

[Whyte 1996] David Whyte, The Heart Aroused, Poetry and the Preservation of the Soul in Corporate America, Doubleday, 1996.

[Wiegiers 1998] Karl E. Wiegiers, Read My Lips: No New Models!, IEEE Software, September/October 1998 (disponível em [www.processimpact.com](http://www.processimpact.com)).

[Wiegiers 1999] Karl E. Wiegiers, Writing Quality Requirements, Software Development Magazine, May 1999. (disponível em [www.processimpact.com](http://www.processimpact.com))

[Zahran 1998] Sami Zahran, Software Process Improvement – Practical Guidelines for Business Success, Addison-Wesley, 447 pages, 1998.

# Apêndice A.1

## A.1 Diagrama de Classe de PRO2PI-MODEL

Os elementos a seguir, com suas descrições e relacionamentos representam conceitualmente o metamodelo PRO2PI.

E exemplos em relação a iCMM, CMMI, 15504-5

### A1.1 Referência para prática

#### **Cl\_ReferencePractice:**

Cl\_PracticeElement é uma classe abstrata que representa uma generalização de todos os elementos do modelo, segundo a visão que todo modelo de capacidade de processo é composto por elementos que são práticas de referência. A classe Cl\_PracticeElement é especializada pelas subclasses Cl\_PracticeElement; Cl\_PracticesGroup e Cl\_PracticesSystem, que representam os três níveis de granularidade das práticas descritas no modelo.

Também é especializada em Cl\_ProcessAreasGroup, mas não tem a mesma relevância conceitual dos outros três níveis de granularidade e é mais uma detalhe de Cl\_PracticesGroup. As classes Cl\_PracticeGuidance e Cl\_SourceModel também poderiam ser modeladas como especializações de Cl\_PracticeElement, pois também são referências de boas práticas, mas para simplificar o desenho, e com isto melhorar o entendimento, elas não foram modeladas como especializações de Cl\_PracticeElement.

Um Cl\_PracticeElement é composto por uma identificação, um nome e uma descrição da prática (os atributos *id*, *name* e *description*) e opcionalmente, por um ou mais guias para um melhor entendimento da prática (Cl\_PracticeGuidance). Conceitualmente, todo elemento de um PRO2PI é uma referência para uma boa prática, e com isto pode ser modelado como uma especialização de Cl\_PracticeElement.

#### **Cl\_PracticeGuidance:**

A classe Cl\_PracticeGuidance representa um guia com informações adicionais sobre um elemento do modelo, para um melhor entendimento. Também pode conter referência para um ou mais modelos e

um indicação do tipo de relacionamento com esse elemento do modelo. Isto serve de base para manter uma rastreabilidade com modelos relevantes.

**CI\_SourceModel:**

A classe CI\_SourceModel representa a identificação e informações sobre um modelo, do qual, um CI\_PracticeElement, por meio de um CI\_PracticeGuidance, tem um relacionamento. Isto é utilizado para manter a rastreabilidade com modelos relevantes. Os principais modelos que devem ser utilizados são aqueles relacionados na Figura 81.

## A1.2 Elemento de referência para prática

**CI\_PracticeElement:**

CI\_PracticeElement é uma classe abstrata e representa uma generalização de práticas de referência para a melhoria. É a menor unidade do modelo. Uma CI\_PracticeElement pode ser especializada em uma das seguintes cinco classes: CI\_Objective, CI\_Outcome, CI\_BasePractice, CI\_Artefact ou CI\_Resource. Note que o termo CI\_PracticeElement é usado com liberdade, pois metas e produtos de trabalho são também considerados como “práticas”. A idéia é consolidar esse nível de granularidade com os cinco classes de elementos que representam referências básicas para a melhoria baseada em processo.

**CI\_Objective:**

A classe CI\_Objective é uma especialização de CI\_PracticeElement e representa um objetivo que deve ser atingido por processos em uma Unidade Organizacional. Para orientar a construção de práticas para o atendimento de um objetivo, outros elementos, também especializações de CI\_PracticeElements, são definidos e relacionados ao objetivo: CI\_Outcome, CI\_BasePractice, CI\_Resource e CI\_Artefact.

Uma CI\_Objective inclui a definição de um conjunto de CI\_Outcomes e um conjunto de CI\_BasePractices. No CMMI-SE/SW CI\_Objective corresponde a uma generalização de Objetivo Específico (*Specific Goal*) e Objetivo Genérico (*Generic Goal*). Na ISO/IEC 15504-5, não existe o conceito de objetivo, mas uma CI\_Objective pode ser sempre definido a partir do conceito de Propósito (Purpose) de um processo, atributo de processo (Process Attribute) ou de um nível de capacidade de processo.

**CI\_Outcome:**

A classe CI\_Outcome é também uma especialização de CI\_PracticeElement e representa um resultado que pode ser observado para evidenciar, junto com outros resultados, o atendimento de um objetivo

(CI\_Objective). Esse resultado descreve “produção de um artefato, uma mudança significativa de estado ou o atendimento de elementos especificados, como, por exemplo, requisitos e objetivos” [ISO/IEC 15504-5 2006]. Um conjunto de CI\_Outcomes é associado a um CI\_Objective.

Na ISO/IEC 15504-5 CI\_Outcome é Resultado (*Outcome*) com exatamente o mesmo significado. No CMMI-SE/SW não existe o conceito de Resultado, mas um CI\_Outcome pode ser definido a partir dos objetivos específicos e genéricos.

### **CI\_BasePractice**

A classe CI\_BasePractice é também uma especialização de CI\_PracticeElement e representa uma atividade que pode ser utilizada como referência para a construção de práticas. Um conjunto de CI\_BasePractices está associado a um CI\_Outcome.

No CMMI, CI\_BasePractice corresponde a uma generalização de Prática Específica e Prática Genérica. Na 15504, CI\_BasePractice corresponde a uma generalização de prática básica (para processo) e prática genérica (para nível de capacidade). Na 15504, uma prática básica pode atender a um ou mais resultados. Nesse caso a prática é dividida de tal forma a cada uma atender a apenas um resultado.

### **CI\_Artifact**

Um CI\_Artifact é também uma especialização de CI\_PracticeElement e representa algo produzido, consumido ou modificado por um processo.

Um conjunto de CI\_Artifacts está relacionado a um CI\_Outcome, podendo um mesmo CI\_Artifact estar relacionado a mais de um CI\_Outcome, até mesmo de diferentes CI\_PracticesGroups.

Na 15504 e no CMMI, CI\_Artifact é produto de trabalho (*work product*) com exatamente o mesmo significado.

### **CI\_Resource**

Um CI\_Resource é também uma especialização de CI\_PracticeElement e representa um recurso que pode ser utilizado. Um conjunto de CI\_Resource está relacionado a um CI\_Objective.

O CMMI não tem o elemento recurso. A 15504 tem o conceito de recurso apenas para os níveis de capacidade e não para processos.

## **A1.3 Grupo de elementos de referência para prática**

### **CI\_PracticesGroup:**

CI\_PracticesGroup é uma classe abstrata e representa uma generalização de um conjunto de práticas de referência. É a segunda granularidade de prática do modelo. Um CI\_PracticesGroup pode ser especializada na classe CI\_ProcessArea ou na classe CI\_CapabilityLevel. Um CI\_PracticesGroup inclui um conjunto de CI\_Objectives, que referencia os outros elementos: CI\_Outcomes, CI\_BasePractices, CI\_Artefacts e CI\_Resources.

**CI\_ProcessArea:**

A classe CI\_ProcessArea é uma especialização de CI\_PracticesGroup que representa atividades de desempenho específica de um determinado processo. As CI\_ProcessAreas estão organizados (sem hierarquia) em CI\_ProcessAreasGroups. Um conjunto de CI\_ProcessAreasGroup pode ser também organizado como um novo CI\_ProcessAreasGroup.

**CI\_ProcessAreasGroup:**

A classe CI\_ProcessAreasGroup representa agrupamento de áreas de processo ou agrupamento de outros grupos de processo. Por exemplo, no CMMI-SE/SW existem as quatro categorias de processo, que agrupam as áreas de processo e na ISO/IEC 15504-5 existem nove grupos de processos, que agrupam processos, e existem também três categorias de processo que agrupam os nove grupos de processos.

**CI\_CapabilityLevel:**

A classe CI\_CapabilityLevel é uma especialização de CI\_PracticesGroup que representa uma capacidade genérica que pode ser aplicada a qualquer CI\_ProcessArea. CI\_CapabilityLevels estão organizados de forma hierárquica (seqüencial e cumulativa).

## **A1.4 Sistema de grupo de elementos de referência para prática**

**CI\_PracticesSystem**

A classe abstrata CI\_PracticesSystem representa os três níveis de perfis de capacidade de processo. Associados a um CI\_PracticesSystem existem elementos sobre a racionalidade. Esses elementos incluem:

- Explicações sobre o elemento
- O quê estará sendo melhorado, e o quê não estará sendo
- Orientações sobre como medir se o quê deveria ter sido melhorado foi realmente melhorado e em quanto.

**CI\_ProcessAreaCapabilityProfile**

A classe `CI_ProcessAreaCapabilityProfile` representa basicamente uma associação de uma `CI_ProcessArea` com um `CI_CapabilityLevel`, acrescida de informações adicionais. Um `CI_ProcessAreaCapabilityProfile` é a terceira unidade do modelo.

Em um `CI_ProcessAreaCapabilityProfile` a associação de uma `CI_ProcessArea` a um `CI_CapabilityLevel` significa que a área de processo tem o comportamento da aplicação do nível de capacidade associado mais todos os níveis inferiores até o nível de capacidade 1.

### **CI\_ProcessCapabilityProfile**

A classe `CI_ProcessCapabilityProfile` representa um conjunto de `CI_ProcessAreaCapabilityProfiles`. Normalmente não existe repetição de `CI_ProcessArea` em um `CI_ProcessCapabilityProfile`. Porém é possível ter esta repetição.

### **CI\_ProcessCapabilityProfilesHierarchy**

A classe `CI_ProcessCapabilityProfilesHierarchy` representa um conjunto de `CI_ProcessCapabilityProfiles` que formam uma lista cumulativa e hierárquica de `CI_ProcessCapabilityProfiles`. Com exceção do primeiro e último `CI_ProcessCapabilityProfile` dessa lista, cada `CI_ProcessCapabilityProfile` PCPa contém um outro `CI_ProcessCapabilityProfile` PCPa-1 e está contido em um outro `CI_ProcessCapabilityProfile` PCPa+1. O primeiro `CI_ProcessCapabilityProfile` PCPa1 não contém nenhum `CI_ProcessCapabilityProfile` da lista e está contido no segundo `CI_ProcessCapabilityProfile` PCPa2. O último `CI_ProcessCapabilityProfile` PCPn não está contido em nenhum `CI_ProcessCapabilityProfile` da lista e contém o penúltimo `CI_ProcessCapabilityProfile` PCPn-1.

## Apêndice A.2

### A.2 Medição de complexidade de uma melhoria

A complexidade de um PRO2PI pode ser medida como uma medida derivada em termos de U-PIC<sup>13</sup>. Existem duas variações da medida de complexidade: a complexidade simplificada e a complexidade detalhada.

A medida derivada DM-PIC é definida a seguir por meio de dez regras.

Regra DM-PIC-01: As áreas de processo (AP) de um PRO2PI têm complexidades similares. Desta forma a complexidade de uma área de processo  $AP_{i1}$  pode ser considerada como igual à de qualquer outra área de processo  $AP_{i2}$ .

As áreas de processo dos modelos do CMMI e da ISO/IEC 15504 têm esta complexidade similar entre elas. Caso alguma área de processo seja criada, ela deve naturalmente manter esta mesma complexidade.

Regra DM-PIC-02: Incrementos unitários de níveis de capacidade de processo (NCP) em uma área de processo têm complexidade similar. Desta forma a complexidade de melhoria de uma área de processo do nível de capacidade  $NCP_i$  para o próximo nível  $NCP_{i+1}$  tem o mesmo valor qualquer que seja o nível de capacidade.

A medição de complexidade simplificada é calculada em função dos níveis de capacidade e é definida nesta seção por meio de um conjunto de regras.

Regra DM-PIC-03: Em um PRO2PI, o nível de capacidade atual  $NCP_{at}$  e o nível de capacidade alvo  $NCP_{al}$ , de cada área de processo  $AP_i$  do PRO2PI é conhecido.

Um PRO2PI inclui não apenas o perfil alvo, mas também o perfil atual em relação a esse perfil alvo.

---

<sup>13</sup> Esta Unidade de Melhoria de Processo foi baseada em uma proposta de um revisor anônimo da publicação [Salviano et. al. 2004]: “Defining a relative improvement metric so competition could be encouraged, e.g. Assign one Improvement Point for every incremental improvement of one Capability Level. Optionally, then divide total Improvement Points by the number of processes to determine the Average Improvement per process”.

Regra DM-PIC-04: A escala de valores da complexidade simplificada de cada área de processo  $AP_i$  com  $AP_i.[NCPal..NCPat]$  em um PRO2PI  $P_j$  é definido como um número inteiro, não negativo, no intervalo de 0 a 5.

O valor da complexidade simplificada de uma área de processo é um número inteiro entre 0 e 5.

Regra DM-PIC-05: O valor da complexidade simplificada de um  $AP_i.[NCPal..NCPat]$  é igual ao resultado de  $(NCPal - NCPat)$ , caso  $NCPal$  seja maior ou igual a  $NCPat$ , e igual a zero, caso contrário.

O valor da complexidade simplificada de uma área de processo está relacionado com a diferença entre o nível de capacidade alvo e o atual, e portanto é um valor inteiro entre 0 e 5. Caso, por algum motivo, o nível de capacidade alvo for menor que o nível atual, o valor de complexidade simplificada será zero.

Regra DM-PIC-06: A escala de valores da complexidade simplificada de um PRO2PI  $P_j$  é definida como um número inteiro, não negativo.

O valor da complexidade simplificada de um PRO2PI é um número inteiro, não negativo.

Regra DM-PIC-07: O valor da complexidade simplificada de um PRO2PI é igual à soma dos valores de complexidade básica de cada área de processo do PRO2PI.

O valor da complexidade simplificada de um PRO2PI é igual à soma da complexidade de cada área de processo, é um valor inteiro entre 0 e cinco vezes o número de áreas de processo.

A medição de complexidade detalhada é calculada em função das pontuações dos atributos de processo e é definida nesta seção por meio de um conjunto de axiomas.

Regra DM-PIC-08: Em um PRO2PI, os valores dos atributos de processo do nível de capacidade atual  $NCPa$  e dos atributos de processo do nível de capacidade desejado  $NCPd$ , de cada área de processo  $AP_i$  do PRO2PI são conhecidos.

Regra DM-PIC-09: Os atributos de processo (ATP) de cada nível de capacidade têm complexidades similares. Desta forma a complexidade de um atributo de processo de um nível de capacidade  $NCP_j ATP_{i1}$  é igual à de qualquer outro atributo de processo desse mesmo nível  $NCP_j ATP_{i2}$ .

Regra DM-PIC-10: Todos os atributos de processo de um nível de capacidade terão o mesmo peso, com soma total igual a 1. Por exemplo, para o nível 2, cada um dos dois atributos contribui com peso igual a 0,5.

Para cada atributo de processo, a pontuação indicará o percentual a ser considerado, seguindo a escala:

- Um incremento de N para P, significa 0,30,
- Um incremento de P para L, significa 0,40, e
- Um incremento de L para F, significa 0,30;

e naturalmente temos:

- Um incremento de N para L, significa 0,70,
- Um incremento de N para F, significa 1,00, e
- Um incremento de P para F, significa 0,70.

Esses valores foram definidos baseados na racionalidade descrita na Tabela 32.

Tabela 32 – Valores das pontuações de atributo de processo

<b>Pontuação</b>	<b>Intervalo de valores âncora</b>	<b>Valor médio</b>	<b>Incremento do anterior</b>	<b>Ajuste valor</b>	<b>Valor</b>
N	[ 0,00 – 0,15]	07,5	0	0	0
P	] 0,15 – 0,50]	32,0	0,25	+0,05	0,30
L	] 0,50 – 0,85]	67,5	0,35	+0,05	0,40
F	] 0,85 – 1,00]	92,5	0,25	+0,05	0,30

A coluna “Intervalo de valores âncora” indica o intervalo definido na ISO/IEC 15504-2 [2003]. A coluna “Valor médio” indica o valor médio entre os dois extremos do intervalo âncora. A coluna “Incremento do anterior” indica a diferença do valor médio da pontuação anterior com a atual, e representa o esforço para evoluir. Como a soma dos incrementos é igual a 0,85, os 0,15 restantes foram distribuídos igualmente como ajuste entre os incrementos. Desta forma a coluna “Ajuste do valor” indica esse incremento e a coluna “Valor” o valor relativo do esforço necessário para evolução de uma pontuação anterior para a pontuação corrente.

Dois fatores de redução podem ser utilizados para o valor de complexidade. Certas áreas de processo apoiam determinados níveis de capacidade. Desta forma caso um nível de capacidade seja planejado e áreas de processo que o apoiem estejam no PRO2PI, o valor de complexidade pode ser reduzido. A Tabela 33 mostra um exemplo de redução.

Tabela 33 – Exemplo de valores de redução para a complexidade de um PRO2PI

<b>Atributo</b>	<b>Processo (15504-5)</b>	<b>Atributo -&gt; Processo</b>	<b>Processo -&gt; Atributo</b>
2.1	MAN.3 Gerência de Projeto	1	1
2.2	SUP.7 Documentação	0	0,3
2.2	SUP.2 Verificação	0	0,3
2.2	SUP.8 Gerência de Configuração	0	0,4
3.1	PIM.1 Estabelecimento de Processo	0,5	0
3.2	PIM.1 Estabelecimento de Processo	0,5	0

A Tabela 34 e Tabela 35 mostram exemplos para PCP com uma área de processo.

Tabela 34 – Exemplo de complexidade para um PRO2PI

<b>Ger.Proj.</b>	<b>PA1.1</b>	<b>PA2.1</b>	<b>PA2.2</b>	<b>PA3.1</b>	<b>PA3.2</b>	<b>Nível</b>
Atual	L	L	P	P	N	1
Alvo	F	L	L	X	X	2
Unid.Melh.	0,3*1	0*0,5	0,4*0,5	-	-	0,5

Tabela 35 – Outro exemplo de complexidade para outro PRO2PI

<b>Ger.Proj.</b>	<b>PA1.1</b>	<b>PA2.1</b>	<b>PA2.2</b>	<b>PA3.1</b>	<b>PA3.2</b>	<b>Nível</b>
Atual	L	L	P	P	N	1
Alvo	F	F	F	L	L	3
Unid.Melh.	0,3*1	0,3*0,5	0,7*0,5	0,4*0,5	0,7*0,5	1,45

## Apêndice A.3

### A.3 Fases e atividades de PRO2PI-WORK

O objetivo deste capítulo é descrever um método para o estabelecimento de um PRO2PI que possa ser utilizado nas fases iniciais de PRO2PI-CYCLE e seja viável para micro e pequenas empresas.

A seguir o método PRO2PI-WORK é ilustrado (Figura 77) e descrito como um processo, na notação ETVX [Radice e Phillips 1988] em termos de propósito, critério para início, entradas, tarefas ou fases, critério para término, saídas e atores. Esta descrição é complementada com considerações sobre medições e relação dos artefatos produzidos ou utilizados no processo.

O propósito do processo PRO2PI-WORK é estabelecer um perfil de capacidade de processo como indutor de um ciclo de melhoria de processo (PRO2PI).

Esse propósito é desdobrado nos seguintes objetivos específicos:

- Capacitar membros da organização em fundamentos da engenharia de processo e de modelos relevantes de capacidade de processo,
- Consolidar informações relevantes sobre a organização,
- Consolidar objetivos estratégicos para a melhoria,
- Definir um PRO2PI,
- Entender os processos atuais em relação ao PRO2PI definido,
- Consolidar uma representação em alto nível dos processos atuais,
- Definir orientações para atingir PRO2PI, e
- Reforçar motivação para melhoria de processo.

Os atores (ou papéis) para esse processo são os seguintes:

- Especialistas: Equipe de consultores, internos e/ou externos à organização, especialistas em melhoria de processo, no processo PRO2PI-WORK e nos modelos relevantes de capacidade processo.
- Patrocinador: Representante da organização, com cargo de direção apropriado para garantir os recursos para a melhoria e demandar o atendimento dos objetivos estratégicos.
- Contato: Pessoa da empresa, indicada pelo patrocinador, para contato com os especialistas.

- Multiplicadores: Equipe de melhoria de processo da organização, participantes e treinandos no trabalho para serem multiplicadores, coordenadas por um líder.
- Dirigentes: Representantes dos dirigentes da organização.
- Gerentes: Representantes dos gerentes da organização.
- Técnicos: Representantes dos técnicos da organização.
- Outros: Outras pessoas.

Todas as equipes têm um coordenador designado. Todas as referências para os atores serão feitas no singular, indicando, no caso de equipe, tanto a equipe toda, ou apenas o coordenador, ou mesmo um subconjunto dos integrantes.

As entradas para esse processo são os seguintes:

- A01-AcordoTrab,
- A02-PlanoTrab, e
- A03-AcordoConf.

Os resultados desse processo são os seguintes:

- A04-RelatPRO2PI, e
- A05-RelatAvalTrab.

O critério de início para esse processo é o artefato A01-AcordoTrab assinado pelas partes, e o critério de conclusão é o artefato A04-RelatPRO2PI aceito pelo cliente.

As fases são início, definição, orientações, e conclusão. A Figura 78 relaciona o nome das fases e atividades do método.

A Figura 79 relaciona o tempo típico para realização de cada atividade e a soma das horas das atividades de cada fase, em uma micro empresa.

A medição principal utilizada é em relação às propriedades de um PRO2PI descrita como PRO2PI-PROP. A viabilidade de PRO2PI pode ser medida com o produto de informação Viabilidade de PRO2PI, descrita como PRO2PI-MEAS.

Vinte e cinco artefatos são produzidos e utilizados em PRO2PI-WORK. A Tabela 36 identifica, relaciona e descreve esses artefatos.

Tabela 36 - Artefatos de PRO2PI-WORK

<b>Id</b>	<b>Nome</b>	<b>Descrição resumida</b>
A06	QuestInfoUO	Questionário para informações sobre a Unidade Organizacional
A07	ApresEngProc	Slides para apresentação da engenharia de processo



6		A	E																	
7			A		E			E												
8			A					E		A			E		E					
9					A	A		E												
10			A									E								
11																			E	
12																			E	
13																			E	
14								S											E	
15																			E	
16																			E	
17			S						A	A				A		A			E	
18																			E	
19																			E	
20								A	E											
21								A	E											
22								A	E											
23																				
24								S												
25																				
26																				

O artefato A04-RelatPRO2PI é o principal resultado do processo e representa uma composição e consolidação de vários resultados intermediários. Esse artefato é iniciado com capa, controle, título, apresentação e resumo executivo; continuado com consolidações de R1DescUO, R2IntroEPS, R3ResTrab, R4DescSubsid, R5DescObjEstr, R6PCPAtual, R7DescPRO2PI, R8OrientMelh, e R9GerMelh; e finalizado com anexos, que podem incluir os subsídios levantados e outros.

As próximas seções descrevem as fases e atividades do método PRO2PI-WORK. Cada fase é descrita em termos do propósito, critérios de entrada, critérios de saída e relação das atividades. Cada atividade é descrita em termos de propósito, produtos de entrada, produtos de saída, tarefas e opcionalmente, considerações sobre a atividade e exemplos de artefatos.

## A3.1 PRO2PI-WORK Fase 1: Preparação do trabalho

**Propósito:** Obter e analisar informações relevantes sobre a unidade organizacional e preparar material para o trabalho.

**Critério de entrada:** Acordo de trabalho A01-AcordoTrab assinado.

**Critério de saída:** Material preparado.

Esta fase é realizada com as três atividades descritas a seguir.

### Atividade 1.1: Obter informações da Unidade Organizacional

**Propósito:** Obter informações relevantes sobre a unidade organizacional

**Critérios de entrada:**

- Um Patrocinador decidiu realizar PRO2PI-WORK na organização.
- Especialista foi designado.
- Um Contato da organização foi identificado pelo Patrocinador para coordenar a obtenção e envio das informações

**Produtos de entrada:** A01-AcordoTrab; A02-PlanoTrab; A03-AcordoConf e A06-QuestInfoUO[formulário].

**Produto de saída:** A06-QuestInfoUO [validado pelo *Contato* e pelo *Especialista*].

**Tarefas:**

- Especialista envia formulário A04-QuestInfoUO ao Contato
- Especialista instrui Contato sobre A04-QuestInfoUO
- Contato coordena preenchimento de A04-QuestInfoUO
- Contato valida A04-QuestInfoUO preenchido
- Contato envia A04-QuestInfoUO validado para Especialista

### Atividade 1.2: Analisar informações da Unidade Organizacional

**Propósito:** analisar informações relevantes sobre a organização e escolher e/ou definir conjunto relevante de áreas de processo, geralmente de 12 a 20 áreas de processo.

**Produtos de entrada:** A06-QuestInfoUO [validado]; A11-R1DescUO [template]; A15-R5DescObjEstr [template]; A07-ApresEPS [padrão]; A08-DescProcRel [padrão] e A17-R7DescPRO2PI [versão 0.1]

**Produtos de saída:** A11-R1DescUO [versão 0.1]; A15-R5DescObjEstr [versão 0.1] e A17-R7DescPRO2PI [versão 0.2].

**Tarefas:**

- Especialista analisa questionário
- Especialista produz descrição da unidade organizacional
- Especialista produz descrição dos objetivos estratégicos

**Atividade 1.3: Preparar próximas atividades**

**Propósito:** Preparar material para as atividades da fase F.2 a partir do conjunto de ativos do método, baseado no resultado da análise das informações relevantes sobre a organização, incluindo principalmente a escolha e adaptação do: (a) slides das apresentações, (b) material de cerca de 12 a 20 áreas de processos e (c) questionários. Identificar e agendar os participantes das atividades e reservar infra-estrutura.

**Produtos de entrada:** A01-AcordoTrab; A02-PlanoTrab e A03-AcordoConf

**Produtos de saída:** A07-ApresEngProc; A08-DescProcREI; eA09-QuestFOrg

**Tarefas:**

- Especialista escolhe de 12 a 20 áreas de processo dos modelos selecionados ou identifica novas áreas de processo
- Se identificou novas áreas de processo: Especialistas define material para os novos processos
- Especialista monta ProcRelev.
- Especialista monta ApresEPS.

## **A3.2 PRO2PI-WORK Fase 2: Escolha de PRO2PI**

**Propósito:** Escolher uma proposta inicial para o PRO2PI incluindo o PCP atual.

**Critério de entrada:** Material preparado

**Critério de saída:** PRO2PI escolhido

Esta fase é realizada com as sete atividades descritas a seguir.

**Atividade 2.1: Apresentar engenharia de processo e trabalho**

**Propósito:** Apresentar a todos os envolvidos a motivação, objetivos e características do trabalho a ser realizado na unidade organizacional, e fundamentos da engenharia de processo, do método PRO2PI-

WORK e de modelos de capacidade de processo mais relevantes. Abrir espaço para perguntas e respostas para esclarecer dúvidas. Obter ou reforçar clima positivo para o trabalho

**Produto de entrada:** A07-ApresEngProc

**Produtos de saída:** Registro com resumo das discussões e Lista de presença atualizada e preenchida.

**Tarefas:**

- Patrocinador abre a apresentação falando sobre motivação e objetivos
- Especialista apresenta os fundamentos
- Sessão de comentários, perguntas e esclarecimentos com todos os participantes

### **Atividade 2.2: Identificar fatores de negócio**

**Propósito:** Identificar com os diretores os principais objetivos estratégicos da organização e pontos fortes, fraquezas, oportunidades e ameaças.

Composta por entrevistas com a alta gerência e leitura de documentos para levantamento dos fatores de negócio, como contexto, objetivos, estratégia, pontos fortes, fraquezas, ameaças e oportunidades. Às vezes esses objetivos não estão documentados ou mesmo não estão claros na cabeça das pessoas. Nestas situações a equipe de trabalho deve auxiliar a definição e documentação desses objetivos.

**Produto de entrada:** A15-R5DescObjEstr []

**Produto de saída:** A15-R5DescObjEstr []

**Tarefas:**

- Patrocinador abre a reunião
- Especialista explica objetivos e forma de condução da reunião
- Especialista expõe versão corrente do A15-R5DescObjEstr
- Todos os participantes discutem os fatores de negócios
- Especialista consolida resultado em nova versão de A15-R5DescObjEstr
- Especialista expõe nova versão de A15-R5DescObjEstr

A Figura 84, da seção 7.2.3, ilustra um exemplo de objetivos estratégicos de uma organização.

### **Atividade 2.3: Obter fatores organizacionais**

**Propósito:** Obter opiniões sobre fatores organizacionais de pessoas representativas da organização. Respostas a Questionário com Opiniões sobre a Organização. Composta por sessões de respostas a um questionário com perguntas diversas sobre opiniões a respeito de vários aspectos da organização,

incluindo pontos fortes e fracos, experiências bem e mal sucedidas, sugestões para melhoria, ameaças e oportunidades.

Esta atividade pode ser omitida se a empresa é muito pequena ou se com a atividade 2.2 já tivermos um resultado satisfatório.

**Produtos de entrada:** A07-APresEPS e A09-QuestFOrg [formulário]

A Tabela 38 ilustra algumas das questões do questionário para opiniões sobre fatores organizacionais da organização.

Tabela 38 - Questões exemplo para fatores organizacionais

<p><b>3.4 Relate de um a três principais pontos fortes da unidade organizacional.</b> (Nota: Pontos fortes são condições do ambiente interno de uma organização, que apresentam situação atual <b>favorável</b>, em relação ao seu desempenho geral) a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.5 Relate de um a três principais fraquezas da organização.</b> (Nota: fraquezas são condições do ambiente interno de uma organização, que apresentam situação atual <b>desfavorável</b>, em relação ao seu desempenho geral) a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.6 Relate de uma a três das principais experiências na organização que foram bem sucedidas</b> (por exemplo, o que já foi feito na organização, deu certo e você faria de novo). a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.7 Relate de uma a três das principais experiências na organização que não foram bem sucedidas</b> (por exemplo, o que já foi feito na organização, não deu certo e você não faria de novo ou faria de outra maneira). a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.8 Relate de uma a três das principais ameaças à organização</b> (Nota: ameaças são variáveis do ambiente externo, de alta importância futura e <b>negativa</b> sobre as atividades e o desempenho da organização) a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.9 Relate de uma a três das principais oportunidades de negócio para a organização.</b> (Nota: oportunidades são variáveis do ambiente externo, de alta importância futura e <b>positiva</b> sobre as atividades e o desempenho da organização) a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.10 Relate de uma a três das ações para a melhoria da organização que você gostaria que fossem realizadas.</b> (podem ser ações novas, modificações de ações já existentes ou mesmo eliminação de alguma ação). a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i></p>
<p><b>3.11 Relate de uma a cinco das principais características de diferenciam esta organização das outras.</b> (por exemplo, aquilo que somente ela faz, aquilo que ela faz melhor do que as outras, aquilo que é o diferencial da organização,...) a) [ ] <i>max 320 cars.</i>; b) [ ] <i>max 320 cars.</i>; c) [ ] <i>max 320 cars.</i> d) [ ] <i>max 320 cars.</i>; e) [ ] <i>max 320 cars.</i></p>

**Produto de saída:** A09-QuestFOrg [preenchido]

**Tarefas:**

- Especialista apresenta objetivos do levantamento de fatores de negócio.
- Participantes respondem ao questionário.
- Especialista compila respostas do questionário.

**Atividade 2.4: Identificar relevância dos processos**

**Propósito:** Continuar capacitação sobre as áreas de processo relevantes à organização, enfatizando a descrição de cada área de processo como referência; os principais sintomas típicos quando a área não é bem executada e os principais benefícios que justificam porque a área deve ser bem executado. Para isto cada processo determinado sua realização atual e posicionado em uma matriz de importância do processo e risco de sua realização atual. Também identificar o processo de ciclo de vida relevante, em alto nível e mapear áreas de processo do modelo nesse processo.

**Produtos de entrada:** A07-ApresEngProc; A08-DescProcRel; A20-InfoRelevProcc [template]; A21-QuadroRelevProc [template] e A22-ProcUO [template]

**Produtos de saída:** A20-InfoRelevProcUO [preenchido para cada processo]; A21-QuadroRelevProcUO [preenchido com todos os processos] e A22-ProcUO.

**Tarefas:**

- Especialista orienta a elaboração de um A20-InfoRelevProcUO pelos multiplicadores, gerentes e técnicos do ciclo de vida de projeto da unidade organizacional, para cada área de processo selecionada.

Para cada área de processo, as seguintes tarefas são realizadas, com uma estimativa de 20 minutos por área de processo:

- Especialista apresenta o processo por: a) definição do processo, conforme o modelo; b) sintomas típicos quando não é bem executado, e c) benefícios típicos porque é importante.
- Discutir e identificar a correspondência dele na organização (Nota: nesse momento é possível a subdivisão do processo, nesses casos os passos seguintes são executados para cada subdivisão).
- Identificar como ele é executado hoje (ou seja, em qual nível de capacidade e justificativas)
- Definir a importância desta área de processo para os objetivos estratégicos da organização, observando os benefícios típicos de sua execução, com pontuação baixa, média ou alta, e descrever as razões da pontuação.

- Definir o risco desta área de processo, em relação ao nível de capacidade atual, observando os problemas típicos de sua não execução, se a área continuar a ser executado como é atualmente, com pontuação baixo, médio ou alto, e descrever as razões da pontuação.
- Posiciona-lo no quadro em função das respostas de importância e risco e rever estas respostas.

A Tabela 39 descreve um trecho do questionário para relevância de uma área de processo.

Tabela 39 - Trecho do questionário para relevância de área de processo

<b>1. Identificações</b>
<b>1.1 Identificação da Unidade Organizacional:</b> [    ] <i>max 72 cars.</i>
<b>1.2 Identificação da Área de Processo:</b> [    ] <i>max 32 cars.</i>
<b>1.3 Identificação do Modelo de Capacidade de Processo:</b> [    ] <i>max 32 cars.</i>
<b>2. Área de Processo e a Unidade Organizacional</b>
<b>2.1 Comentários gerais sobre qualquer assunto relacionado:</b> (tamanho sugerido: 10 linhas) [    ] <i>max 800 caracteres</i>
<b>2.2 Correspondência desta área de processo na Unidade Organizacional:</b> (tamanho sugerido: 10 linhas) [    ] <i>max 800 caracteres</i>
<b>2.3 Comentários sobre como esta área de processo é realizada na Unidade Organizacional:</b> (tamanho sugerido: 10 linhas) [    ] <i>max 800 caracteres</i>
<b>2.4 Estimativa do Nível de Capacidade do processo atual na Unidade Organizacional:</b> [    ]
<b>2.5 Importância da área de processo para o desempenho global da Unidade Organizacional:</b> [    ] <b>Justifique:</b> (tamanho sugerido: 10 linhas) [    ] <i>max 800 caracteres</i>
<b>2.6 Risco de manter o desempenho atual da área de processo para o desempenho global da Unidade Organizacional:</b> [    ] <b>Justifique:</b> (tamanho sugerido: 10 linhas) [    ] <i>max 800 caracteres</i>

A Figura 92 e a Figura 93 ilustram exemplos de preenchimento do quadro de relevância das áreas de processo.

Cada dimensão tem três unidades: baixo, médio e alto. Os processos posicionados, por exemplo, em (alto, alto) são aqueles considerados muito importante para a organização e que, se continuar a serem praticados da forma atual, devem causar um grande impacto negativo no desempenho global da unidade organizacional. Na Figura 92, por exemplo, a área de processo PP do CMMI-SE/SW é considerada com importância alta e risco médio, enquanto que a área de processo VER é considerada



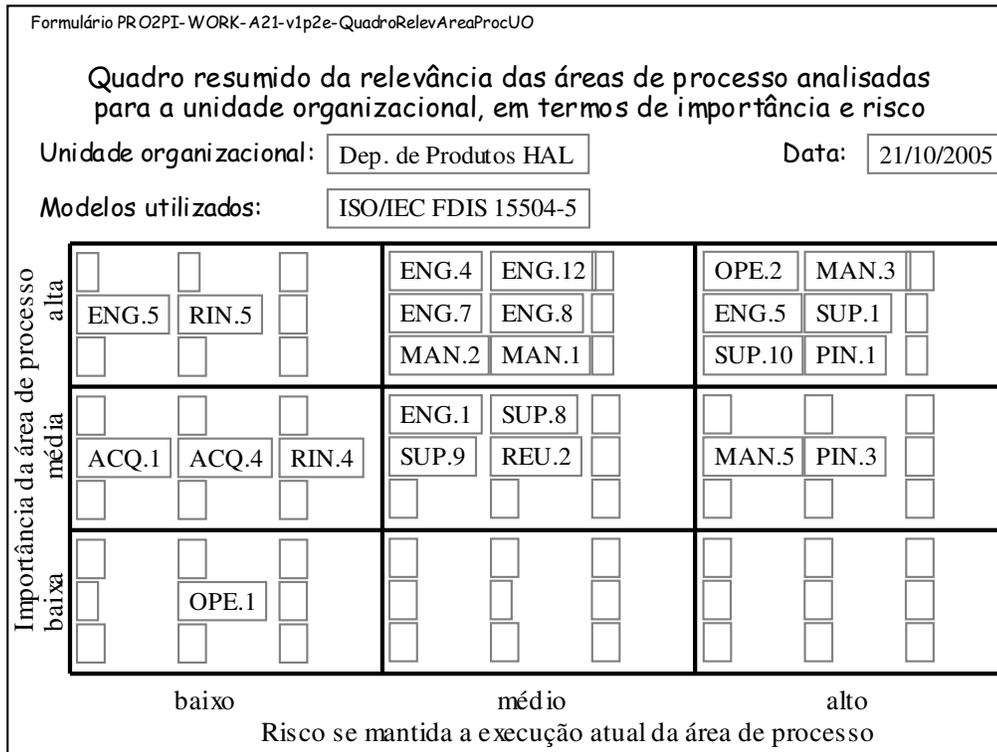


Figura 93 – Exemplo de relevância de processos da ISO/IEC 15504-5

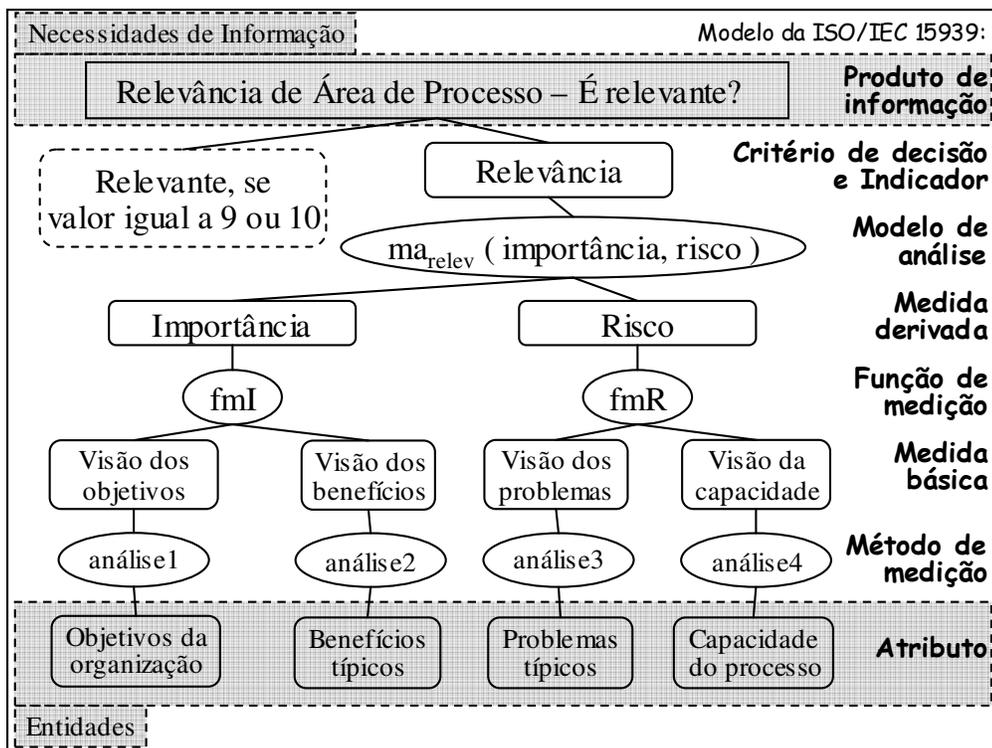


Figura 94 – Modelo de medição para relevância de uma área de processo

Tabela 40 - Valores para combinações da importância e risco

<b>Risco:</b> <b>Importância</b>	Baixo	Médio	Alto
Alta	6	9	10
Média	4	7	8
Baixa	2	3	5

Na Figura 92, por exemplo, a relevância da área de processo PP é pontuada como 9 e da área de processo VER como 8. Na Figura 93, como outro exemplo, a relevância do processo ENG.4 é pontuada como 6 e do processo REU.2 como 7.

### **Atividade 2.5: Mapear subsídios nos processos**

**Propósito:** Compilar os subsídios gerados nas atividades 2.2 a 2.4 e juntamente com subsídios de processos mais utilizados em outras experiências, mapeá-las em uma escala de processos mais importantes.

Cada subsídio deve ser relacionado a um ou mais processos. Esses relacionamentos são quantificados de tal forma que a cada processo é obtido um número com a quantidade de vezes que o processo foi citado com um fator de ajuste.

**Produtos de entrada:** A06-DescNeg; A09-QuestFOrg e A10-QuadroRelevProc

**Produtos de saída:** A24-MapSubsidProc e A14-R4DescSubsid

#### **Tarefas:**

- Especialista e multiplicadores analisam A15-R5DescObjEstr e relacionam resultado com áreas de processo
- Especialista e multiplicadores analisam os questionários A09-QuestFOrg e relacionam resultado com áreas de processo
- Especialista e multiplicadores analisam os questionários A20-InfoRelevProcUO e A21-QuadroRelevProcUO em relaciona resultado com áreas de processo
- Especialista e multiplicadores compilam resultados dos relacionamento e produzem uma pontuação para cada área de processo

A pontuação é um número normalizado entre 0 e 10 que representa o grau de relevância do processo para a melhoria.

No exemplo dos objetivos estratégicos ilustrado na Figura 84, da seção 7.2.3, resultante da atividade 2.2, o processo CUS.2 Fornecimento foi associado ao objetivo Melhorar Fornecimento de Software. No exemplo do quadro preenchido ilustrado na Figura 93, resultante da atividade 2.2, o processo MAN.3 Gerência de Projetos recebeu um número alto por ter sido posicionado em (alto, alto). Os processos posicionados em (médio, alto), (médio, médio) e (alto, médio) recebem números médios.

A Tabela 41 ilustra exemplos de respostas aos questionários utilizados na atividade 2.3 e um relacionamento feito pela equipe de trabalho associado a cada resposta. No caso a pergunta é: Relate três ou mais ações para a melhoria da organização que você gostaria que fossem realizadas - podem ser ações novas, modificações de ações já existentes ou mesmo eliminação de alguma ação. O mapeamento é realizado para os processos da ISO/IEC 15504-5. Às vezes não é fácil decidir com qual processo uma resposta deve ser associada ou outras vezes uma resposta não é associada a processo algum. Mas como são muitas respostas, geralmente em torno de 500 respostas, eventuais erros em poucos relacionamentos não impactam o resultado final.

Tabela 41 - Exemplos de respostas para questão 3.10 e mapeamento para 15504-5

<b>Resposta</b>	<b>Processo da ISO/IEC 15504-5</b>
Equipe exclusiva para suporte técnico ao cliente	OPE.2 Suporte ao cliente
Método para controle de versão de software	SUP.8 Gerência de configuração
Comunicar melhor a estratégia da empresa	MAN.1 Alinhamento organizacional
Definição de um plano de cargos e salários	RIN.1 Gerência de recursos humanos
Melhorar a gerência dos projetos	MAN.3 Gerência de projeto
Melhorar a estrutura física da empresa	RIN.4 Infra-estrutura
Melhorar estimativas e planejamento dos projetos	MAN.3 Gerência de projetos

Os processos informados na atividade 2.3 recebem números médios. Note que a associação de números é normalizada pela importância da origem do mapeamento. No final desse mapeamento o resultado é um número normalizado para cada processo e aqueles com maior número são os candidatos a serem escolhidos.

### **Atividade 2.6: Escolher perfil de capacidade**

**Propósito:** Escolher um PRO2PI composto geralmente por de 3 a 5 áreas de processo, com base em uma análise do resultado do mapeamento dos subsídios nos processos.

Composto por discussões sobre o resultado do mapeamento, considerando limitações sobre a quantidade de processos que devem ser melhorados a cada ciclo. A equipe escolhe um conjunto preliminar de processos.

Composta pela análise do conjunto como um sistema. Um processo pode precisar de outro. Após a seleção dos mais indicados, deve ser verificado se outro processo também deve ser necessário.

**Produto de entrada:** A14-R4DescSubsid

**Produto de saída:** A17-R7DescPRO2PI

**Tarefas:**

- Especialista e Multiplicadores discutem o mapeamento dos subsídios
- Especialista e Multiplicadores escolhem primeira proposta para PRO2PI

Em duas organizações foram escolhidos os perfis de capacidade de processo ilustrados na Figura 95.

Processo ISO/IEC FDIS 15504-5	Nível de Capacidade
SPL.1: Prospecção de Fornecimento .....	2
SPL.2: Liberação de Produto .....	2
SPL.3: Apoio para Aceitação do Produto .....	2
MAN.3: Gerência de Projeto .....	2
SUP.5: Garantia da Qualidade .....	1
MAN.1: A linha mento Organizacional.....	1
ENG.1: Elicitação de Requisitos.....	1
ENG.8: Teste de Software .....	1
SUP.1: Documentação .....	1
SUP.2: Gerência de Configuração .....	1

Processo ISO/IEC FDIS 15504-5	Nível de Capacidade
OPE.2: Suporte ao Cliente .....	2
MAN.3: Gerência de Projeto .....	2
SUP.5: Garantia da Qualidade .....	2
MAN.1: A linha mento Organizacional ...	1
PIM.1: Estabelecimento de Processo .....	1

Figura 95 – Dois exemplos de perfis de capacidade de processo

**Atividade 2.7: Apresentar e revisar o perfil**

**Propósito:** Apresentar e discutir PRO2PI com patrocinador e diretoria, revisar PRO2PI baseado no resultado da discussão, e obter aprovação do PRO2PI revisado.

**Produto de entrada:** A17-R7DescPRO2PI

**Produto de saída:** A17-R7DescPRO2PI

**Tarefas:**

- Especialista apresenta PRO2PI para Patrocinador

- Especialista e Multiplicadores discutem o PRO2PI com o Patrocinador
- Especialista e Multiplicadores analisam resultado das discussões e podem ajustar o PRO2PI.

### **A3.3 PRO2PI-WORK Fase 3: Orientações para PRO2PI**

**Propósito:** Avaliação dos processos em relação ao PRO2PI, revisão do PRO2PI e identificação de orientações para ações de melhoria.

**Critério de entrada:** Um PRO2PI escolhido

**Critério de saída:** PRO2PI definido

Esta fase é realizada com as sete atividades descritas a seguir.

#### **Atividade 3.1: Preparar próximas atividades**

**Propósito:** Selecionar e adaptar material para o PCP definido na Fase 2.

**Produtos de entrada:** Material para as atividades

**Produtos de saída:** Material ajustado para as atividades

**Tarefas:**

- Especialista material
- Especialista ajusta material selecionado

#### **Atividade 3.2: Apresentar técnicas de engenharia de processo**

**Propósito:** Apresentar técnicas de engenharia de processo, especialmente para definição, representação e avaliação de processo e para orientações e planejamento de ações de melhoria de processo.

**Produtos de entrada:** A10-ApresTecEPS, e Lista de presença

**Produtos de saída:** Resumo das discussões, e Lista de presença atualizada e preenchida.

**Tarefas:**

- Especialista apresenta técnicas de engenharia de processo
- Participantes e Especialista realizam sessão de perguntas e respostas

#### **Atividade 3.3: Avaliar práticas correntes**

**Propósito:** Avaliar práticas correntes em relação ao PRO2PI

**Produtos de entrada:** A16-R6PCPAtual

**Produtos de saída:** A16-R6PCPAtual

**Tarefas:**

- Especialista e Multiplicadores se reúnem com os participantes
- Participantes relatam como cada referência do PRO2PI está sendo executada

Sessão conjunta com elaboração de identificação da situação atual em relação ao PCP. Podem ser utilizados quadros na parede para viabilizar um trabalho conjunto. Para cada atividade do PCP, buscar situação atual. Também obter uma representação do processo atual.

**Atividade 3.4: Consolidar avaliação e revisar perfil**

**Propósito:** Consolidar uma avaliação preliminar e superficial dos processos a partir do resultado da atividade anterior, analisar o resultado e revisar PRO2PI em função desta análise.

**Produto de entrada:** A16-R6PCPAtual

**Produto de saída:** A16-R6PCPAtual

**Tarefas:**

- Consolidar avaliação como um PCP
- Aplicar PRO2PI-MEAS para determinar viabilidade
- Revisar PRO2PI até atender viabilidade

**Atividade 3.5: Identificar orientações para melhoria**

**Propósito:** Identificar ações e orientações para melhoria, buscando atingir o PRO2PI.

**Produtos de entrada:** A17-R8OrientMelh

**Produtos de saída:** A17-R8OrientMelh

**Tarefas:**

- Especialista e Multiplicadores discutem o PRO2PI
- Especialista e Multiplicadores identificam ações para melhoria

**Atividade 3.6: Consolidar orientações e revisar perfil**

**Propósito:** Analisar resultado da atividade anterior e consolidar. Inclui concluir retrato do PCP atual (podendo alterá-lo) e revisar PCP projetado. Usar estimativa de esforço para concluir PCP projetado.

**Produtos de entrada:** A17-R8OrientMelh

**Produtos de saída:** A17-R8OrientMelh, e A03-RelatPRO2PI

**Tarefas:**

- Especialista e Multiplicadores revisam PRO2PI
- Especialista e Multiplicadores analisam PRO2PI

### **Atividade 3.7: Apresentar perfil estabelecido**

**Propósito:** Apresentar resultado final a um grupo de pessoas e obter comentários.

**Produto de entrada:** A03-RelatPRO2PI

**Produto de saída:** A03-RelatPRO2PI

**Tarefas:**

- Especialista apresenta PRO2PI para Participantes
- Especialista e Multiplicadores discutem PRO2PI com os Participantes

## **A3.4 PRO2PI-WORK Fase 4: Conclusão do trabalho**

**Propósito:** Conclusão do trabalho.

**Critério de entrada:** PRO2PI definido

**Critério de saída:** Relatório produzido e entregue

Esta fase é realizada com as três atividades descritas a seguir.

### **Atividade 4.1: Consolidar relatório final**

**Propósito:** Consolidar resultado do trabalho como um relatório final

**Produto de entrada:** A03-RelatPRO2PI

**Produto de saída:** A03-RelatPRO2PI

**Tarefas:**

- Especialista produz A03-RelatPRO2PI a partir do resultado do trabalho
- Especialista revisa e prepara A03-RelatPRO2PI em mídia adequada para envio

### **Atividade 4.2: Entregar relatório final**

**Propósito:** Entregar relatório final para o patrocinador e obter aprovação.

**Produtos de entrada:** A03-RelatPRO2PI

**Produtos de saída:** Registro de recebimento e aprovação

**Tarefas:**

- Especialista envia A03-RelatPRO2PI ao Patrocinador e solicita aprovação.

- Especialista confirma recebimento de A03-RelatPRO2PI pelo Patrocinador
- Especialista aguarda aprovação pelo *Patrocinador*

### Atividade 4.3: Encerrar o trabalho realizado

**Propósito:** Avaliar o trabalho realizado, incluindo aderência ao processo, formas de realização, resultados, problemas e oportunidades de melhoria (*post-mortem*).

**Produtos de entrada:** A03-RelatPRO2PI

**Produto de saída:** A19-AvalTrabalho

#### Tarefas:

- Especialista verifica todo o trabalho
- Especialista identifica resultados a serem armazenados, incluindo A03-RelatPRO2PI.
- Especialista armazena resultados.

Para concluir a descrição das fases e atividades do método PRO2PI-WORK, a Figura 96 ilustra o método PRO2PI-WORK, PRO2PI e seus principais relacionamentos em termos de modelos de capacidade de processo, outros tipos de modelos, modelos de processo mais genérico, modelo de processo mais específico da organização e o processo da organização.

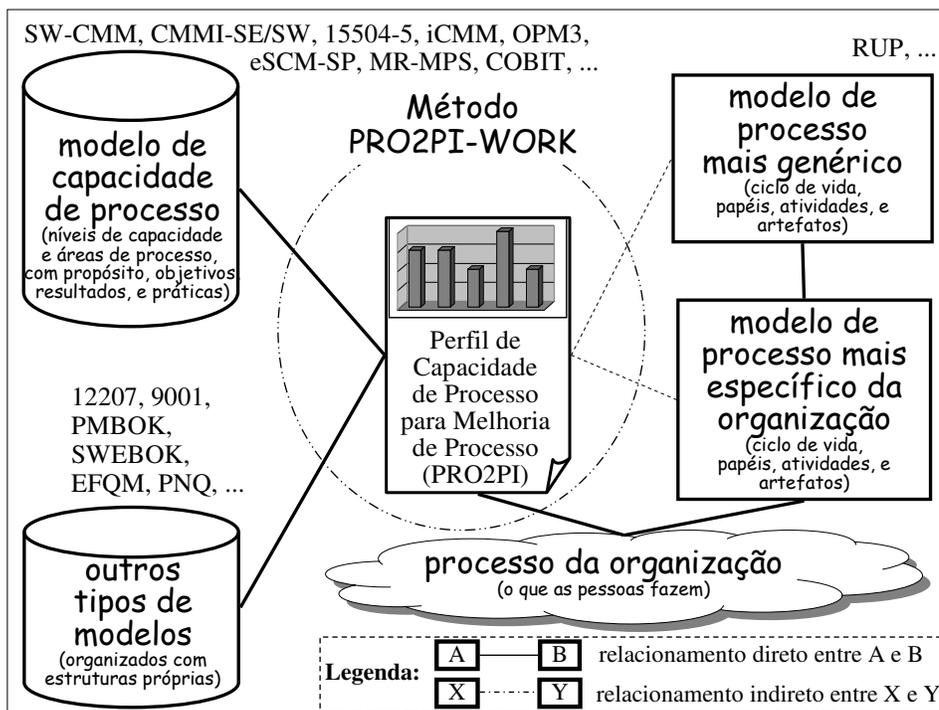


Figura 96 – Método PRO2PI-WORK, PRO2PI e seus principais relacionamentos